Data Mining Fall 2023

Project 3

Lydia Malone, Rick Lattin, Armin Charkhkar

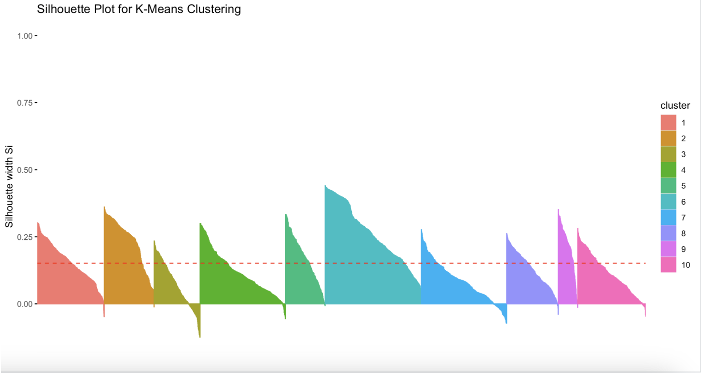
Introduction

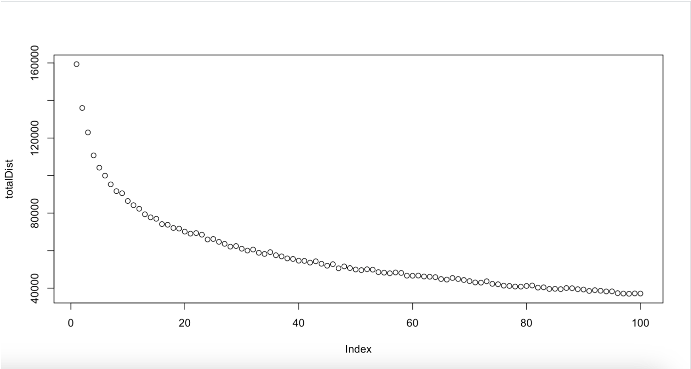
The given dataset for this project contains 5,456 instances of average user reviews regarding 24 different categories of European tourist sites. The ranking system ranges from 0-5, with zero being the worst, and 5 being the best. We were tasked with applying a variety of clustering algorithms, in order to determine any similarities and differences found between the reviews. The clustering algorithms we used in this project are: K-Means, Hierarchical, Gaussian Mixture, and DBScan.

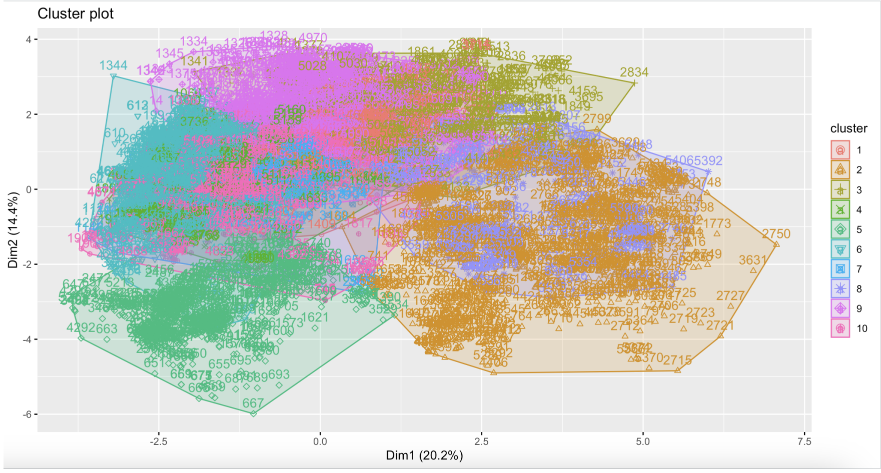
Data Pre-Processing

After loading in the dataset, we began the data pre-processing with removing the first column, which is the user ID column, as well as a blank column on the far right side of the dataset. Next we went ahead and removed any missing data with the na.omit function. The next step was to convert the “Category 11” column to contain all numerical values. The last step in the pre-processing stage was to remove all zeros in the dataset, as well as drop their corresponding indices.

