Social Media Activity Analysis using K-means and K-Medioids Clustering

-Soham Neeraj Agarkar & Huakang Lu

Abstract

This project aims to make use of the Facebook-Live-Sellers Dataset acquired from UCI machine Learning Repository. It is a CSV Dataset consisting of 7,050 Facebook posts of various types (text, deferred and live videos, images). These posts were extracted from the Facebook pages of 10 Thai fashion and cosmetics and retail sellers from March 2012 - June 2018. The dataset was collected via the Facebook API and anonymized in compliance with the Facebook Platform Policy for Developers. For each Facebook post, the dataset records the resulting engagement metrics comprising shares, comments, and emoji reactions within which we distinguish traditional "likes" from recently introduced emoji reactions, that are "love", "wow", "haha", "sad" and "angry".

The Goal of the project is to perform K-means and K-Medioids classifications on the data and derive suitable insights as well as compare the results of clustering methods.

DataSet

Variable Name	Role	Type	Missing Values
status_is	ID	Integer	No
status_type	Feature	Categorical	No
status_published	Feature	Categorical	No
num_reactions	Feature	Integer	No
num_comments	Feature	Integer	No
num_shares	Feature	Binary	No
num_likes	Feature	Integer	No
num_loves	Feature	Binary	No
num_wows	Feature	Binary	No
num_hahas	Feature	Binary	No
num_sads	Feature	Binary	No
num_angrys	Feature	Binary	No

Our Dataset contains **7051** instances and **12** features. As mentioned in the table above, our data doesn’t contain any missing values or duplicate rows.

The image below shows the head of the data. A small sample of the first 6 rows in the dataset.

Description: df [6 × 11]

	status_type <fctr>	status_publish... <chr>	num_reactions <int>	num_comments <int>
1	4	4/22/2018 6:00	529	512
2	2	4/21/2018 22:45	150	0
3	4	4/21/2018 6:17	227	236
4	2	4/21/2018 2:29	111	0
5	2	4/18/2018 3:22	213	0
6	2	4/18/2018 2:14	217	6

6 rows | 1-5 of 11 columns

The image below shows the characteristics of each feature:

```
'data.frame':   7050 obs. of  11 variables:
 $ status_type     : Factor w/ 4 levels "1","2","3","4": 4 2 4 2 2 2 4 4 2 2 ...
 $ status_published: chr  "4/22/2018 6:00" "4/21/2018 22:45" "4/21/2018 6:17"
"4/21/2018 2:29" ...
 $ num_reactions   : int  529 150 227 111 213 217 503 295 203 170 ...
 $ num_comments    : int  512 0 236 0 0 6 614 453 1 9 ...
 $ num_shares      : int  262 0 57 0 0 0 72 53 0 1 ...
 $ num_likes       : int  432 150 204 111 204 211 418 260 198 167 ...
 $ num_loves       : int  92 0 21 0 9 5 70 32 5 3 ...
 $ num_wows        : int  3 0 1 0 0 1 10 1 0 0 ...
 $ num_hahas       : int  1 0 1 0 0 0 2 1 0 0 ...
 $ num_sads        : int  1 0 0 0 0 0 0 0 0 0 ...
 $ num_angrys      : int  0 0 0 0 0 0 3 1 0 0 ...
```

The following image provides us a summary of the dataframe:

