Movie Recommendation System for Netflix

SILVIA MANONO
AMOS KIPNGETICH
ANGELA MAINA
CHARLES NDEGWA
SANDRA KOECH
GLORIA TISNANGA
ALEX MININGWA

Introduction

This project aims to improve Netflix's recommendation system to solve the issue of users endlessly scrolling to find content they enjoy.

By enhancing the recommendation engine, Netflix seeks to provide more personalized and relevant suggestions, increasing user engagement and viewing time, while reducing decision fatigue.

Business and data Understanding

Netflix, aims to provide personalized content to ensure user engagement and satisfaction. However, with thousands of options available, users often face difficulty in finding content that matches their preferences, leading to decision fatigue and lower engagement. This results in users endlessly scrolling without finding something to watch quickly.

To address this, Netflix is focused on enhancing its recommendation engine by suggesting films similar to those users have enjoyed, based on the content and genre of the movies. The goal is to build a content-based recommendation system that provides relevant movie suggestions, helping users discover new content easily, subscribe and thus more sales to the company

Data description, Movie titles & Genres

User IDs, Movie IDs, Ratings This data allows for both content-based and collaborative filtering approaches.

EDA Highlights

Exploratory data analysis revealed key patterns in user ratings, popular genres, and user-movie interactions. Visualization techniques were used to identify trends and correlations in the data.

Modeling Techniques

Content-Based Filtering: to recommend movies similar to what users have liked using cosine similarity based on genres.

Collaborative Filtering:
KNNBasic: Predicts ratings based on similar users.
KNNWithMeans: Adjusts for user biases by incorporating mean ratings.
SVD: Matrix factorization to uncover hidden patterns in user-item interactions.

Hybrid Model: Combines content-based and collaborative filtering techniques for enhanced performance.

Objectives

1. **Develop a Personalized Movie Recommendation System**
   Build a recommendation model using collaborative filtering (e.g., matrix factorization or deep learning techniques like neural collaborative filtering) to predict user ratings for movies. Recommend the top 5 most relevant movies for each user, enhancing the personalization experience on the platform.

2. **Address Cold Start for New Users**
   Tackle the cold start issue by implementing content-based filtering and recommending movies based on user preferences (genres, actors).

3. **Enhance System Precision and User Feedback Integration**
   Improve recommendation accuracy and relevance through a hybrid approach that combines collaborative filtering and content-based filtering.

Data Description

1. movies.csv

movieId: Unique identifier for each movie.title: Movie titles.genre: Movie genres.

2. ratings.csv

userId: Unique identifier for each user.movieId: Unique identifier for each movie.rating: User rating from 0 (lowest) to 5 (highest).timestamp: Seconds since January 1, 1970 (UTC).

3. tags.csv

userId: Unique identifier for each user.movieId: Unique identifier for each movie.tag: User-determined phrase.timestamp: Seconds since January 1, 1970 (UTC).