K-Means



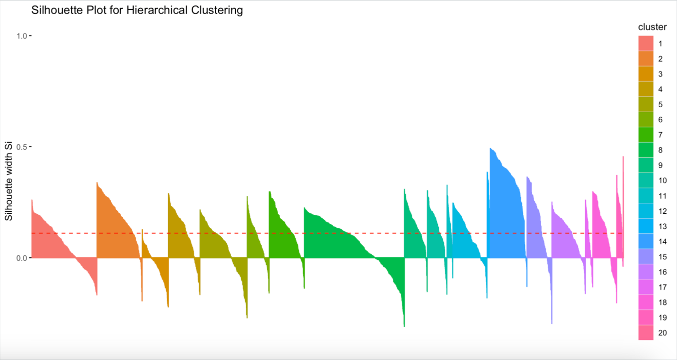


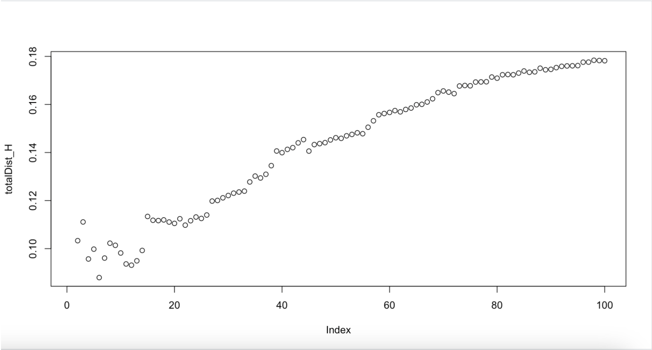


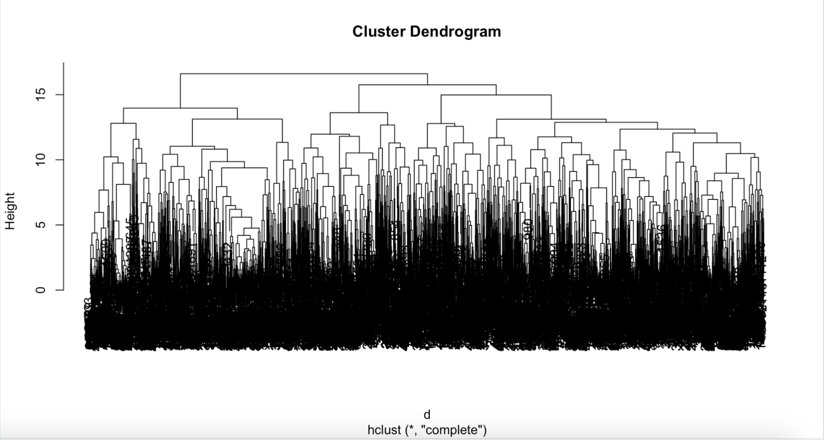
The silhouette plot for the K-means clustering that we performed seems to suggest that this clustering method worked very well on the dataset, as most of the data points on the plot are positive. This is shown clearly through the silhouette value which is represented by the red line across the graph. We ran the K-means function with a cluster size parameter of 4 first to test that the function worked properly before trying it with cluster values from 2 to 100 and plotted the results. The final K-means function to run was given a value of 10 to represent the number of clusters, as we determined that would be the best option from using the elbow method on the plot of the attempted cluster sizes shown above. We also used the fviz_cluster function from the factoextra package to visualize the clusters so we could visually verify the results.

