K-Means Clustering

Background

The goal of this exercise is to segment the TripAdvisor reviews of 249 high volume reviewers into distinct clusters, by using K-Means clustering techniques, and identify potential uses of the generated cluster scheme.

Data Source

The dataset is a flat csv file, obtained from Marsh, 2020, as an altered subset of data, originally taken from TripAdvisor, a travel review website. It consists of 249 observations from various reviewers that post comments in high volumes, within six review categories. These can be seen in more detail in the provided data dictionary in Appendix 1.

Data Transformation and Cleaning

UserId Variable

The UserId variable is a unique identifier and was removed from the dataset since it does not need to participate in any of the analysis.

Other Variables

All remaining variables were abbreviated to a maximum of three characters for ease in summarization during the various tasks below. They were also appended with the '_TG' suffix, since they were all altered in some way. The updates can be seen in seen in Appendix 2, to complement the Data Dictionary.

Standardization

The variables were then standardized and used in various clustering techniques. The transformed Sports and Religious variables were identified as two centroids of interest and used heavily in the clustering process. Variables based on percentage seemed to be more tightly clustered than Age, Income and Number of Reviews. Since all variables needed to be transformed, the normalization approach was used to accommodate those variables that were more dispersed, as seen in the Descriptive Analysis below. This is beneficial if further investigation is required that involves a mix of highly and lowly dispersed variables. All transformation source code can be seen in Appendix 3.

Descriptive Data Analysis

A summary of the variables was generated to identify trends in dispersion or presence of any outliers. All source code can be seen in Appendix 4.

```
> summary(reviews_data_TG)
```

Spt_TG	Rel_TG	Nat_TG	Thr_TG	Shp_TG
Min. :0.00508	Min. :0.109	Min. :0.0883	Min. :0.112	Min. :0.106
1st Qu.:0.01191	1st Qu.:0.156	1st Qu.:0.1658	1st Qu.:0.163	1st Qu.:0.146
Median :0.01920	Median :0.179	Median :0.2085	Median :0.187	Median :0.183
Mean :0.01866	Mean :0.184	Mean :0.2099	Mean :0.197	Mean :0.188
3rd Qu.:0.02483	3rd Qu.:0.211	3rd Qu.:0.2656	3rd Qu.:0.234	3rd Qu.:0.216
Max. :0.03234	Max. :0.274	Max. :0.3772	Max. :0.303	Max. :0.319

Pcc_TG	Age_TG	Inc_TG	Nbr_TG
Min. :0.144	Min. :18.0	Min. : 963	Min. :353
1st Qu.:0.180	1st Qu.:27.0	1st Qu.:23790	1st Qu.:494
Median :0.197	Median :38.0	Median :47986	Median :595
Mean :0.202	Mean :37.4	Mean :47433	Mean :596
3rd Qu.:0.225	3rd Qu.:48.0	3rd Qu.:67165	3rd Qu.:710
Max. :0.269	Max. :55.0	Max. :99949	Max. :843

No outliers were found. However, the variables can be grouped into categories of high dispersion, Age_TG to Nbr_TG variables (highlighted), and low dispersion, Spt_TG to Pcc_TG percentage variables. To accommodate the dispersion difference, the variables were standardized via normalization, seen below.

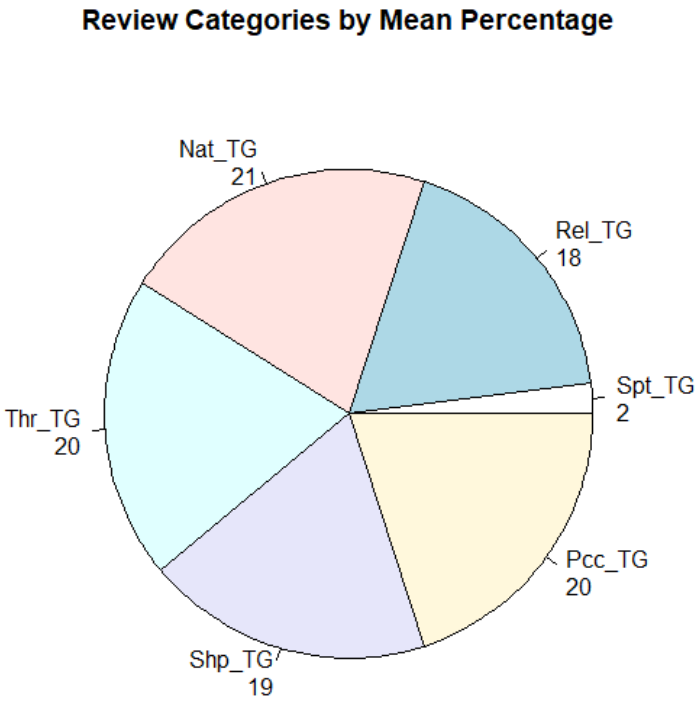
```
> summary(reviews_ndata_TG)
```

Spt_n_TG	Rel_n_TG	Nat_n_TG	Thr_n_TG	Shp_n_TG
Min. :-1.8873	Min. :-2.060	Min. :-1.9831	Min. :-1.979	Min. :-1.669
1st Qu.:-0.9387	1st Qu.:-0.784	1st Qu.:-0.7187	1st Qu.:-0.793	1st Qu.:-0.857
Median : 0.0746	Median :-0.154	Median :-0.0235	Median :-0.243	Median :-0.089
Mean : 0.0000	Mean : 0.000	Mean : 0.0000	Mean : 0.000	Mean : 0.000
3rd Qu.: 0.8564	3rd Qu.: 0.711	3rd Qu.: 0.9065	3rd Qu.: 0.834	3rd Qu.: 0.577
Max. : 1.9001	Max. : 2.423	Max. : 2.7269	Max. : 2.423	Max. : 2.662

Pcc_n_TG	Age_n_TG	Inc_n_TG	Nbr_n_TG
Min. :-1.962	Min. :-1.7137	Min. :-1.6758	Min. :-1.89306
1st Qu.:-0.730	1st Qu.:-0.9171	1st Qu.:-0.8526	1st Qu.:-0.79327
Median :-0.162	Median : 0.0565	Median : 0.0199	Median :-0.00548
Mean : 0.000	Mean : 0.0000	Mean : 0.0000	Mean : 0.00000
3rd Qu.: 0.804	3rd Qu.: 0.9416	3rd Qu.: 0.7116	3rd Qu.: 0.89151
Max. : 2.268	Max. : 1.5612	Max. : 1.8938	Max. : 1.92890