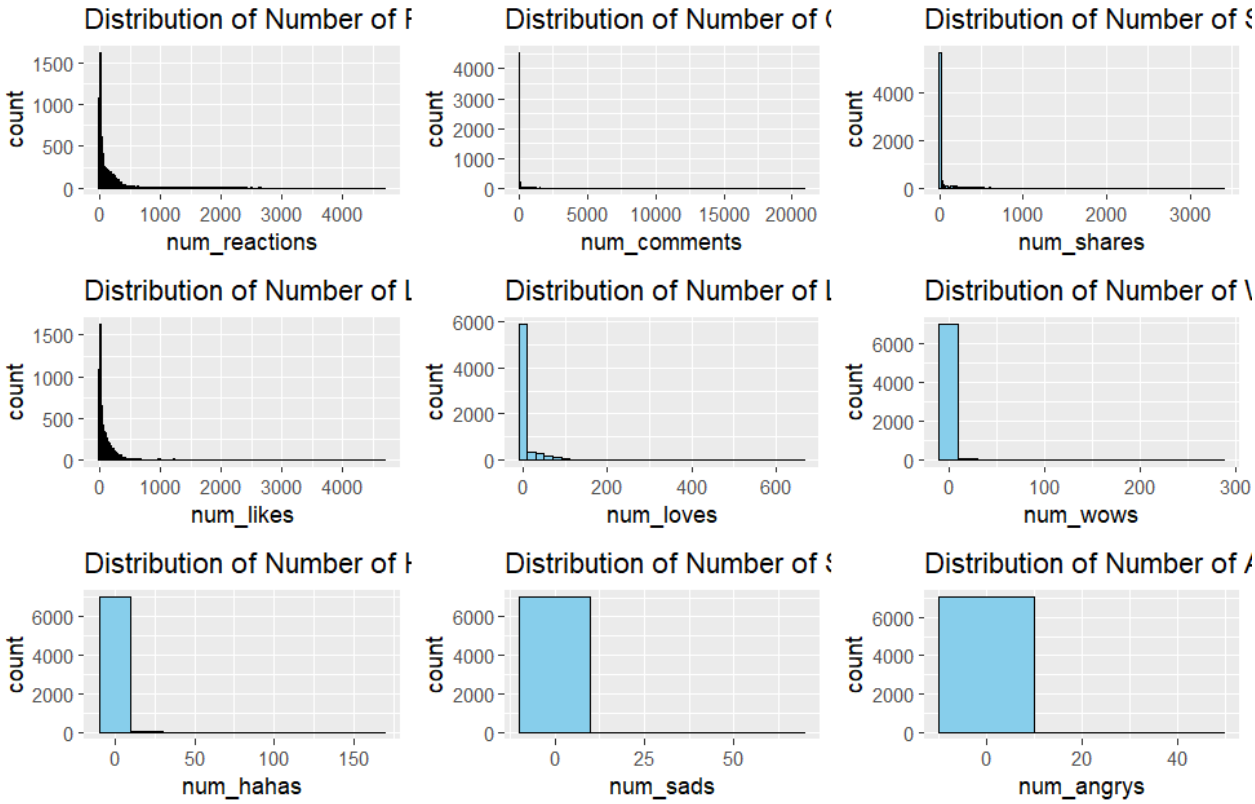
```
 status_type status_published   num_reactions     num_comments       num_shares
 1:  63      Length:7050        Min.   :   0.0   Min.   :    0.0   Min.   :   0.00
 2:4288      Class :character   1st Qu.:  17.0   1st Qu.:    0.0   1st Qu.:   0.00
 3: 365      Mode  :character   Median :  59.5   Median :    4.0   Median :   0.00
 4:2334                         Mean   : 230.1   Mean   :  224.4   Mean   :  40.02
                                3rd Qu.: 219.0   3rd Qu.:   23.0   3rd Qu.:   4.00
                                Max.   :4710.0   Max.   :20990.0   Max.   :3424.00
   num_likes        num_loves         num_wows          num_hahas
num_sads
 Min.   :   0.0   Min.   :  0.00   Min.   :  0.000   Min.   : 0.0000   Min.   :
0.0000
 1st Qu.:  17.0   1st Qu.:  0.00   1st Qu.:  0.000   1st Qu.: 0.0000   1st Qu.:
0.0000
 Median :  58.0   Median :  0.00   Median :  0.000   Median : 0.0000   Median :
0.0000
 Mean   : 215.0   Mean   : 12.73   Mean   :  1.289   Mean   : 0.6965   Mean   :
0.2437
 3rd Qu.: 184.8   3rd Qu.:  3.00   3rd Qu.:  0.000   3rd Qu.: 0.0000   3rd Qu.:
0.0000
 Max.   :4710.0   Max.   :657.00   Max.   :278.000   Max.   :157.0000   Max.
:51.0000
   num_angrys
 Min.   : 0.0000
 1st Qu.: 0.0000
 Median : 0.0000
 Mean   : 0.1132
 3rd Qu.: 0.0000
 Max.   :31.0000
```