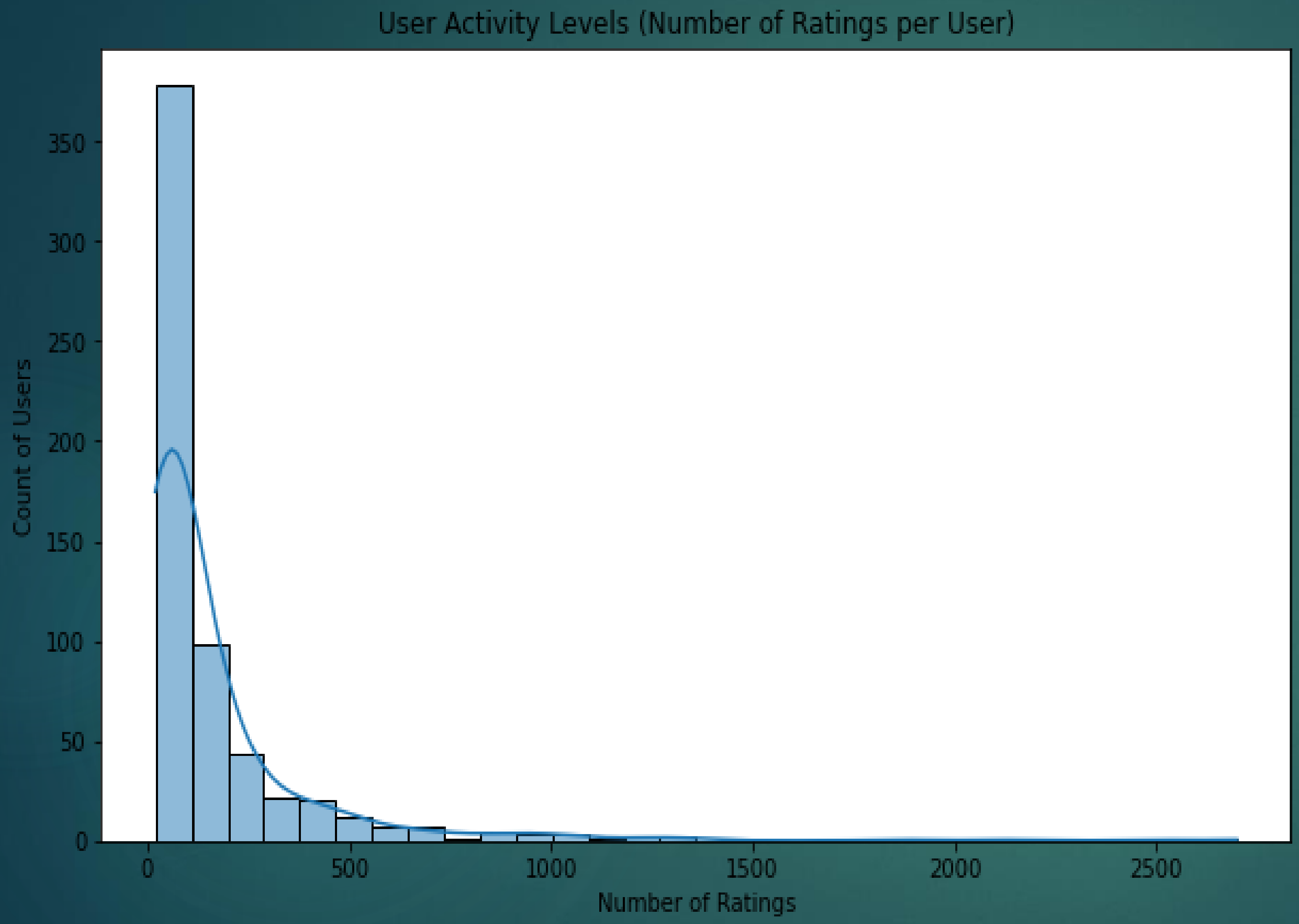
4. links.csv

movieId: Identifier for movies used by MovieLens.

imdbId: Identifier for movies used by IMDb.

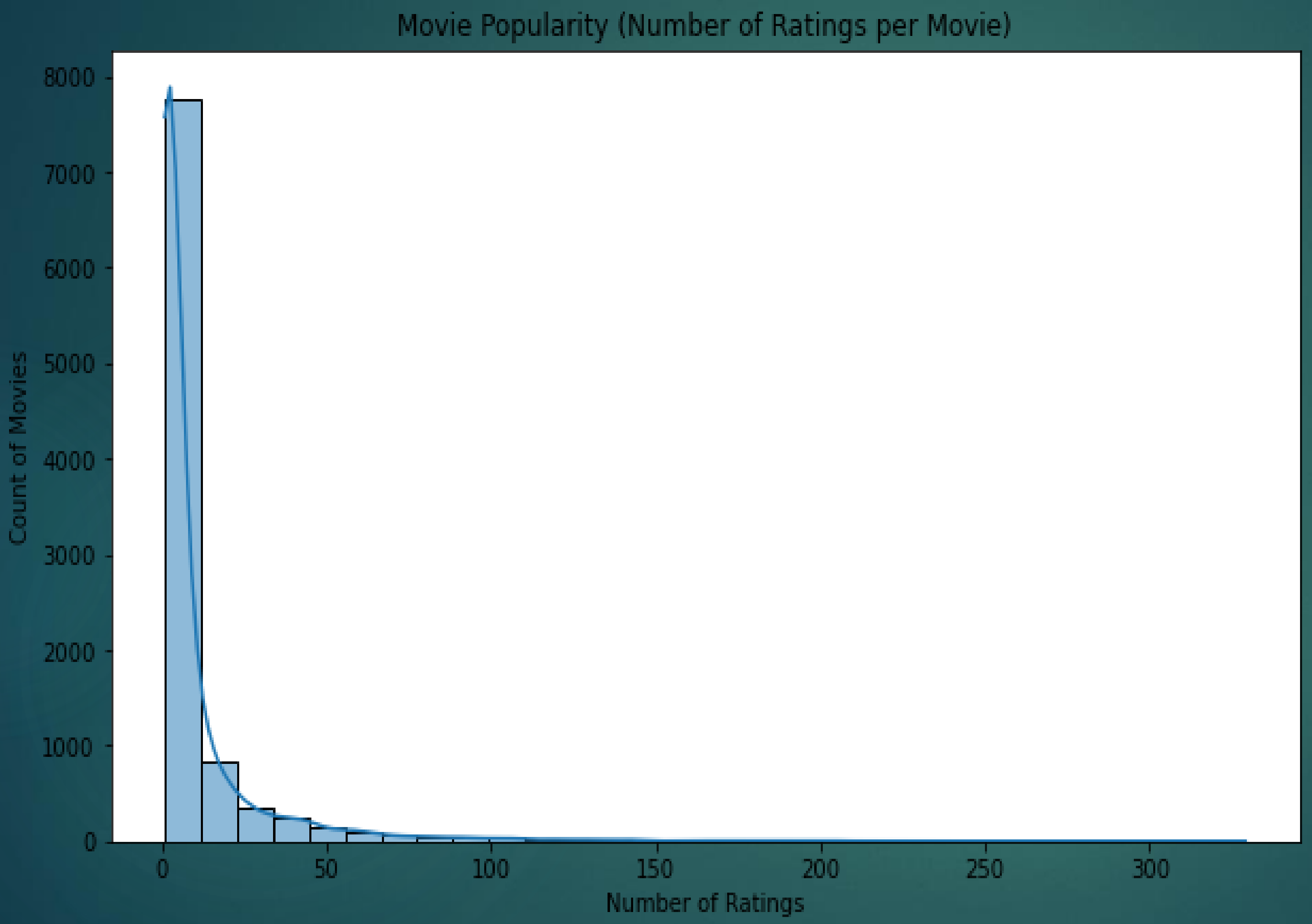
tmdbId: Identifier for movies used by TMDb.