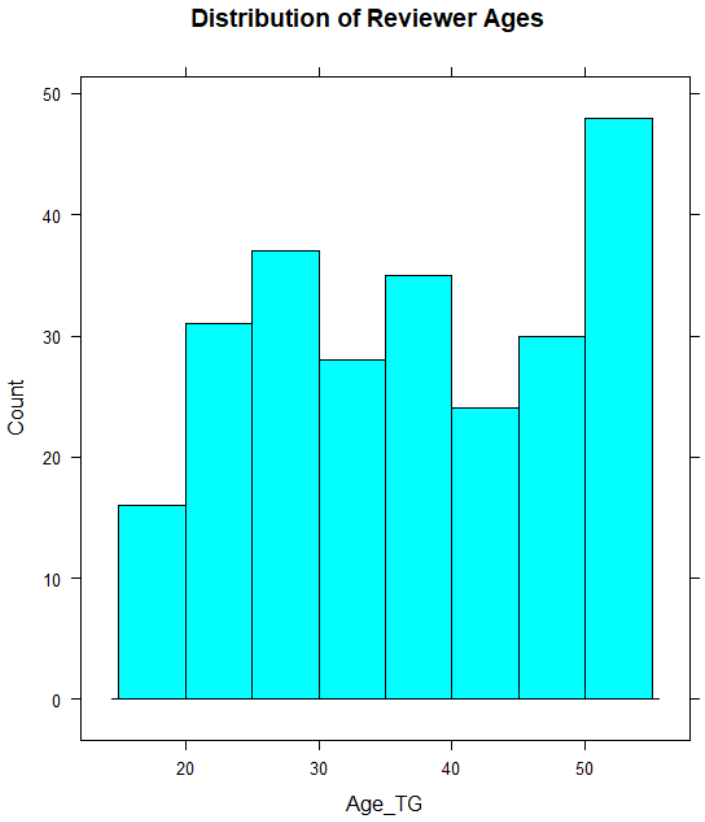
After standardization, the means of all the variables are now 0 and a relative uniformity in dispersion was seen across all variables.

From the means in the summary table of the original values, a pie chart was constructed to depict the mean percentage of each review category, from Spt_TG to Pcc_TG variables. All source code can be seen in Appendix 4.

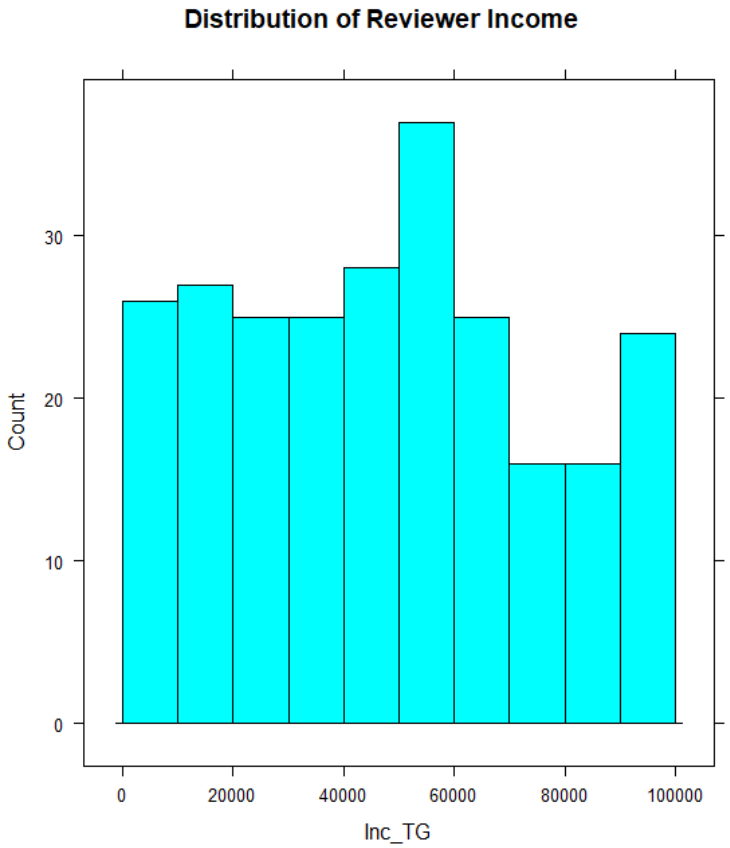


Like the original summary table above, this pie chart shows that almost all the categories were of similar proportions, of approximately 20%. However, the Spt_TG category was significantly smaller than any of the others, accounting for only 2%.

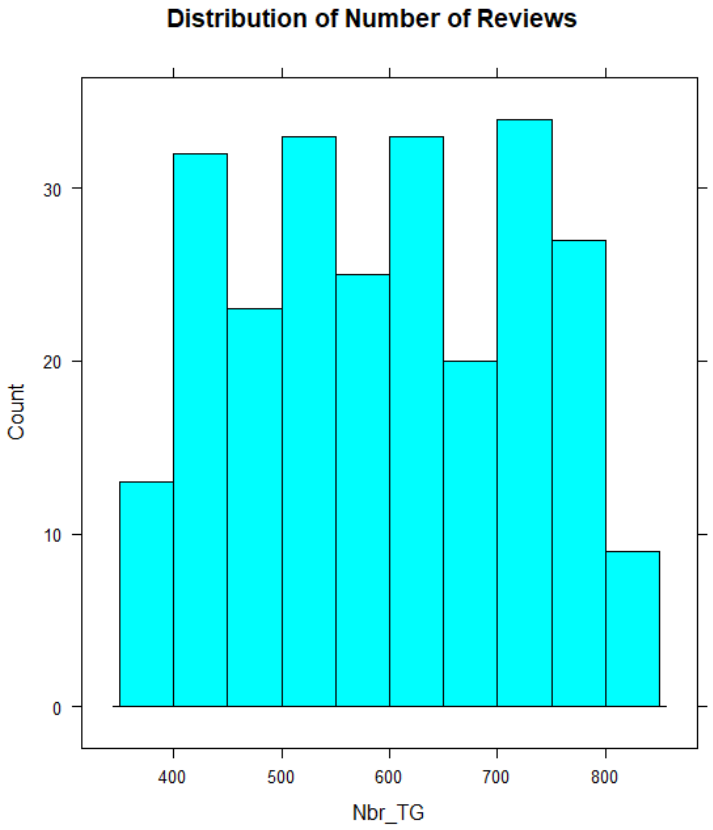
A series of histograms were generated to display the distributions of the remaining variables, Age_TG, Inc_TG, and Nbr_TG. Since the two centroids of interest are Spt_TG and Rel_TG, individual histograms were also generated for those variables as well. All source code can be seen in Appendix 4.



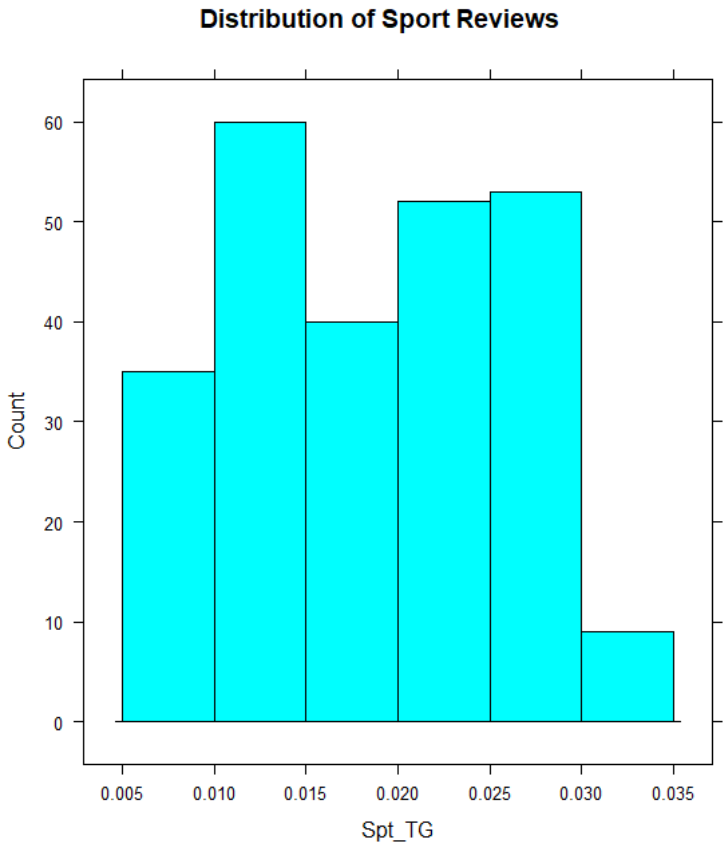
Most of the reviewer ages seem to fluctuate between 25 and 45 years of age, with an average count of approximately 30. The lowest range of reviewer ages was between 15 and 20 years, with a count of approximately 15, while the largest range was between 45 and 50 years, with a count of approximately 48. This distribution is not normal and there may be some bias since the ages are reviewer-reported instead of verified or calculated.