EDA

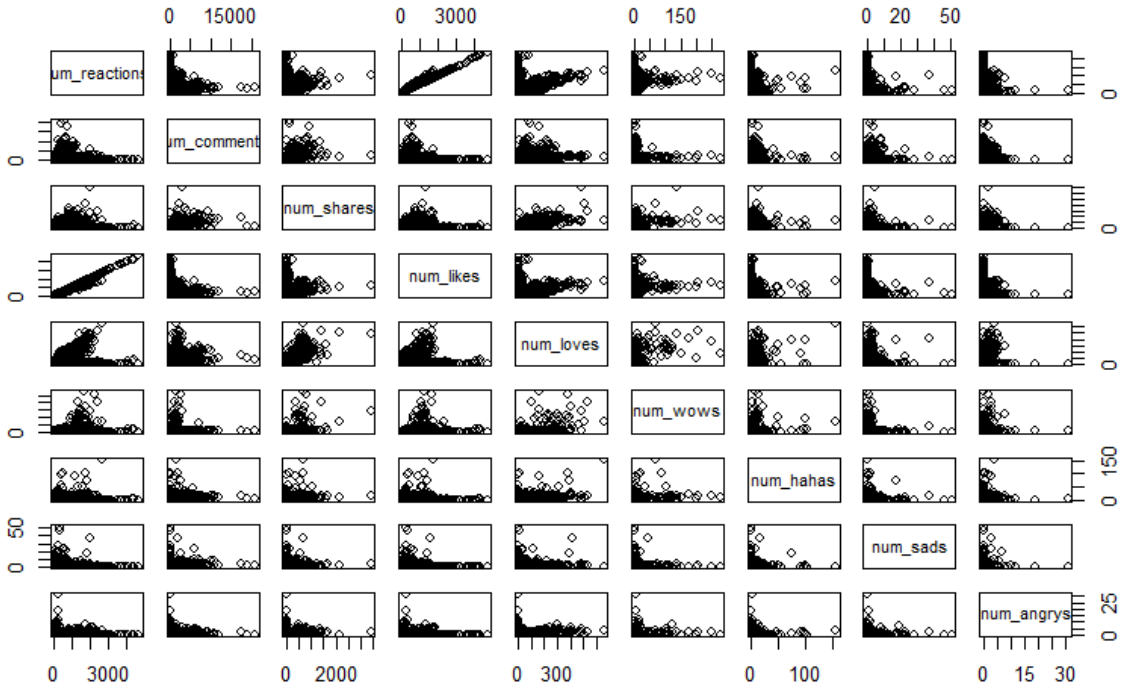
Plots

Since there aren't any missing values, our next step is immediately EDA using plots. To achieve this, we plotted a few graphs. The first being a count plot for all the numerical features. Additionally, since we drop the categorical variables **"status_published, status_type"** and" **status_id"** since they don't mean anything to us in the analysis and the models can't process them anyway.

Below is the count plot:

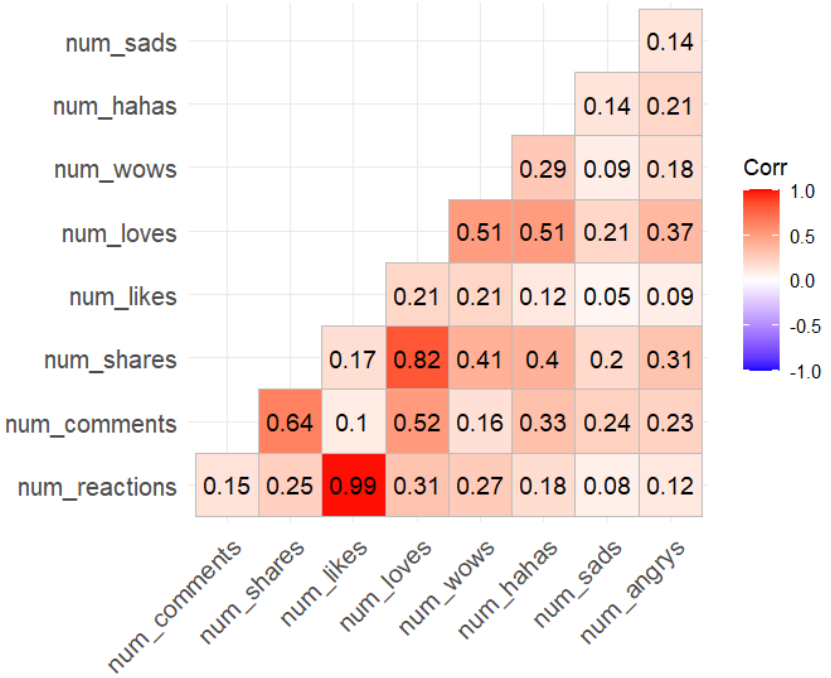


The above plot tells a tale of how interactions usually go on the internet. In most cases, the first interaction and also the most common one seems to be either a like or a love. This might even be an indicator of the target audience having a fairly short attention span. There number of interactions only increase by a little.



The above scatterplot matrix for the numerical variables shows the relation between the features. It is worth noting that num_likes shares a very positive correlation with num_reactions. In all other cases, it appears that either the features are positively correlated or not all.

To further understand the relation between the variables, we plotted a correlation matrix.



The plot above further corroborates our theory so far. num_reactions seems to have an extremely strong positive correlation with num_likes. This tells us that most users tend to interact with the post on Facebook through a simple like and usually nothing more.

It is also worth noting that num_shares also shares a positive correlation with num_comments and num_loves. This indicates that aside from the likes, users tend to move to commenting or sharing the post as interaction. And among comments, the heart emoji tends to be the most frequently occurring one.

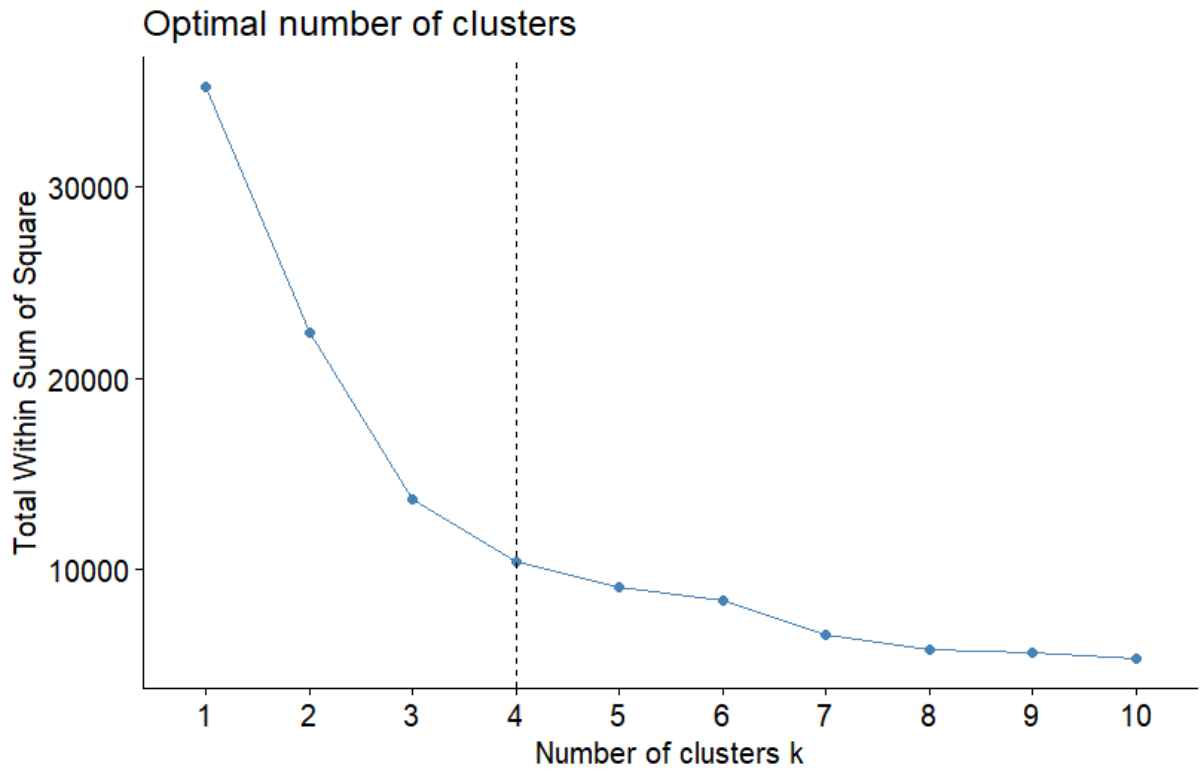
Having discovered all this, we knew the relationship we had to focus on was that of **“num_reactions, num_likes, num_shares, num_comments and num_loves”.**