Exploratory Data Analsysis



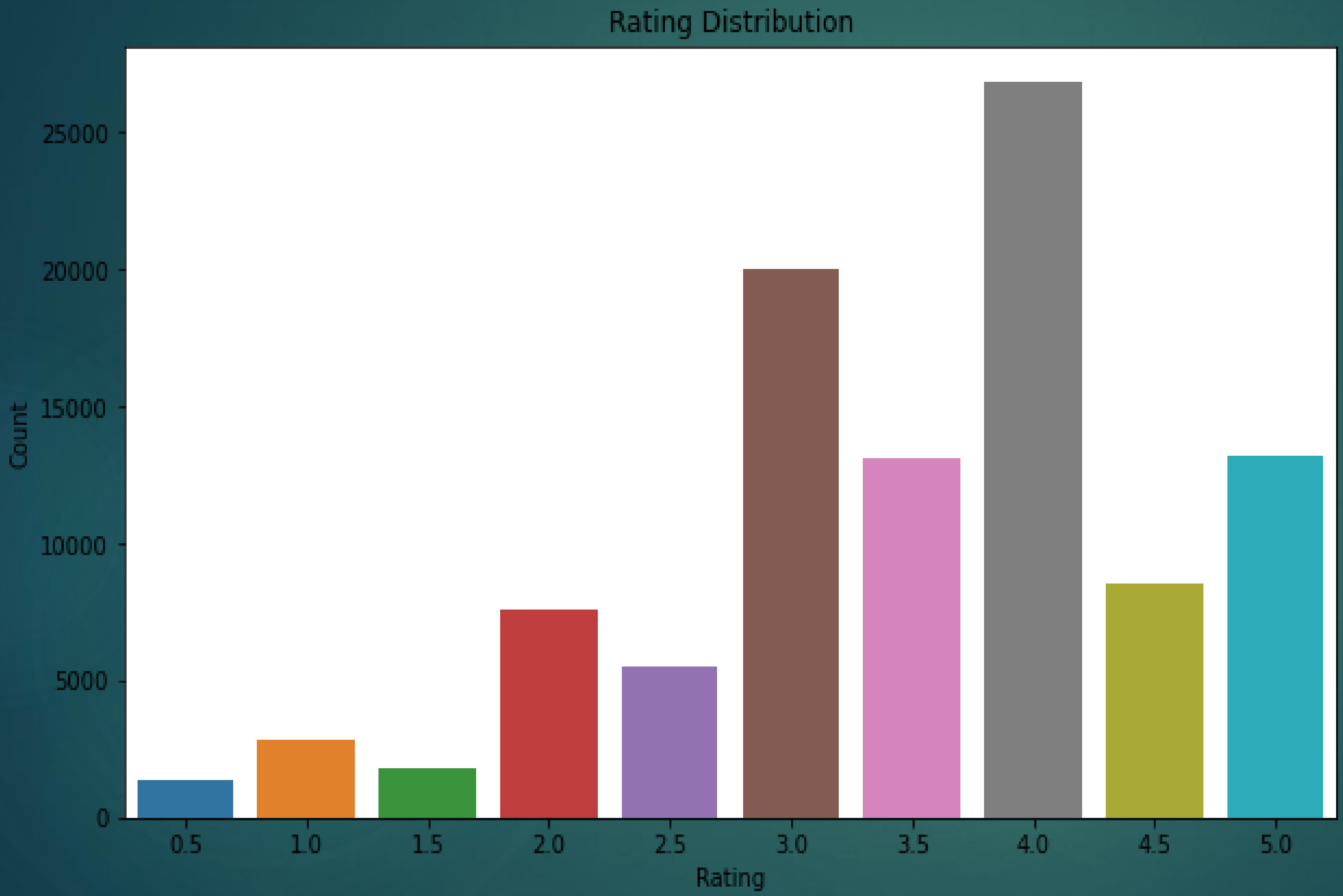
Majority of users have movie ratings below 500