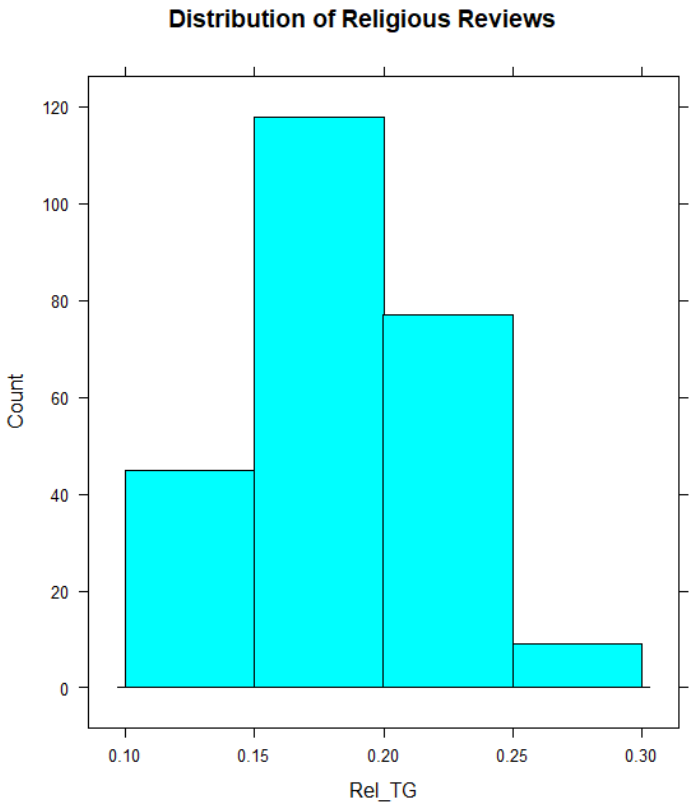
Most of the income ranges from $0 to $100,000 were consistent with an average frequency of approximately 25 reviewers. The exceptions to this observation are the $50,000 to $60,000 range, having the highest frequency of approximately 38 reviewers and the $70,000 to $90,000 range, having the lowest frequency of approximately 15 reviewers.



The number of reviews fluctuates drastically, with alternating dips and rises in the distribution. The highs of the distribution are more evenly consistent, seen with ranges 400 to 450, 500 to 550, 600 to 650, and 700 to 750 all having a stable frequency of approximately 33 reviewers. The two outer ranges are very low with frequencies of approximately 13 reviewers between 350 and 400, and approximately 8 reviewers between 800 and 850.



The proportion of Sport reviews with the highest frequency of approximately 60 reviewers was seen in the 1% to 1.5% range. This is followed by the second highest frequency of 50 reviewers for proportions between 2% and 3%. It should be noted that the overall mean percentage of this distribution is only 2% of all the reviews, seen in the pie chart above.

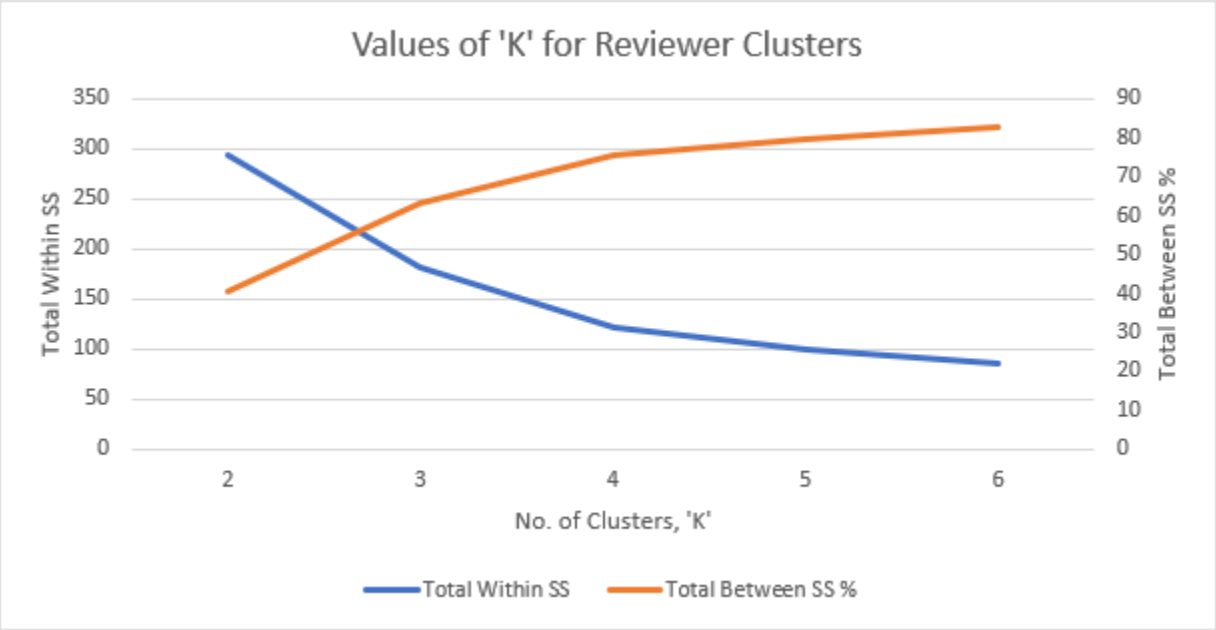


The distribution of the Religious review percentages is only categorized in four bins, due to the tight clustering of the data. The largest frequency of reviews of approximately 118 reviewers can be seen with a proportion range of 15% to 20%, while the lowest of approximately 10 reviewers can be seen in the 25% to 30% range.

Clustering

Clusters were created, using the transformed centroid variables, Spt_n_TG and Rel_n_TG. This technique was executed five times to generate several clusters (K) from 2 to 6, inclusive. Five cluster schemes were generated and the results of each can be found in Appendix 5. A summary table was then derived from the schemes and was used to create a Within Cluster Sum of Squares (WSS) plot. The source code can also be seen in Appendix 5. The 'Elbow' method (Marsh, 2020 b) was then used to identify a suitable K value for further analysis.