So, we created a new dataframe of only those features. This feature was then scaled so that the data is on even footing and finally modelling was performed.

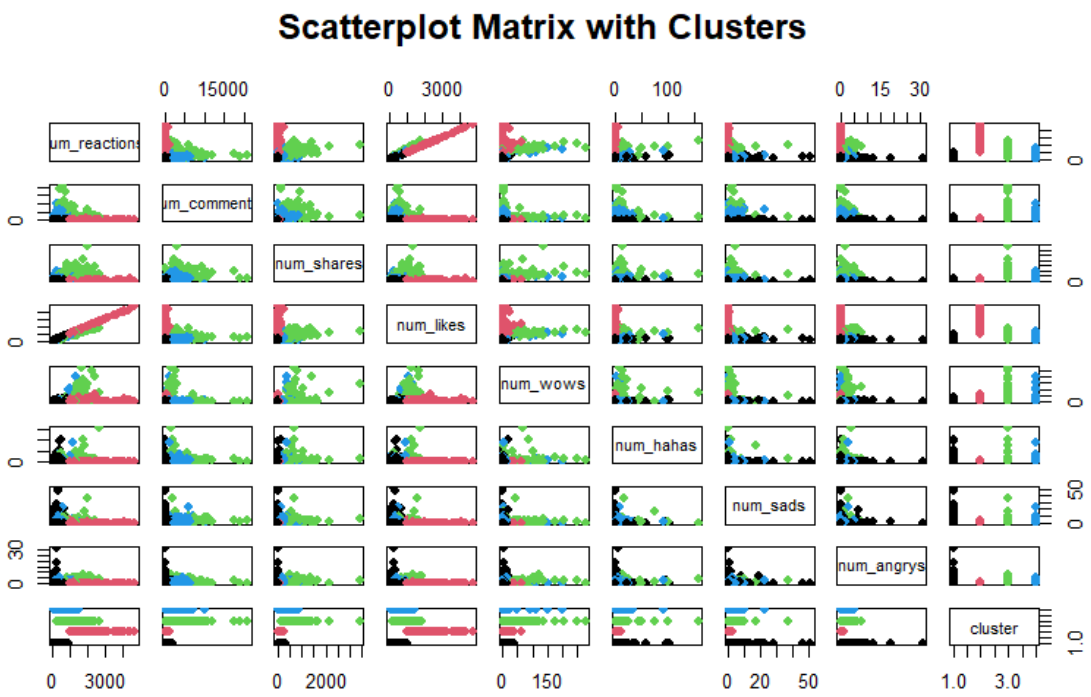
Modelling

K-Means

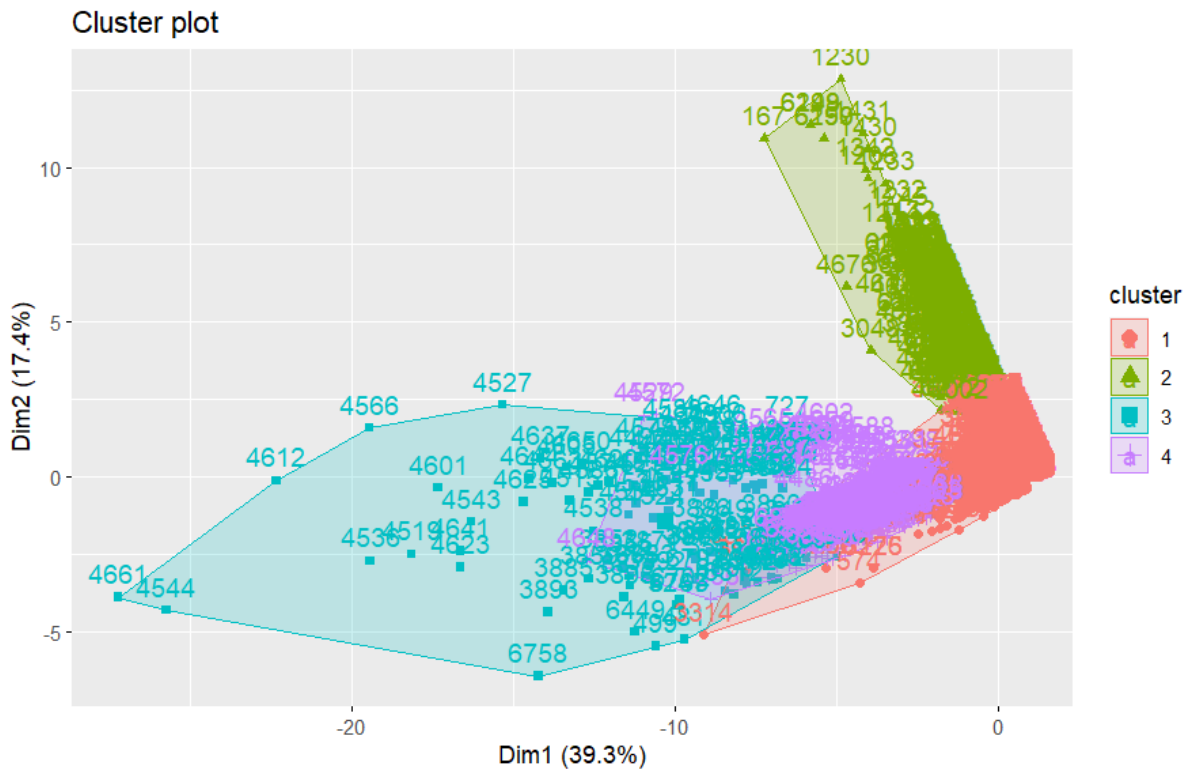
The first step in any clustering classification is to determine the optimal number of clusters. The cluster was obtained using the “Total within sum of Squares” Method.



In our case, the optimal number of clusters is 4 for K-Means.



Above is the scatterplot matrix from earlier but its is visualized with K-Means clusters. Attached below is another plot that makes is easier to visualize the clusters and its boundaries.