EDA



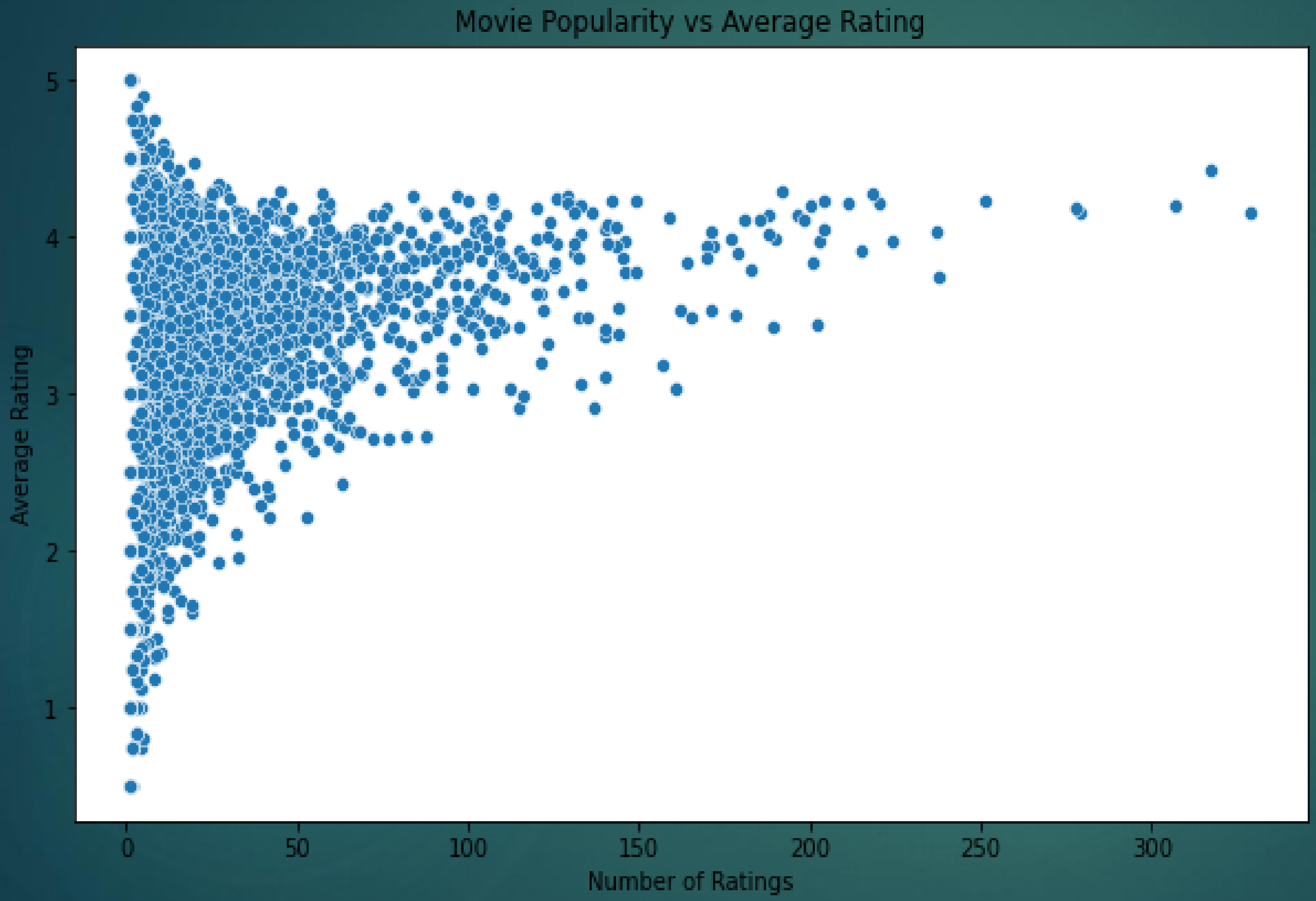
Most movies constitute
Average ratings below 50