`

Hierarchical

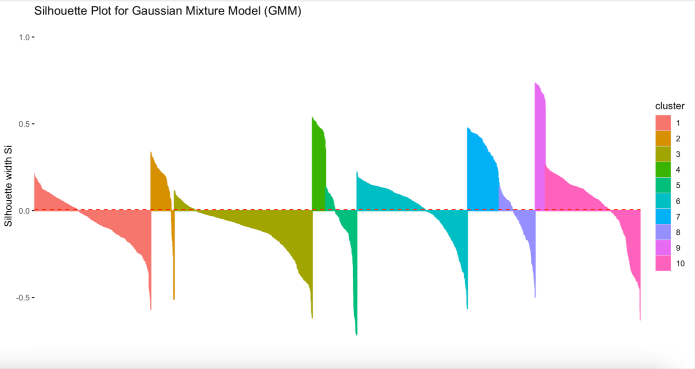


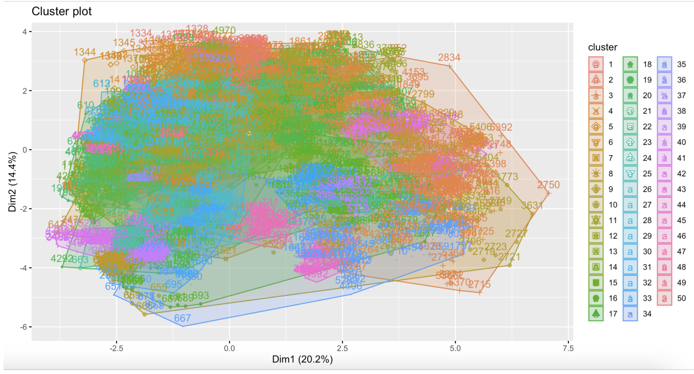


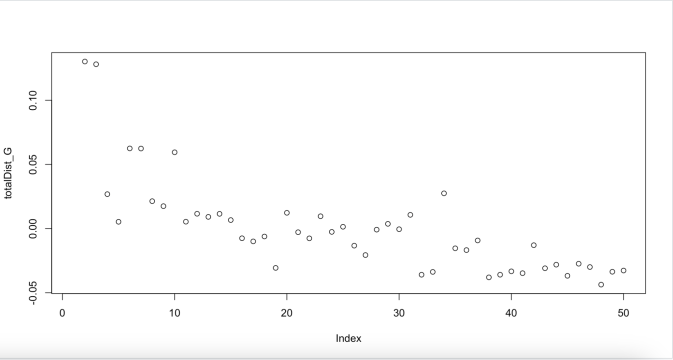


The Hierarchical method of clustering also seemed to be very effective from the silhouette plot, although not as effective as K-means. Most of the data points are positive along with the silhouette value, but unlike the K-means clustering we did not have as clear of a metric to determine which number of clusters would be best. We plotted different uses of the cutree function with parameters from 2 to 100 to represent the number of clusters and used the silhouette values of each to compare those. Unfortunately the plot of these values seemed to increase semiconsistently, so there was not a clear choice to make. As a result we decided to choose 20 clusters for the cutree function as it represented the beginning of the gradually slowing upward trend. We also used a dendrogram to visualize the original hierarchical tree that was created with the hclust function.