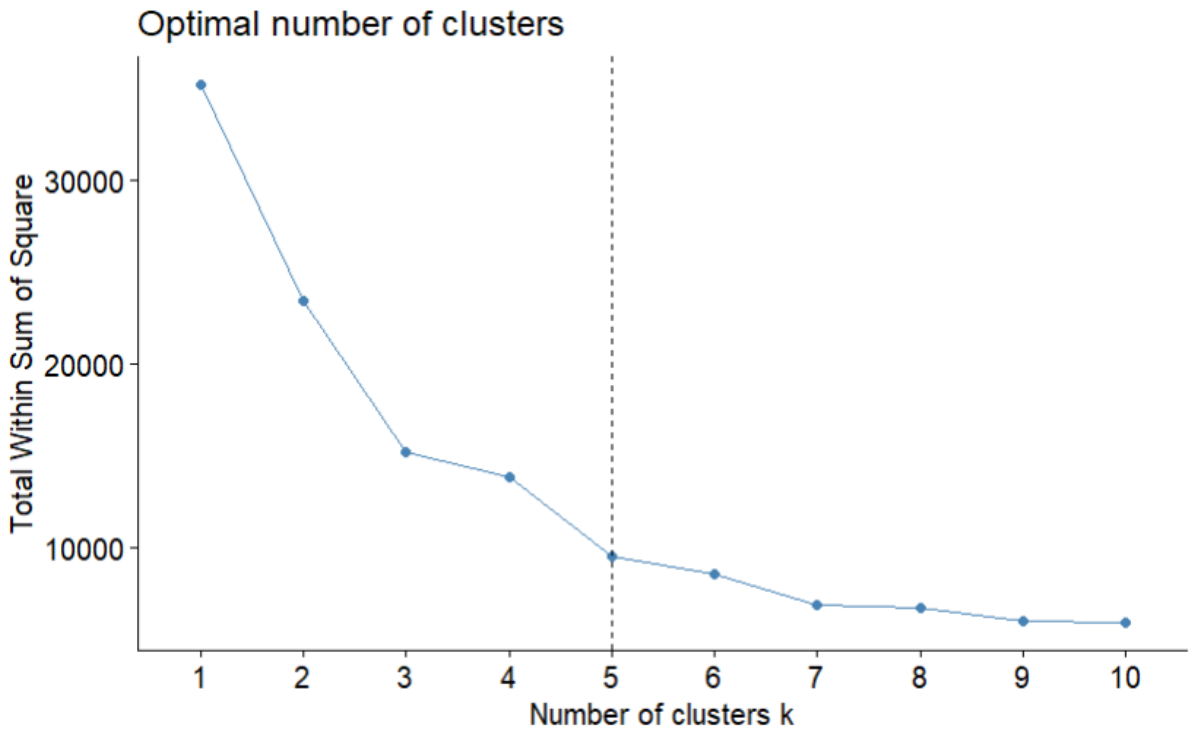


The silhouette score for this model is **0.75688.** A silhouette score of 0.75688 suggests that the clusters are well-separated and that each data point is relatively close to its own cluster centroid

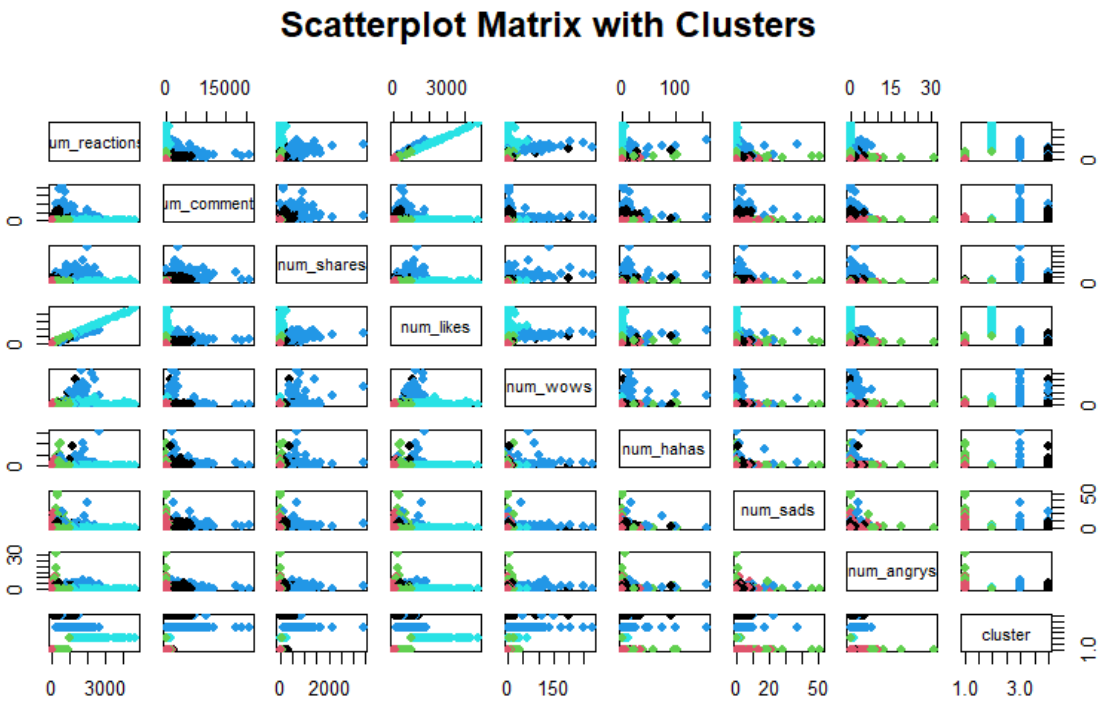
compared to other clusters. This indicates a strong clustering structure in our data, where the clusters are distinct and well-defined.