EDA



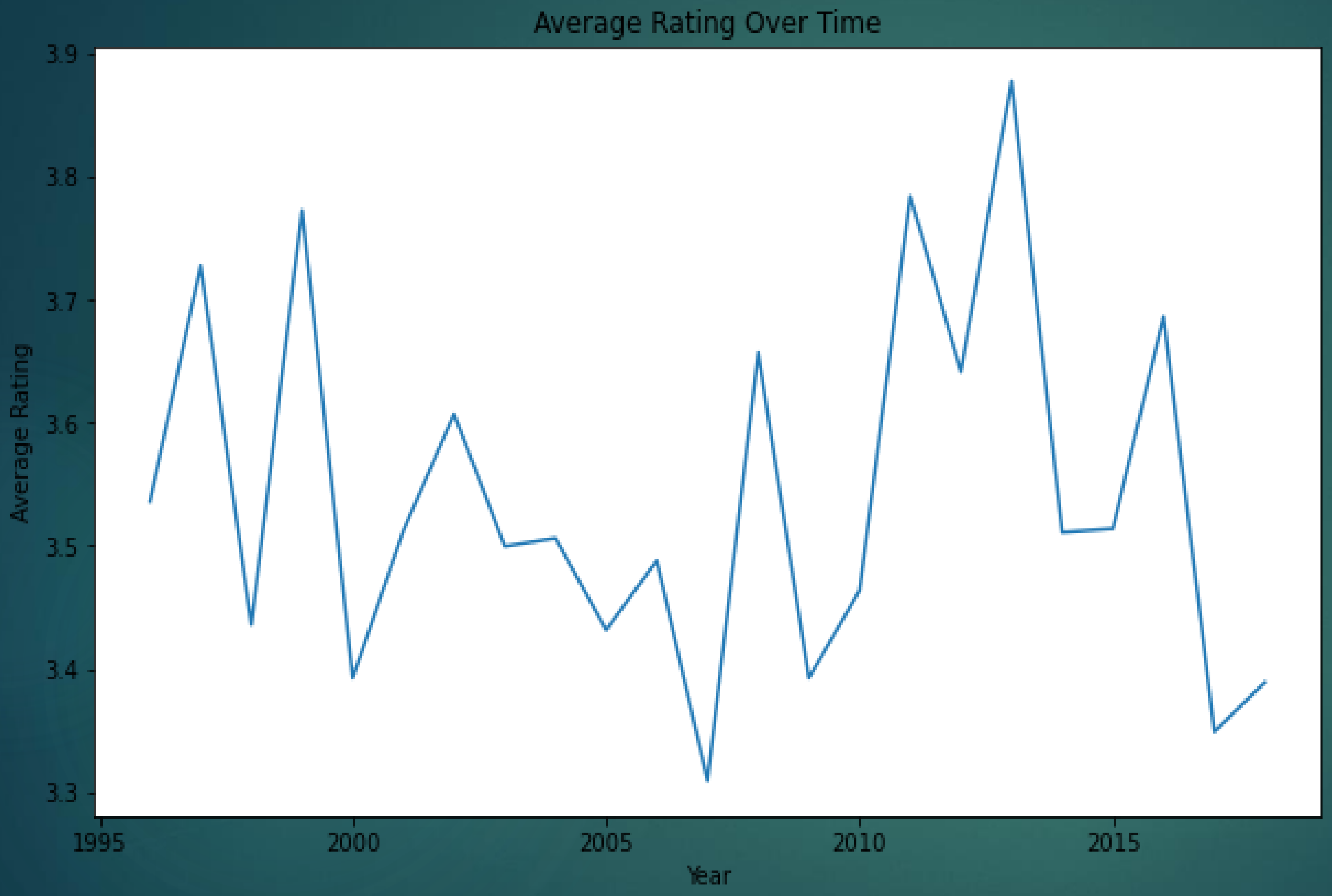
Majority of movies by NETFLIX attracted ratings above 3.0

EDA



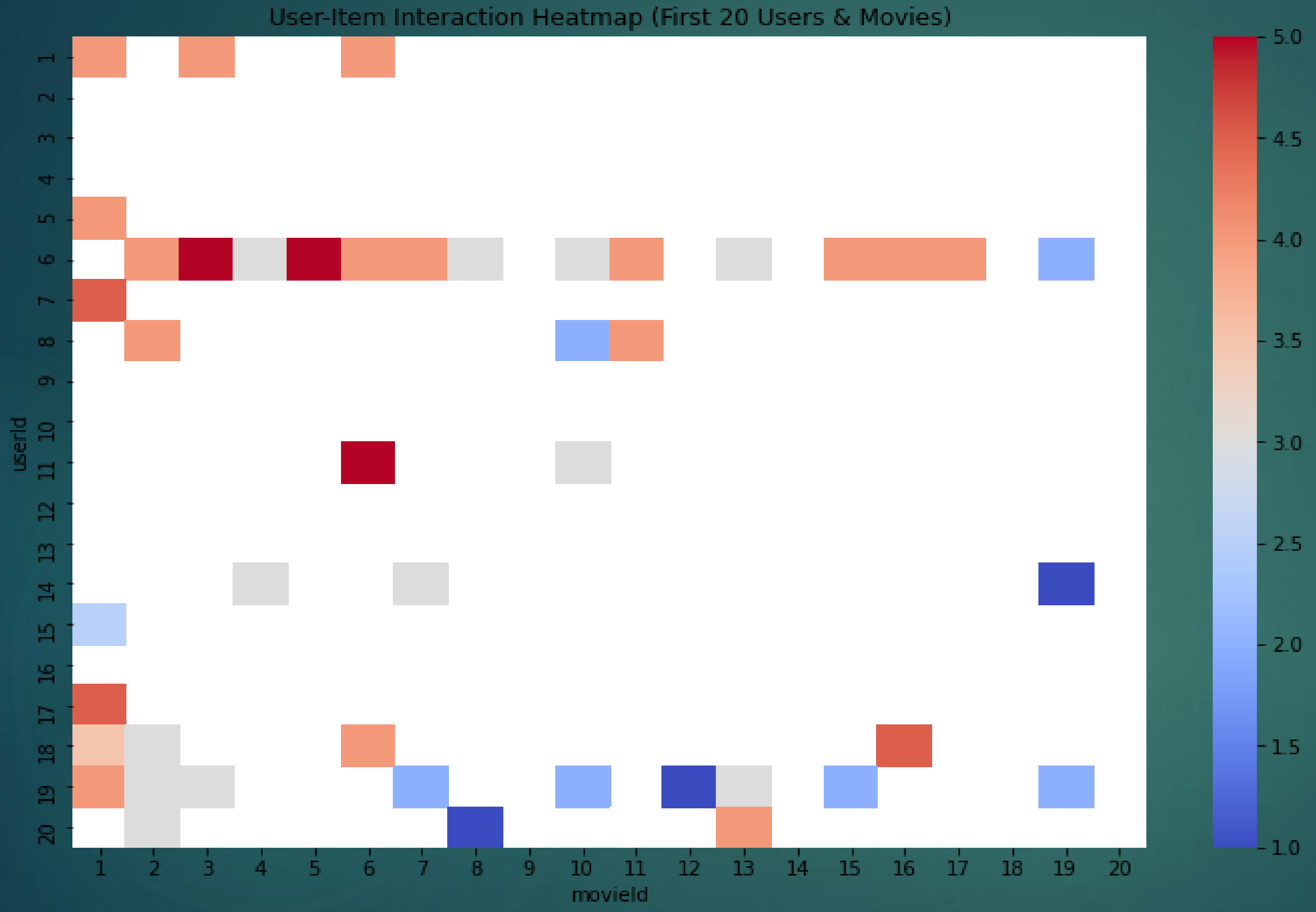
The popularity of majority of movies by NETFLIX fall below 150 ratings.