Gaussian Mixture

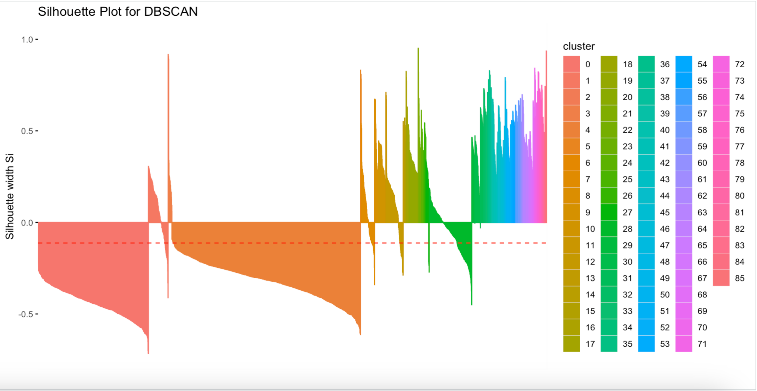






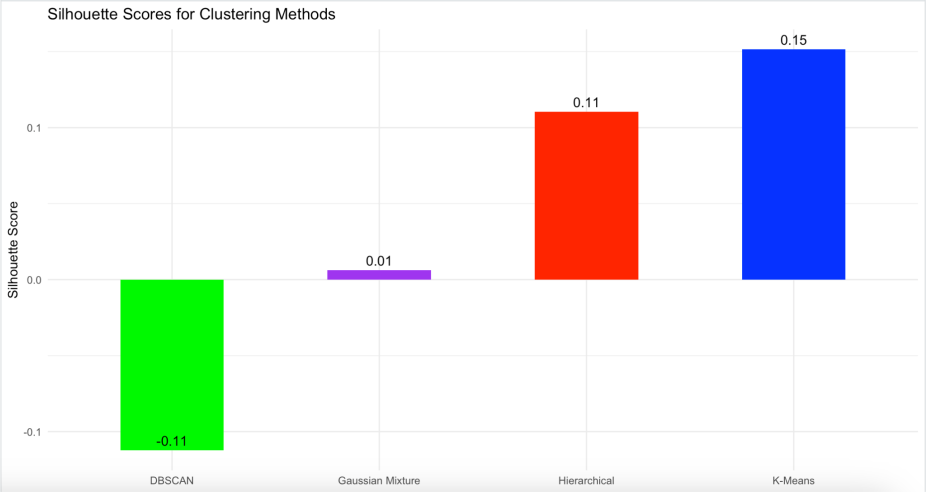
Gaussian Mixture was an interesting clustering algorithm to test with. The first plot shows a pretty split representation of positive and negative silhouette scores, and the average silhouette score is just barely positive. The second plot shows that the Gaussian Mixture does not give the clearest dot plot curve, yet there is still a general trend that can be made out, allowing us to use the elbow method. Thus, we decided to use the value of 10 for the number of clusters used in the Gaussian Mixture Model. The third visualization utilizes the fviz_cluster function to give us a different perspective of the actual data clustering. What is unique about the Gaussian Mixture Model is that despite the loose dot plot curve and barely positive average silhouette score, it performed relatively well in terms of other metrics. For instance, when using the Calinski Index, Gaussian ranked the second highest. We also used the Davies-Bouldin Index, which desires a low score and means that the clusters are well separated. Using this index, Gaussian Mixture Model ranked second again. Thus, using metrics other than the average silhouette score can help one realize that a clustering algorithm can do well with data.

DBScan



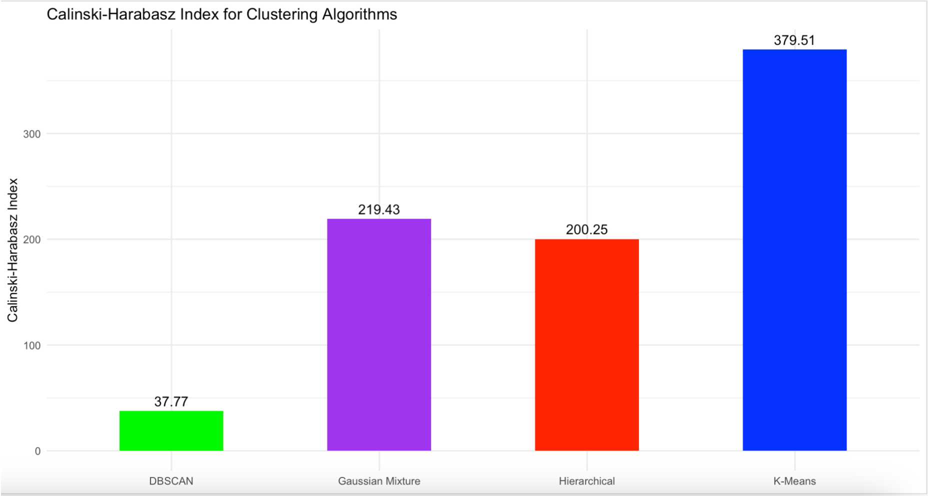
For DBScan, we can observe that there is a high prevalence of negative silhouette scores. Most notably in the first cluster in red, which also showcases the high number of outliers present in the data. The parameters used to achieve this visualization were an eps value of 3.0 and a MinPts value of 5. While this plot is not ideal, it is the most representative of the data, using DBScan. We tweaked the parameters of the dbscan object quite frequently, in order to see if we could achieve different amounts of clusters, as well as a positive average silhouette score. During this period of trial and error, we were able to come across a positive average silhouette score plot, but decided not to use it. The reasoning behind this was because it was not fully representative of the data and had improper clustering. Despite the positive score, the plot only had three clusters, and with such a large dataset, it is almost impossible to take anything away with only three clusters. Thus, we decided to use this plot because of its more accurate depiction of the data, which involves 86 clusters.