K-Medioids

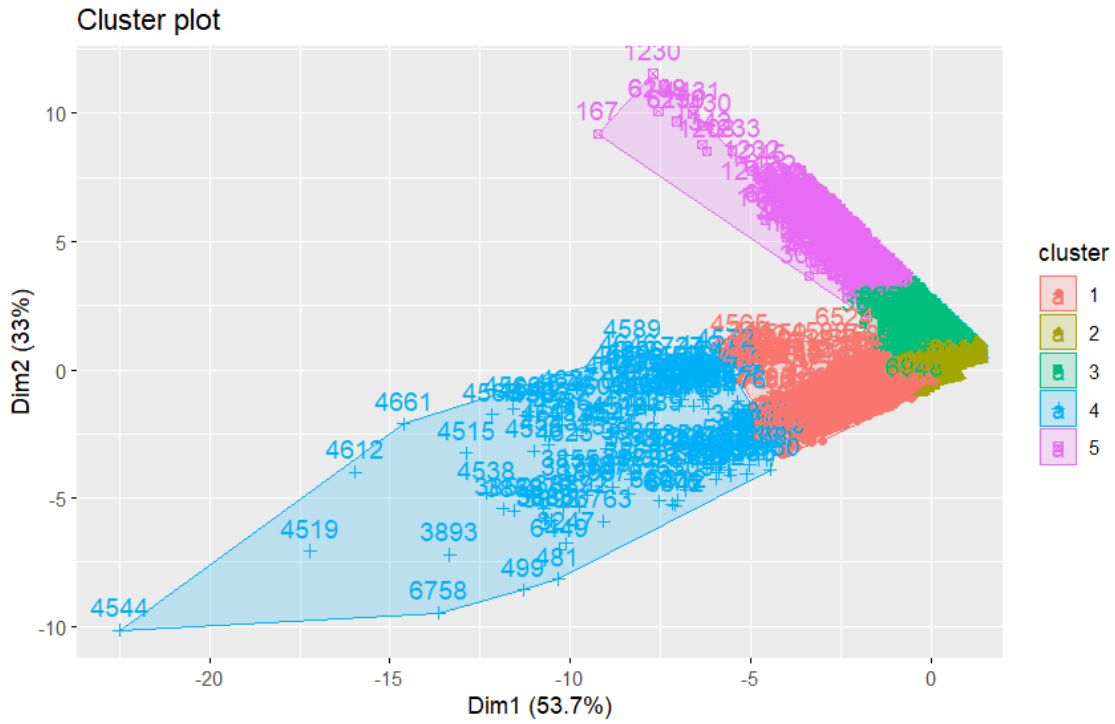
We begin by finding the optimal clusters for K-Medioids.



In our case, our optimal number of clusters were 5. Like K-means, we acquired the clusters using the “Total Within Sum of Square” method.



Above is the scatterplot matrix from earlier but it is visualized with K-Medioids clusters



Above is the plot that visualizes the clustering for K-Medioids.

The silhouette score for K-Medioids is 0.57905.

Conclusions

If the silhouette score of K-medoids (0.57905) is lower than that of K-means (0.75688), it suggests that the clusters produced by K-means are more compact and well-separated compared to K-medoids.

A higher silhouette score generally indicates better-defined and more separated clusters. So, in this comparison, K-means appears to have produced better-defined clusters than K-medoids for our specific dataset.

In summary, based solely on the silhouette scores provided K-means appears to have performed better in terms of cluster separation and cohesion compared to K-medoids for our dataset.