EDA



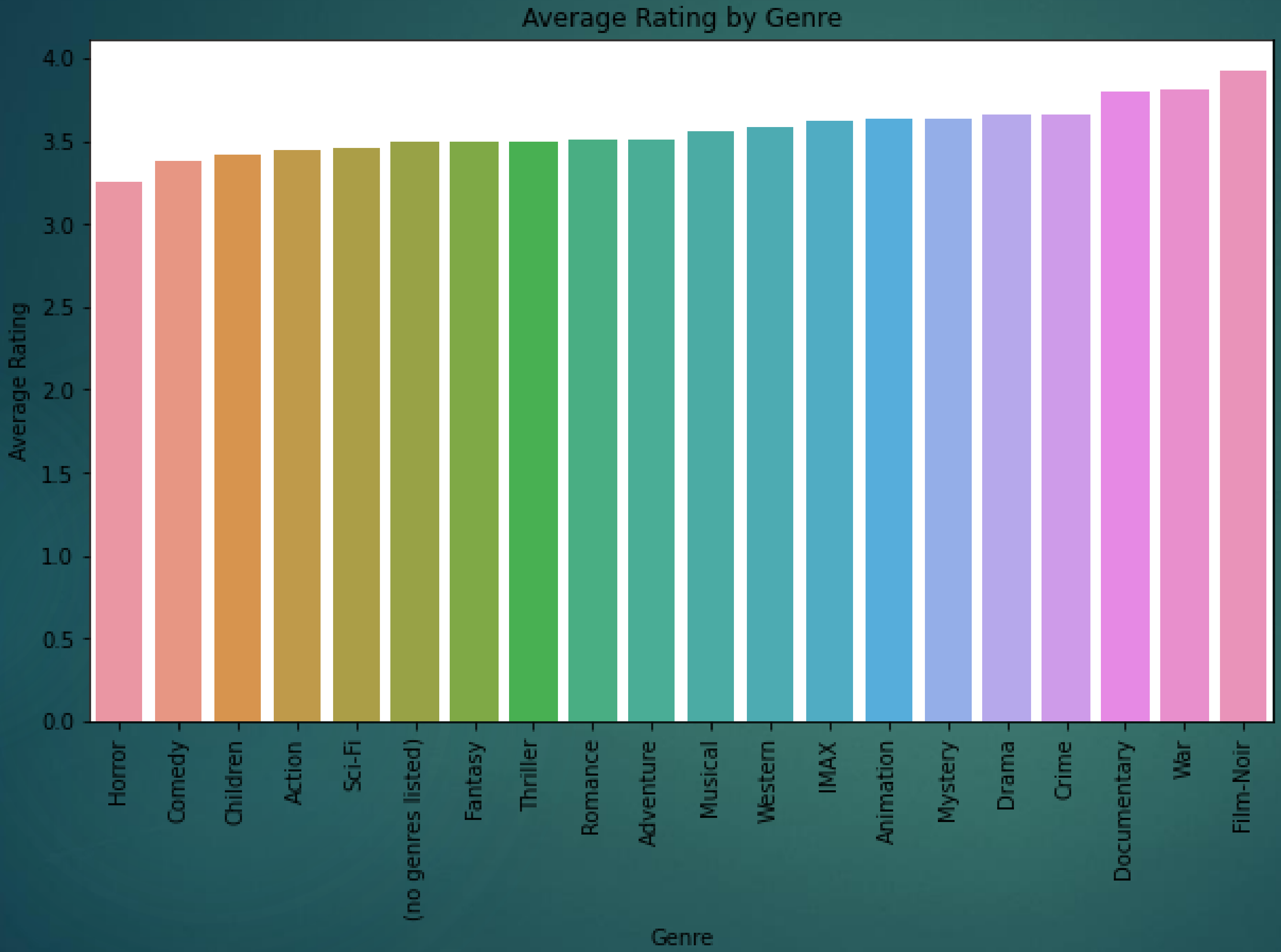
Average rating of movie Collections has been fluctuating overtime

EDA



Most movie ratings at NETFLIX range between 2.5-4.0.
This imply the need for more user engagement for better Movie ratings