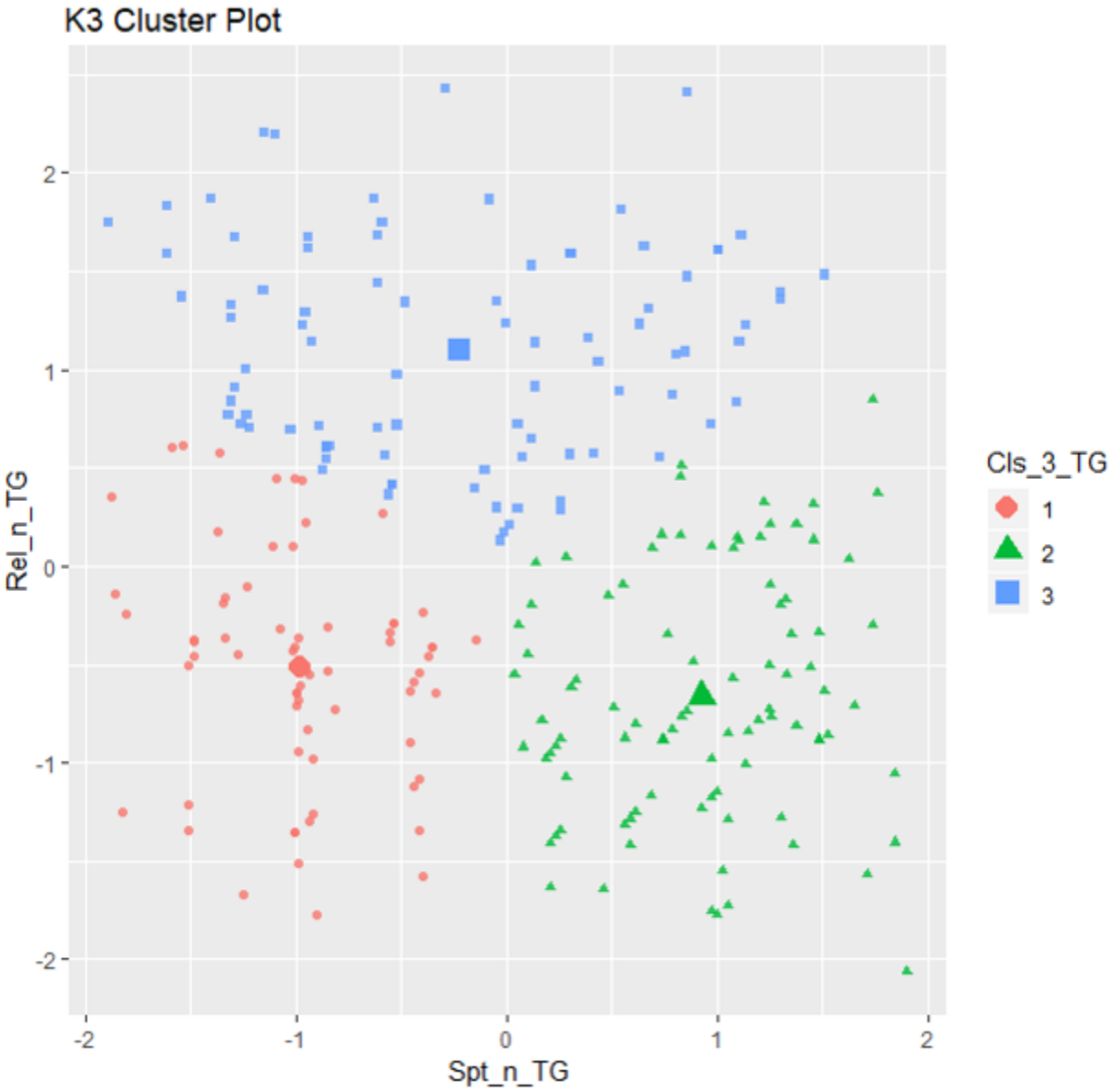
K	2	3	4	5	6
Total Within SS	294	182.5	121	99.9	85.3
Total Between SS %	41	63	76	80	83



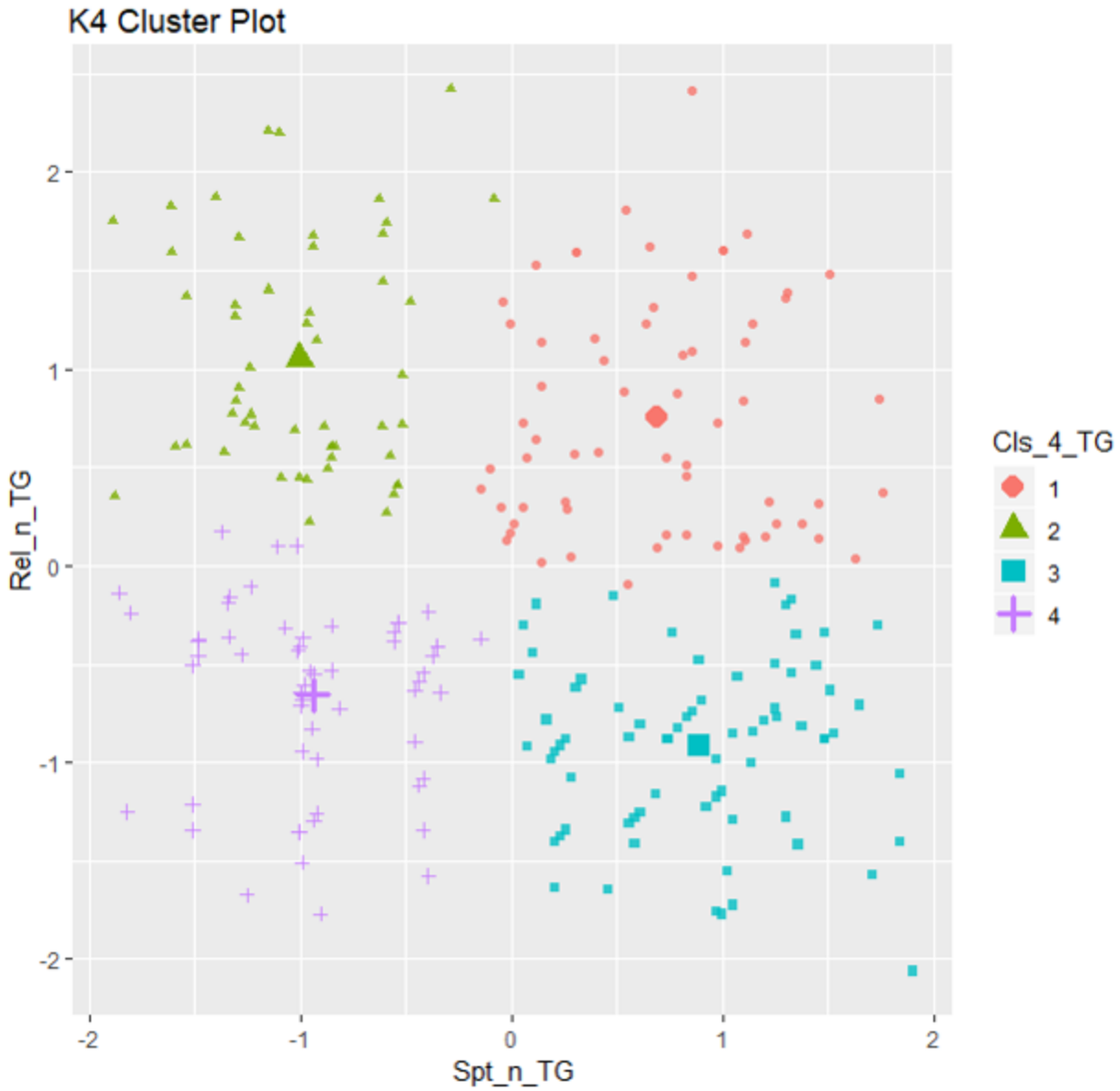
Based on the WSS plot above, the 'Elbow' of the Total Within SS seems to occur around K4, as a point of inflection. This is also reflected in the summary table above, where the Total Between SS % of 76% for K4 seems to mark the beginning of more stable values, with lower differences of 4% and 3% between K4 and K5, and K5 and K6, respectively.

