Movie Clustering for Recommender Systems

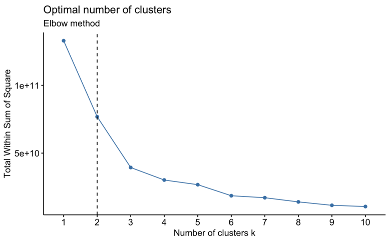
Our project aims to address the issue of overwhelming choices faced by users when selecting a movie to watch. We propose implementing clustering on a movie dataset to create a recommendation system that suggests relevant movies to users, providing a more personalized and convenient experience. This movie recommendation system has the potential to increase user engagement and satisfaction, leading to higher revenue for movie platforms.

To analyze patterns and relationships in the data without the need for labeled training data, unsupervised learning algorithms are used in movie recommendation systems. We specifically utilized k-means clustering and hierarchical clustering algorithms. These techniques allow us to cluster movies based on their attributes and similarity.

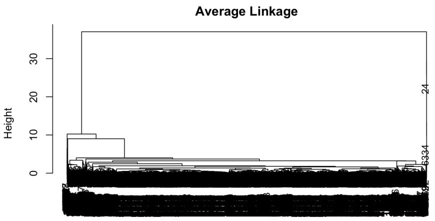
Below is the followed approach:

Data Cleaning and Data Analysis: The data has 8 columns and 9463 records. It has id, title, release date, overview, popularity, vote average, vote count and video columns. We extracted year and month from release date. We considered popularity, vote average, vote count, and release year and month features for clustering. Records with null values were omitted as they were minimal in our dataset. Our analysis revealed that year by year, movie popularity increases significantly, and there is a higher number of movie releases. We also observed a strong correlation between vote average and vote count.

KMeans Clustering: Elbow method is used to find the optimal number of clusters. From the graph, although the suggested optimal number of clusters are 2, we considered 3 clusters as the elbow is at 3. After clustering, we observed that cluster 3 has most number of movies followed by cluster 1. When evaluated with between cluster sum of squares, we got 14410.69. Analyzing the clusters, we observed that even though the vote count and vote average for cluster 1 and cluster 3 are high, the popularity is low. Cluster 1 has recently released movies.



Hierarchical Clustering: When tried average, complete and single linkage, average linkage gave better results. From the dendrogram, we selected 4 clusters. After clustering, cluster 1 has 90% of the movies. We tried with different number of clusters like 2, 3, 6, 8 but the results remained the same. As a result of the poor clustering results, after analyzing the clustered data, we were unable to find any useful insights.



Conclusion: KMeans clustering performed better when compared to hierarchical clustering and the results of KMeans can be used for movie recommendation systems. In summary, our project focuses on creating a movie recommendation system using clustering techniques. By incorporating user behavior and movie attributes, we aim to provide personalized recommendations to users and enhance their movie-watching experience.