EDA

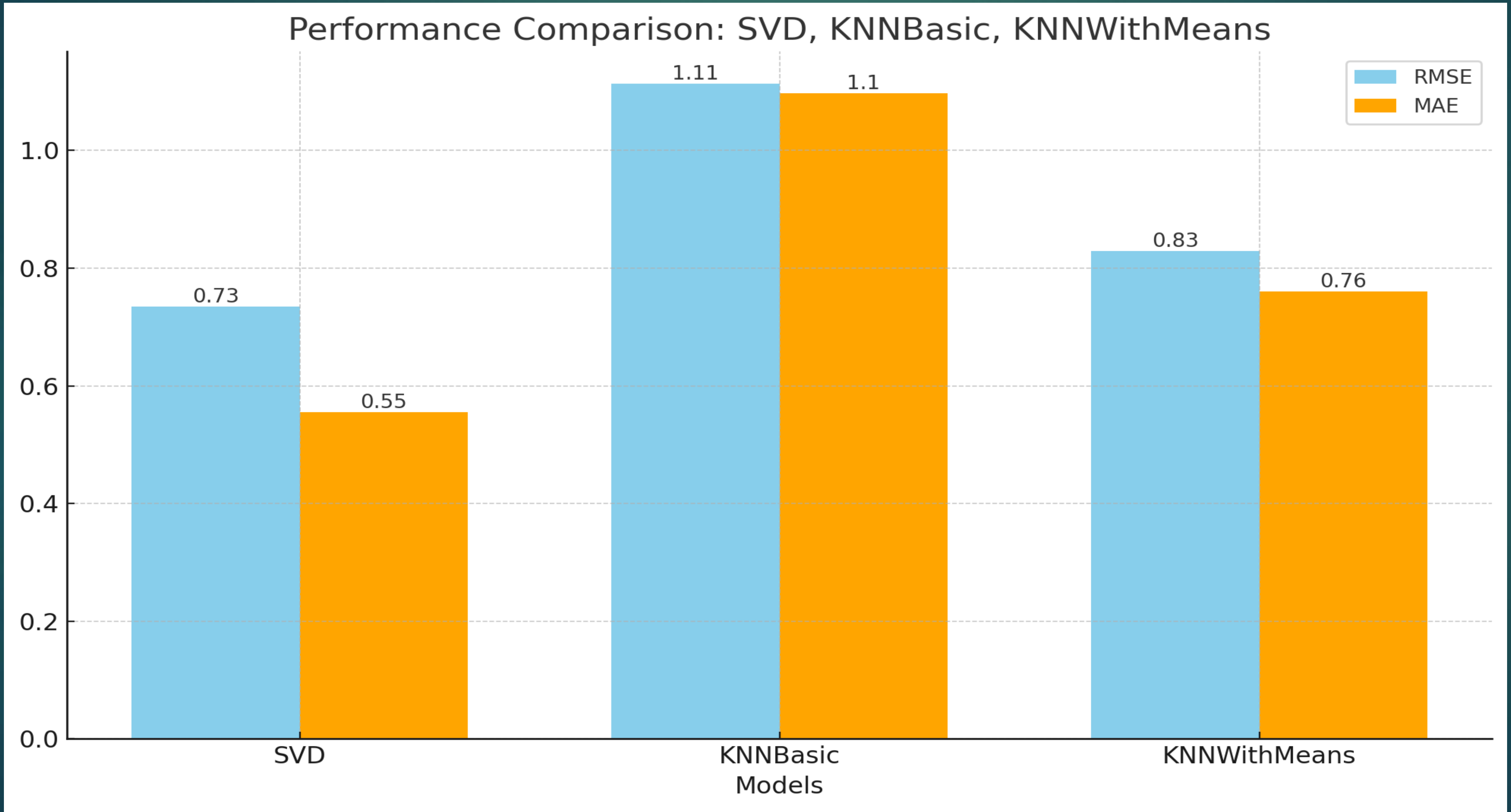


Average movie ratings
For most of the genre are
Slightly above 3.0

Modelling

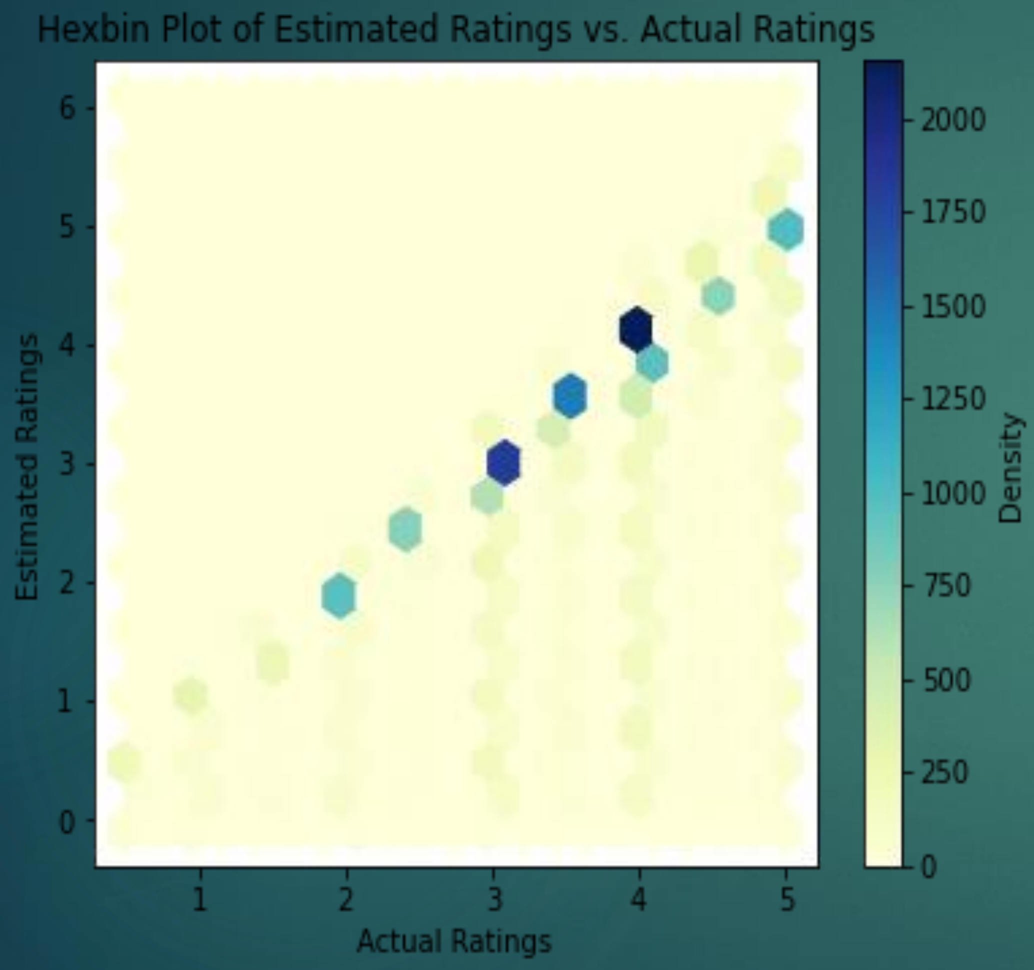
- Models Evaluated:
- SVD (Singular Value Decomposition)Captures latent factors in user-item interactions.Best for handling large, sparse datasets.RMSE: 0.7342, MAE: 0.5546Conclusion: Lowest errors, most accurate predictions.
- KNNBasic (K-Nearest Neighbors Basic)Finds similar users and recommends based on their preferences.Does not adjust for user bias.RMSE: 1.1131, MAE: 1.0969Conclusion: Highest errors, least accurate.
- KNNWithMeansRefined KNN that accounts for user bias (average ratings).RMSE: 0.8294, MAE: 0.7602Conclusion: Better than KNNBasic but not as good as SVD.Overall Results:SVD outperforms both KNN models with the lowest error rates.KNNWithMeans improves on KNNBasic by incorporating user bias.

Modelling



Performance Comparison: SVD, KNNBasic, KNNWithMeans
RMSE
MAE
1.0
0.8
0.6
0.4
0.2
0.0
0.73
0.55
1.11
1.1
0.83
0.76
SVD
KNNBasic
KNNWithMeans
Models

Model Validation and Deployment



RMSE of Deployed Model:

1.17Interpretation: The model's predicted ratings deviate from actual ratings by about 1.17 units on average.Rating Scale (1 to 5): This is a strong result, showing predictions are relatively accurate.Improvement: Performs much better than the hybrid model tested earlier (RMSE: 2.51).

Validation Results:Strong Positive Correlation:The plot shows a clear alignment between predicted and actual ratings, confirming the model's accuracy after deployment.

Conclusions and Recommendations

- Conclusion:

The evaluation of the movie recommendation system shows promising results with an RMSE of 1.1689, indicating predictions closely align with actual ratings.

This marks a significant improvement over the hybrid model's RMSE of 2.5133, reflecting a refined approach that effectively captures user preferences.