Evaluation of Clusters

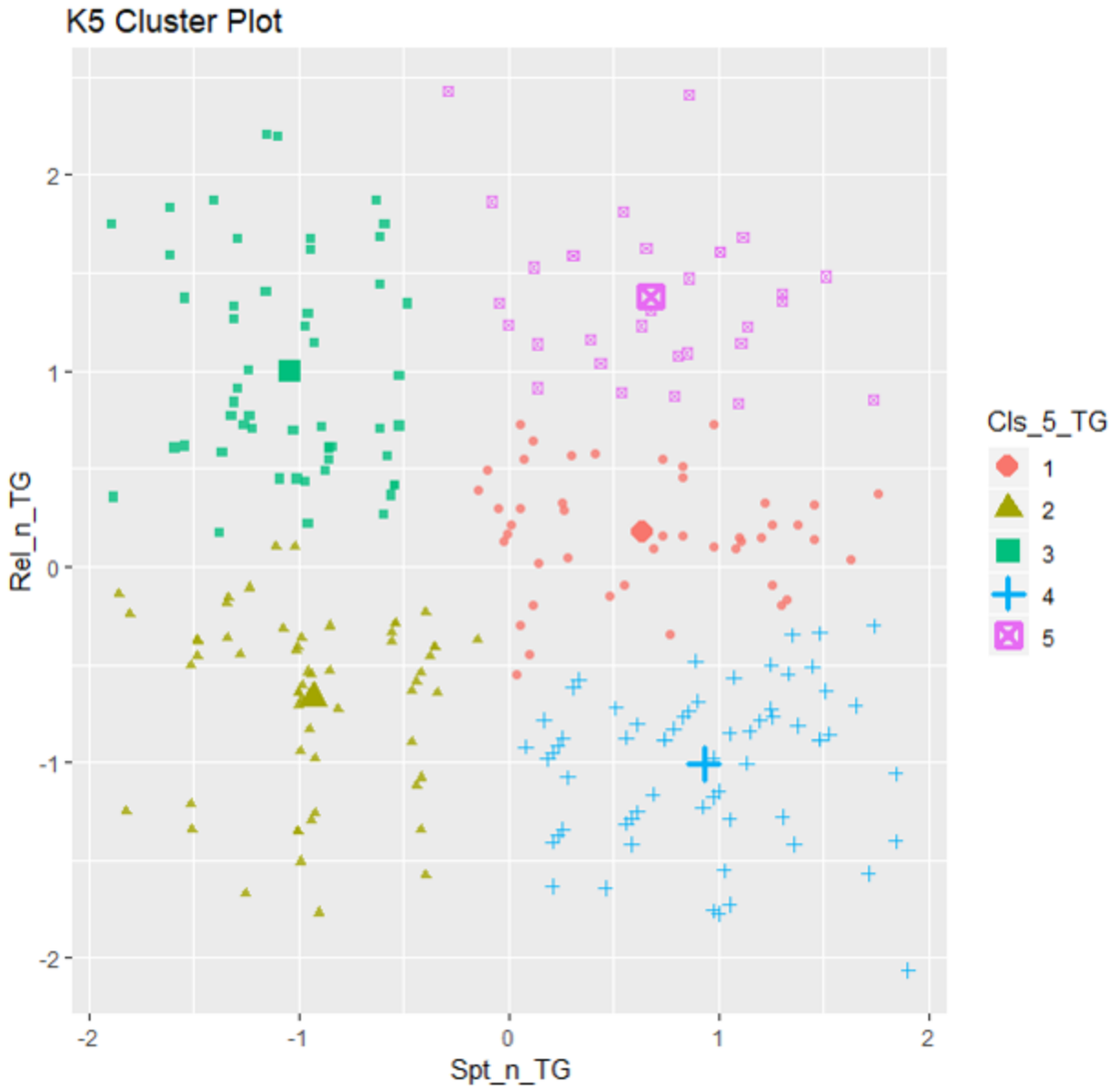
From the K4 cluster chosen above, using the 'Elbow' method, three scatter plots were generated to compare the K4 cluster with the cluster below it (K3) and the cluster above it (K5) and identify which of the three cluster plots best describes the data. All source code can be seen in Appendix 6.



This plot contains three clusters. Cluster 3 dominates the top half of the data on its own. The entire range of Spt_n_TG, from high to low, is accounted for by high Rel_n_TG. This suggests that the observations may not be as tightly clustered around their centroid as they could be.



This plot divides the data into four balanced clusters. The wide range for Spt_n_TG, originally seen in the K3 plot above is now split better into two distinct categories in the top half of the data.



This plot segments the data into five clusters. It introduces an intermediate cluster in the upper vertical half of high Spt_n_TG observations, which does not seem to provide any further distinction than in K4, except a notable colour change and a higher repositioning of the cluster 5 centroid. The lower vertical half of Spt_n_TG observations remains mostly unaffected by a fifth cluster.

A cluster size of four (K4) seems to best describe the data, where it strikes a balance between creating equidistant and equisized clusters. It visually divides the data into four quarters.

A summary table was then created for the selected clustering scheme to identify any trends amongst the variables. The means of the variables were generated for each cluster group along with the number of observations for each cluster. Variables were relatively ranked by clusters. Clusters were identified by appropriate descriptors and suggestions for the use of the cluster scheme were provided. All source code can be seen in Appendix 6.

Ck4_TG	Sptm_TG	Relm_TG	Natm_TG	Thrm_TG	Shpm_TG	Pccm_TG	Agem_TG	Incm_TG	Nbrm_TG	Nobs_TG
1	0.0236	0.212	0.157	0.185	0.219	0.204	37.5	53899	699	65
2	0.0114	0.223	0.162	0.189	0.210	0.205	38.1	45885	485	53
3	0.0251	0.151	0.260	0.203	0.160	0.200	37.0	44982	689	72
4	0.0119	0.160	0.251	0.211	0.167	0.198	37.0	44694	468	59

Similarities

All reviewer clusters shared a similar mean interest in picnicking, theatre visits and age. These reviewers would enjoy marketing materials and promotional discounted events that they can inform their followers of. The Sports category has a significantly low proportion of reviews. Even though clusters can still be ranked with high and low sports ratios, as seen in the K4 cluster chart above, considerations should be relative to the overall proportion instead.

