Analysis of the MovieLens

CIT646: Data Mining Final Project

Team Members:

Salma Hisham Mohamed, 231000533

Shaden Abdelrahman Mohamed _ 231002276

Norhan Mohamed Swar, 231000486

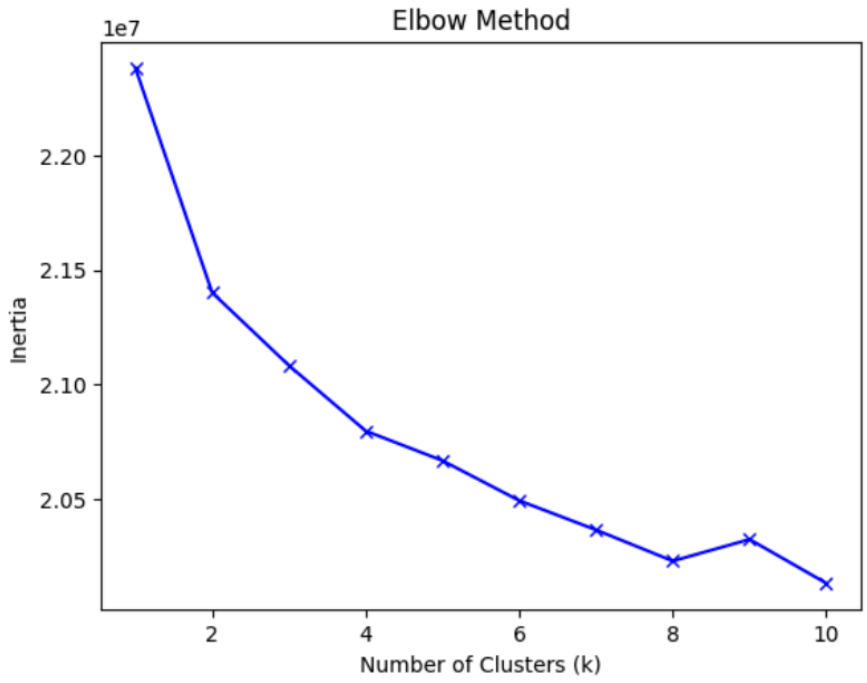
Supervised by:

Dr.Tamer Arafa

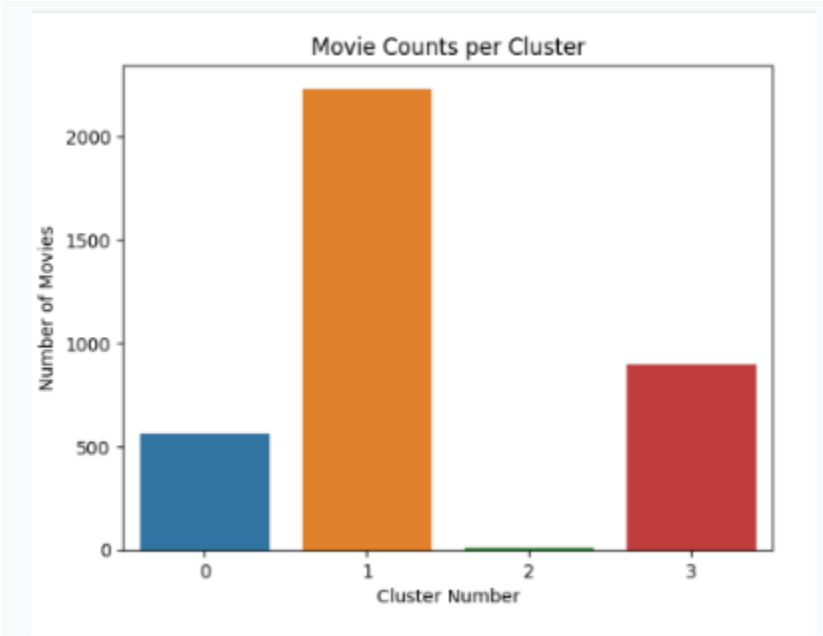
6- Clustering

Methodology:

1. **Data Preparation:**
    - **Load the dataset containing user ratings for movies.**
    - **Normalize the ratings using a suitable method (e.g., min-max scaling) to ensure features have a consistent scale.**
    - **Extract relevant features (user IDs and movie ratings) and create a matrix representation.**
2. **K-Means Clustering:**
    - **Choose a suitable number of clusters (k) using techniques like the elbow method or silhouette analysis.**
    - **Initialize cluster centroids randomly or using strategies like K-means++.**
    - **Iterate until convergence:**
        - **Assign each user to the cluster with the closest centroid (using Euclidean distance or other appropriate metrics).**
        - **Recalculate centroids as the mean of all users within each cluster.**



Results:


Movie Counts per Cluster
Number of Movies
2000
1500
1000
500
0
0
1
2
3
Cluster Number

7.community Detection

Methodology:

1.getting subset of the data filtered on the top 10 ranked movies.

2.getting the cosine similarity between users according to their ratings.

- The higher the cosine similarity, the more similar the preferences of the users.

4. Create a user similarity graph.

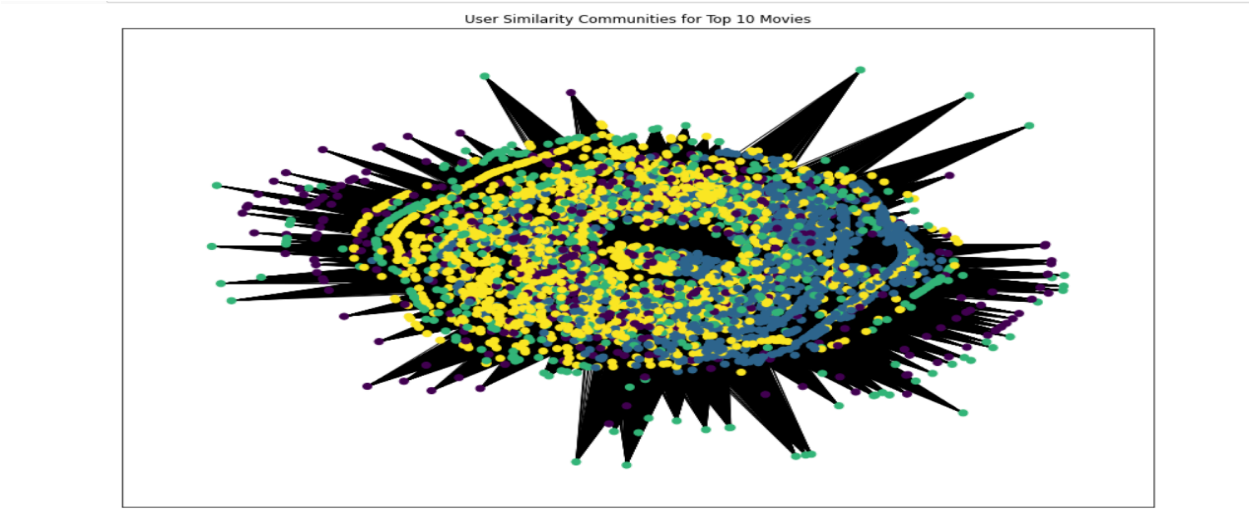
5.Apply the Louvain method for community detection.

- communities are used to group users based on their similarity in movie preferences

6. Visualize and analyze the communities.

Results:

Four communities,each of them contains users who exhibit similar patterns in their movie preferences.



8. Recommendation System

Methodology:

1. Collaborative Filtering with Matrix Factorization:
    - Choose a matrix factorization algorithm SVD.
    - Decompose the user-item rating matrix into two lower-dimensional matrices:
        - User factor matrix: Represents user preferences for latent features.
        - Item factor matrix: Represents movie associations with those latent features.
    - Optimize the model parameters to minimize reconstruction error.
2. Model Training:
    - Train the model on the training set to learn user and item latent factors.
3. Prediction:
    - For each user-item pair in the test set:
        - Multiply corresponding user and item latent factors.
        - Predict the rating based on the product.
4. Evaluation:
    - Calculate Root Mean Squared Error (RMSE) between predicted and actual ratings on the test set.
    - RMSE = √(Σ(predicted_rating - actual_rating)^2 / (number of ratings))
    - Lower RMSE indicates better model performance.
    -