Hexbin plot analysis reveals a strong positive correlation between actual and estimated ratings, underscoring the model's reliability.

- Recommendations:

Refine Collaborative Filtering Approach:Experiment with different matrix factorization techniques and tune hyperparameters to further reduce RMSE.

Leverage Content-Based Filtering:Implement techniques for new users to alleviate the cold start problem using movie features like genres and directors.

User Feedback Mechanism:Introduce a feedback loop for users to rate recommended movies, enabling continual model refinement.

Evaluate Additional Metrics:Use metrics like Precision@K, Recall, and F1 Score for a comprehensive assessment of recommendation effectiveness.Continuous Learning:

Explore advanced techniques such as reinforcement learning or deep learning to enhance the model's adaptability based on user interactions.

Way forward

1. **Conduct A/B Testing**: Implement A/B testing for different recommendation strategies to determine the most effective model. This will provide insights into user engagement and satisfaction levels.

2. **Regular Model Updates**: Schedule regular updates to the model, incorporating new data and user feedback to maintain the system's relevance and accuracy.

3. **Expand Data Sources**: Consider integrating external data sources (e.g., social media trends, movie reviews) to enrich the recommendation process and improve user engagement.

4. **User Segmentation**: Analyze user data to create segments based on viewing habits and preferences. Tailor recommendations for different segments to enhance personalization.

5. **User Interface Enhancement**: Improve the user interface to make the recommendations more visible and accessible. Clear communication of why specific movies are recommended can enhance user trust and satisfaction.