Cluster 1 - Religious Sports Fan High Spender

Cluster 1 consists of high sports and high religious reviewers. It also consists of high shopping, low nature enthusiasm, and high income. Marketing of luxurious products and services and some sporting events and products would be highly attractive to this group. A religious spin can be added to promotions to keep the reviewers interested.

Cluster 2 – Religious Neutral High Spender

Cluster 2 consists of low sports and high religious reviewers. It also consists of high shopping, low nature enthusiasm and moderately high income. The same marketing and promotional events as Cluster 1 can apply to this group but with significantly less emphasis on sports.

Cluster 3 - Outgoing Sports Fan Low Spender

Cluster 3 consists of high sports and low religious reviewers. It also consists of high nature enthusiasm, low shopping, and moderate income. Moderately priced products and services would be popular in this group. High emphasis on nature and adventure-related activities and products would cater well to this group. An inclusion of sporting materials would be ideal as well.

Cluster 4 – Outgoing Neutral Low Spender

Cluster 4 consists of low sports and low religious reviewers. It also consists of high nature enthusiasm, low shopping, and lower income. More cost-effective services and products for lower incomes should be considered, especially deals and savings. Like Cluster 3, a high emphasis can be placed on nature and adventure related marketing material.

References

Marsh, D. (2020). *Assignment 7 – Clustering: K-Means*. eConestoga. Retrieved August 5, 2020 from https://conestoga.desire2learn.com/d2l/lms/dropbox/user/folder_submit_files.d2l?db=349651&grpid=0&isprv=0&bp=0&ou=354666

Marsh, D. (2020) b. *PROG8430 – Data Analysis, Modeling and Algorithms: Lecture 11 – Classification: Clustering*. eConestoga. Retrieved August 5, 2020 from https://conestoga.desire2learn.com/d2l/le/content/354666/viewContent/7447584/View

APPENDIX 1: Data Dictionary (Marsh, 2020)

Name	Description
User ID	Unique Identifier
Sports	Percentage of reviews related to sporting locations.
Religious	Percentage of reviews related to religious locations.
Nature	Percentage of reviews related to natural locations.
Theatre	Percentage of reviews related to theatres.
Shopping	Percentage of reviews related to shopping locations.
Picnic	Percentage of reviews related to picnic locations.
Age	Self-reported age
Income	Income inferred from geographic tax records.
Nbr	Total Number of Reviews

APPENDIX 2: Updated Variable Names

Dictionary Name	Dataset Name	New Name	Notes
User ID	UserId	N/A	Removed from dataset.
Sports	Sports	Spt_TG	Involved in rename transformation (minor).
Religious	Religious	Rel_TG	Involved in rename transformation (minor).
Nature	Nature	Nat_TG	Involved in rename transformation (minor).
Theatre	Theatre	Thr_TG	Involved in rename transformation (minor).
Shopping	Shopping	Shp_TG	Involved in rename transformation (minor).
Picnic	Picnic	Pcc_TG	Involved in rename transformation (minor).
Age	Age	Age_TG	Involved in rename transformation (minor).
Income	Income	Inc_TG	Involved in rename transformation (minor).
Nbr	Nbr	Nbr_TG	Involved in rename transformation (minor).
		Ck4_TG	Cluster variable added at the end of the analysis.

Notes:

- The transformed versions of these variables were used in an intermediate dataset for analysis purposes only.
- The primary intermediate dataset consisted only of the transformed versions of the two centroids of interest: Sports and Religious.
- The means of the variables were used in the cluster summary, depicted by an ‘m’ in the name.

APPENDIX 3: Transformation and Clean-up Source Code

Dropping and Renaming Variables

```
# drop unnecessary identifiers
reviews_data_TG <- reviews_data_TG[-c(1)]

# rename variables with _TG suffix
names(reviews_data_TG) <- c("Spt_TG", "Rel_TG", "Nat_TG", "Thr_TG",

                            "Shp_TG", "Pcc_TG", "Age_TG", "Inc_TG", "Nbr_TG")
```

Standardizing Variables

```
# summary to get an idea of the dispersion of data
summary(reviews_data_TG)

# creates standardization function
stdnrm_TG <- function(x) {
  return ((x-mean(x))/sd(x))
}

# to store normalized data
reviews_ndata_TG <- reviews_data_TG

reviews_ndata_TG$Spt_n_TG <- stdnrm_TG(reviews_data_TG$Spt_TG)
reviews_ndata_TG$Rel_n_TG <- stdnrm_TG(reviews_data_TG$Rel_TG)
reviews_ndata_TG$Nat_n_TG <- stdnrm_TG(reviews_data_TG$Nat_TG)
reviews_ndata_TG$Thr_n_TG <- stdnrm_TG(reviews_data_TG$Thr_TG)
reviews_ndata_TG$Shp_n_TG <- stdnrm_TG(reviews_data_TG$Shp_TG)
reviews_ndata_TG$Pcc_n_TG <- stdnrm_TG(reviews_data_TG$Pcc_TG)
reviews_ndata_TG$Age_n_TG <- stdnrm_TG(reviews_data_TG$Age_TG)
reviews_ndata_TG$Inc_n_TG <- stdnrm_TG(reviews_data_TG$Inc_TG)
reviews_ndata_TG$Nbr_n_TG <- stdnrm_TG(reviews_data_TG$Nbr_TG)

# drop unnecessary variables
reviews_ndata_TG <- reviews_ndata_TG[-c(1:9)]
```

APPENDIX 4: Descriptive Data Analysis Source Code

Summaries of Original and Standardized Variables

```
# summary of original variables
summary(reviews_data_TG)

# summary of transformed variables
summary(reviews_ndata_TG)
```

Pie Chart of Review Category Variables

```
# generate table of summaries for only the percentage variables
prct_means_TG <- round(sapply(reviews_data_TG[, 1:6], mean),2)* 100

# generate list of labels
prct_labels_TG <- paste(names(prct_means_TG), "\n", prct_means_TG, sep="")

# generate a pie chart of the percentages of the review categories
pie(prct_means_TG, labels = prct_labels_TG, main="Review Categories by Mean Percentage")
```

Histograms of Some Variables

```
# generate histogram of reviewers by age
histogram( ~ Age_TG, data = reviews_data_TG, breaks = 10, type = "count",
           main = "Distribution of Reviewer Ages")

# generate histogram of reviewers by income
histogram( ~ Inc_TG, data = reviews_data_TG, breaks = 10, type = "count",
           main = "Distribution of Reviewer Income")

# generate histogram of reviewers by income
histogram( ~ Nbr_TG, data = reviews_data_TG, breaks = 10, type = "count",
           main = "Distribution of Number of Reviews")

# generate histogram of reviewers by income
histogram( ~ Spt_TG, data = reviews_data_TG, breaks = 5, type = "count",
           main = "Distribution of Sport Reviews")

# generate histogram of reviewers by income
histogram( ~ Rel_TG, data = reviews_data_TG, breaks = 5, type = "count",
           main = "Distribution of Religious Reviews")
```