Conclusion



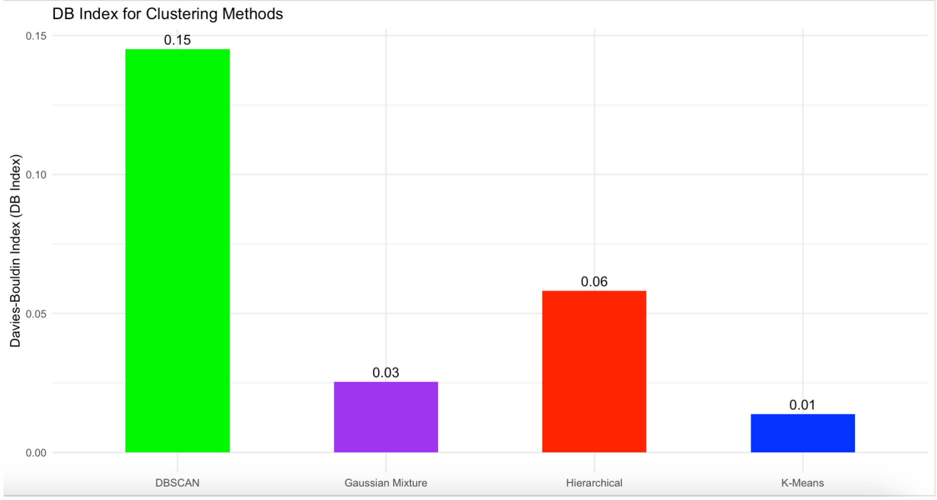
Silhouette Scores

The bar plot above shows the average scores of the four clustering algorithms used in this project. The silhouette scores determine two things when it comes to the clustering of the data. The silhouette scores range from anywhere between -1 and +1. The first thing that the silhouette scores considers is how close together the data points within a cluster are with one another. The second consideration is how far away points in different clusters are from each other. The higher a positive score is, the better, and vice versa for the negative score. A score of zero implies data points are close between two different clusters. As we can see from the plot, DBScan did the worst, Gaussian Mixture is in the middle, hierarchical did fairly well, and K-means performed the best.



Calinski Index(Exceptional Work)

The Calinski Index measures the ratio of between-cluster variance to within-cluster variance. A higher Calinski Index indicates that the data points within clusters are tightly packed and well separated from each other, while the clusters themselves are distinct from one another. In other words, a higher CH Index suggests that the clustering solution is more optimal and that the data is naturally organized into distinct clusters.



Davies-Bouldin Index(Exceptional Work)

The Davies-Bouldin Index measures the average similarity between each cluster and its most similar neighboring cluster. A lower Davies-Bouldin Index indicates that the clusters are well-separated and distinct, with minimal overlap. In essence, it quantifies the extent of dissimilarity between clusters, and a smaller value suggests a more optimal clustering solution. Therefore, a low Davies-Bouldin Index signifies that the data is well structured into separate and distinct clusters, contributing to the effectiveness of the clustering algorithm and providing valuable insights into the organization of the data.

Overall, it seems apparent that through the comparisons of the four clustering methods using the Silhouette Scores, the Calinski Index and the Davies-Bouldin Index that the K-means clustering seems to be most effective on this particular dataset. The K-means method had the highest score on both the Silhouette and the Calinski comparisons, in which a higher score is desirable, and the lowest score on the Davies-Bouldin comparison, in which the lowest score is desirable. On the other hand, the DBScan method performed the worst for this dataset, as it scored the lowest on the Silhouette and the Calinski comparisons and the lowest score on the Davies-Bouldin comparison. The other two clustering methods were comparable in performance throughout the methods, although notably, despite having a relatively low Silhouette Score, the Gaussian Mixture method of clustering scored the second best in both the Calinski and the Davies-Bouldin Indexes, displaying how the Silhouette score is does not always accurately represent the overall success of a method of clustering with a given dataset.