APPENDIX 5: Segmentation Schemes

Initial Preparation

```
# prepare dataset for clustering, choosing 2 centroid variables

# one for storing cluster variables
reviews_spt_rel_n_TG <- reviews_ndata_TG[c('Spt_n_TG','Rel_n_TG')]

# one only containing the centroid variables
cluster_spt_rel_n_TG <- reviews_ndata_TG[c('Spt_n_TG','Rel_n_TG')]
```

K 2

```
# create cluster 2
cluster2_TG <- kmeans(cluster_spt_rel_n_TG, iter.max=10, centers=2, nstart=10)

# add cluster tags to variables
reviews_spt_rel_n_TG$Cls_2_TG <- factor(cluster2_TG$cluster)

# generate dataframe of centers by factor
centers2_TG <- data.frame(cluster=factor(1:2), cluster2_TG$centers)

# rename default cluster column to match reviews_spt_rel_n_TG column for plotting
names(centers2_TG)[names(centers2_TG) == 'cluster'] <- 'Cls_2_TG'
```

```
> cluster2_TG
K-means clustering with 2 clusters of sizes 132, 117

Cluster means:
  Spt_n_TG Rel_n_TG
1 -0.78853  0.31183
2  0.88962 -0.35180

Clustering vector:
  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [41] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [81] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 2 1 2 2 2 2 2 1 1 2 1 2
[121] 1 2 2 2 2 2 1 2 2 1 2 2 2 2 2 2 1 2 1 1 1 1 2 1 2 1 2 1 1 2 2 2 2 2 1 2 1 2 1 2
[161] 1 2 1 2 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
[201] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2
[241] 2 2 2 2 2 2 2 2 2

Within cluster sum of squares by cluster:
[1] 170.35 123.66
 (between_SS / total_SS =  40.7 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

Cluster	1	2	Total
Within SS	170.35	123.66	294.01
Between SS %			40.7

K 3

```
# create cluster 3
cluster3_TG <- kmeans(cluster_spt_rel_n_TG, iter.max=10, centers=3, nstart=10)

# add cluster tags to variables
reviews_spt_rel_n_TG$Cls_3_TG <- factor(cluster3_TG$cluster)

# generate dataframe of clusters by factor
centers3_TG <- data.frame(cluster=factor(1:3), cluster3_TG$centers)

# rename default cluster column to match reviews_spt_rel_n_TG column for plotting
names(centers3_TG)[names(centers3_TG) == 'cluster'] <- 'Cls_3_TG'
```

```
> cluster3_TG
K-means clustering with 3 clusters of sizes 68, 93, 88

Cluster means:
  Spt_n_TG Rel_n_TG
1 -0.97716 -0.51285
2  0.92722 -0.66145
3 -0.22483  1.09532

Clustering vector:
  [1] 1 1 1 1 3 1 1 1 1 1 3 3 1 3 1 1 1 1 3 3 3 3 1 1 1 1 1 3 1 1 1 1 3 3 1 1 3 1 3 3
 [41] 1 1 1 1 3 1 1 3 3 1 1 3 3 1 1 1 3 3 1 3 1 3 1 1 1 1 1 3 1 3 3 3 1 1 1 1 3 1 3 3 3
 [81] 1 3 3 1 3 1 1 3 1 1 3 1 1 1 1 3 3 3 1 3 3 1 3 1 1 1 3 1 3 3 2 2 2 2 2 2 3 3 2 3 2
[121] 3 2 2 3 2 3 3 2 2 3 3 2 2 2 2 2 3 3 3 1 3 3 2 3 2 3 2 3 3 2 2 2 2 2 1 2 3 2 3 2
[161] 3 2 3 2 2 3 2 2 3 2 2 2 2 2 2 3 2 2 2 3 2 2 2 3 2 3 2 2 2 2 3 2 2 2 3 2 2 2 2 3
[201] 2 2 3 2 3 2 2 2 3 3 2 2 3 3 3 2 2 3 2 2 2 2 2 2 3 2 2 3 2 2 2 3 2 2 2 2 2 1 2 2
[241] 3 2 2 2 2 2 2 2 2 2

Within cluster sum of squares by cluster:
[1] 34.390 58.693 89.434
 (between_SS / total_SS =  63.2 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

Cluster	1	2	3	Total
Within SS	89.434	58.693	34.39	182.517
Between SS %				63.2

K 4

```
# create cluster 4
cluster4_TG <- kmeans(cluster_spt_rel_n_TG, iter.max=10, centers=4, nstart=10)

# add cluster tags to variables
reviews_spt_rel_n_TG$Cls_4_TG <- factor(cluster4_TG$cluster)

# generate dataframe of clusters by factor
centers4_TG <- data.frame(cluster=factor(1:4), cluster4_TG$centers)

# rename default cluster column to match reviews_spt_rel_n_TG column for plotting
names(centers4_TG)[names(centers4_TG) == 'cluster'] <- 'Cls_4_TG'
```

```
> cluster4_TG
K-means clustering with 4 clusters of sizes 65, 53, 72, 59

Cluster means:
  Spt_n_TG Rel_n_TG
1  0.68815  0.75341
2 -1.00504  1.05309
3  0.88877 -0.91601
4 -0.93990 -0.65818

Clustering vector:
  [1] 2 4 4 4 2 4 4 4 4 2 2 2 4 2 4 2 4 4 2 2 2 2 4 4 2 4 4 2 4 4 4 4 2 2 2 4 2 4 2 2
 [41] 4 4 4 2 2 4 2 2 4 4 1 2 4 4 4 2 2 4 2 4 2 4 4 4 2 4 2 2 2 2 2 4 4 4 4 2 4 2 2 2
 [81] 4 2 2 4 2 2 4 2 4 4 2 4 4 4 4 2 2 2 4 2 2 4 2 4 4 4 2 4 1 1 3 3 3 3 3 1 1 3 1 3
[121] 1 3 3 1 1 1 1 3 3 1 1 3 3 1 3 3 1 1 1 4 1 1 3 2 3 1 3 1 1 3 3 3 3 3 4 3 1 3 1 1
[161] 1 3 1 3 3 2 3 3 1 1 3 3 3 3 1 1 3 3 3 1 3 3 1 1 3 1 3 3 3 1 1 1 3 3 1 3 3 1 3 1
[201] 1 3 1 3 1 3 1 3 1 1 1 3 1 1 1 3 3 1 3 3 3 1 1 3 1 3 3 1 1 1 1 1 1 3 3 3 1 4 3 3
[241] 1 3 3 3 1 3 3 3 1

Within cluster sum of squares by cluster:
[1] 39.281 26.033 32.821 22.910
 (between_SS / total_SS =  75.6 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

Cluster	1	2	3	4	Total
Within SS	39.281	26.033	22.91	32.821	121.045
Between SS %					75.6

K 5

```
cluster5_TG <- kmeans(cluster_spt_rel_n_TG, iter.max=10, centers=5, nstart=10)

# add cluster tags to variables
reviews_spt_rel_n_TG$Cls_5_TG <- factor(cluster5_TG$cluster)

# generate dataframe of clusters by factor
centers5_TG <- data.frame(cluster=factor(1:5), cluster5_TG$centers)

# rename default cluster column to match reviews_spt_rel_n_TG column for plotting
names(centers5_TG)[names(centers5_TG) == 'cluster'] <- 'Cls_5_TG'
```

```
> cluster5_TG
K-means clustering with 5 clusters of sizes 45, 58, 52, 63, 31

Cluster means:
  Spt_n_TG Rel_n_TG
1  0.64123  0.17908
2 -0.93242 -0.67248
3 -1.04366  0.99423
4  0.92940 -1.00777
5  0.67558  1.37854

Clustering vector:
  [1] 3 2 2 2 3 2 2 2 2 3 3 3 2 3 2 3 2 2 3 3 3 3 2 2 3 2 2 3 2 2 2 2 3 3 3 2 3 2 3 3
 [41] 2 2 3 3 3 2 3 3 2 2 5 3 2 2 2 3 3 2 3 2 3 2 2 2 3 2 3 3 3 3 3 2 2 2 2 3 2 3 3 3
 [81] 2 3 3 2 3 3 2 3 2 2 3 2 2 2 2 3 3 3 2 3 3 2 3 2 2 2 3 2 1 1 4 1 4 1 4 1 5 4 1 4
[121] 1 4 4 5 1 1 1 4 4 5 1 4 4 1 4 1 5 1 1 2 1 1 1 5 4 5 4 1 5 4 4 4 4 1 2 4 1 4 5 1
[161] 5 1 5 4 4 5 4 4 5 1 4 4 4 4 1 5 4 1 4 5 4 4 1 5 4 5 4 4 4 1 5 1 4 4 5 4 4 1 4 5
[201] 1 4 5 4 5 4 1 4 1 5 1 4 5 5 5 4 1 5 4 4 4 1 1 1 1 4 4 5 1 1 1 5 1 4 4 4 5 2 4 4
[241] 5 4 4 4 1 4 4 4 1

Within cluster sum of squares by cluster:
[1] 17.773 22.018 22.793 24.985 12.327
 (between_SS / total_SS =  79.9 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

Cluster	1	2	3	4	5	Total
Within SS	22.793	24.985	12.327	17.773	22.018	99.896
Between SS %						79.9

K 6

```
# create cluster 6
cluster6_TG <- kmeans(cluster_spt_rel_n_TG, iter.max=10, centers=6, nstart=10)

# add cluster tags to variables
reviews_spt_rel_n_TG$Cls_6_TG <- factor(cluster6_TG$cluster)

# generate dataframe of clusters by factor
centers6_TG <- data.frame(cluster=factor(1:6), cluster6_TG$centers)

# rename default cluster column to match reviews_spt_rel_n_TG column for plotting
names(centers6_TG)[names(centers6_TG) == 'cluster'] <- 'Cls_6_TG'
```

```
> cluster6_TG
K-means clustering with 6 clusters of sizes 45, 34, 56, 48, 31, 35

Cluster means:
  Spt_n_TG Rel_n_TG
1  0.74259 -1.15039
2  0.12268  0.22080
3 -0.95602 -0.68569
4 -1.08422  1.04685
5  0.70249  1.37248
6  1.32043 -0.28962

Clustering vector:
  [1] 4 3 3 3 4 3 3 3 3 4 4 4 3 4 3 4 3 3 4 4 4 4 3 3 4 3 3 4 3 3 3 3 4 4 4 3 4 3 4 4
 [41] 3 3 4 4 4 3 4 4 3 3 5 4 3 3 3 2 4 3 4 3 4 3 3 3 2 3 4 4 2 2 4 3 3 3 3 4 3 4 4 4
 [81] 3 4 4 2 4 4 3 4 3 3 4 3 3 3 3 4 4 4 3 4 4 3 4 3 3 3 4 3 2 2 1 2 1 2 1 2 5 1 2 1
[121] 2 1 1 5 2 2 2 1 1 5 2 1 1 2 1 2 2 2 2 2 2 2 2 2 5 1 5 1 2 5 1 1 1 1 2 3 1 2 1 5 2
[161] 5 6 5 1 1 5 6 6 5 6 6 1 6 1 6 5 1 6 1 5 1 1 6 5 1 5 6 6 1 6 5 6 1 1 5 6 6 6 6 5
[201] 6 1 5 1 5 1 2 1 2 5 6 6 5 5 5 1 6 5 1 1 1 6 2 6 5 1 6 5 2 6 2 5 6 1 6 1 5 3 6 1
[241] 5 6 1 6 6 6 6 6 6

Within cluster sum of squares by cluster:
[1] 14.6009  9.1177 20.7894 20.0224 12.3169  8.4228
 (between_SS / total_SS =  82.8 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

Cluster	1	2	3	4	5	6	Total
Within SS	20.7894	8.4228	14.6009	20.0224	12.3169	9.1177	85.2701
Between SS %							82.8

APPENDIX 6: Cluster Display and Evaluation Source Code

Generating Models: K-1, K, K+1

```
# K4 selected using elbow method

# K - 1: K3 plot
ggplot(data=reviews_spt_rel_n_TG, aes(x=Spt_n_TG, y=Rel_n_TG, color=Cls_3_TG, shape=Cls_3_TG)) +
  geom_point(alpha=.8) + ggtitle("K3 Cluster Plot") +
  geom_point(data=centers3_TG, aes(x=Spt_n_TG, y=Rel_n_TG), size=3, stroke=2)


# K: K4 plot
ggplot(data=reviews_spt_rel_n_TG, aes(x=Spt_n_TG, y=Rel_n_TG, color=Cls_4_TG, shape=Cls_4_TG)) +
  geom_point(alpha=.8) + ggtitle("K4 Cluster Plot") +
  geom_point(data=centers4_TG, aes(x=Spt_n_TG, y=Rel_n_TG), size=3, stroke=2)


# K + 1: K5 plot
ggplot(data=reviews_spt_rel_n_TG, aes(x=Spt_n_TG, y=Rel_n_TG, color=Cls_5_TG, shape=Cls_5_TG)) +
  geom_point(alpha=.8) + ggtitle("K5 Cluster Plot") +
  geom_point(data=centers5_TG, aes(x=Spt_n_TG, y=Rel_n_TG), size=3, stroke=2)
```

Summary Table for Clusters

```
summary_reviews_TG <- reviews_data_TG %>%
  group_by(Ck4_TG) %>%
  summarise(Sptm_TG = mean(Spt_TG), Relm_TG = mean(Rel_TG), Natm_TG=mean(Nat_TG),
            Thrm_TG=mean(Thr_TG), Shpm_TG=mean(Shp_TG),Pccm_TG=mean(Pcc_TG),
            Agem_TG=mean(Age_TG), Incm_TG=mean(Inc_TG), Nbrm_TG=mean(Nbr_TG), Nobs_TG=n())
```

```
> summary_reviews_TG
# A tibble: 4 x 11
  Ck4_TG Sptm_TG Relm_TG Natm_TG Thrm_TG Shpm_TG Pccm_TG Agem_TG Incm_TG Nbrm_TG Nobs_TG
  <fct>     <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <int>
1 1       0.0236   0.212   0.157   0.185   0.219   0.204    37.5  53899.    699.      65
2 2       0.0114   0.223   0.162   0.189   0.210   0.205    38.1  45885.    485.      53
3 3       0.0251   0.151   0.260   0.203   0.160   0.200    37.0  44982     689.      72
4 4       0.0119   0.160   0.251   0.211   0.167   0.198    37    44694.    468.      59
```