# Capstone Project –

# Netflix Recommendation System

-by Mayank Vyas



Table of Contents

1. Problem Statement
2. Project Objective
3. Data Description
4. Data Pre-processing Steps and Inspiration
5. Choosing the Algorithm for the Project
6. Motivation and Reasons for Choosing the Algorithm
7. Assumptions
8. Model Evaluation and Techniques
9. Inferences from the Same
10. Future Possibilities of the Project
11. Conclusion
12. References

# Problem Statement

Customer Behaviour and its prediction lies at the core of every Business Model. From Stock Exchange, e-Commerce and Automobile to even Presidential Elections, predictions serve a great purpose. Most of these predictions are based on the data available about a person’s activity either online or in-person.

Recommendation Engines are the much-needed manifestations of the desired Predictability of User Activity. Recommendation Engines move one step further and not only give information but put forth strategies to further increase user’s interaction with the platform.

In today’s world OTT platform and Streaming Services have taken up a big chunk in the Retail and Entertainment industry. Organizations like Netflix, Amazon etc. analyze User Activity Pattern’s and suggest products that better suit the user needs and choices.

For the purpose of this Project, we will be creating one such Recommendation Engine from the ground-up, where every single user, based on their area of interest and ratings, would be recommended a list of movies that are best suited for them.

# Project Objective

Our project is to build a recommender system for users based on the interactions and ratings provided for the content.

Specifically, we will try to recommend movies a user has not seen that they may enjoy, based on previous movies they saw and rated.

We plan to use SVD algorithm to create this recommendation system.

# Data Description

The dataset available is Copy\_of\_movie\_titiles.csv and Copy\_of\_combined\_data.csv

Data description, various insights from the data.

The Copy\_of\_movie\_title.csv contains 17770 rows and 3 columns.

1. Movie\_ID – Contains the separate keys for movies
2. Year – Year on which movie got released
3. Name – Movie Name

The Copy\_of\_combined\_data.csv contains 24058263 rows and 3 columns

1. Cust\_ID: Contains the separate keys for customer
2. Rating: – A section contains the user ratings for all the movies

# Data Preprocessing Steps and Inspiration

The preprocessing of the data included the following steps:

1. **Checked for Null Values:** Identify any missing values in the dataset to ensure data completeness.
2. **Checked for Duplicates:** Detect and remove any duplicate entries to maintain data integrity.
3. **Checked for Data Types:** Verify the data types of each column to ensure they are appropriate for analysis.
4. **Converted Cust\_ID Column Data Type:** Changed the data type of the Cust\_ID column from an object to an int format for accurate analysis.

These steps are crucial for preparing the dataset for further analysis and modelling.

The main inspiration for this study is to leverage the power of data analytics to enhance best movie recommendation to the users.

# Choosing the Algorithm for the Project

Description for the Time Series algorithm for the project.

I have chosen the below algorithm for this project for the following reasons:

1. **Recommendation System – Singular Value Decomposition** **(SVD)**

Recommender systems are algorithms designed to suggest relevant items to users. These systems are used in various domains such as e-commerce, streaming services, and social media. They enhance user experience by filtering vast amounts of information to deliver personalized content.

SVD is a linear algebra technique that breaks down a matrix into smaller matrices. It's a popular method for collaborative filtering, which is used to make recommendations.

Step by step:

1. **Install and import libraries**: - We have used a special library called **scikit-surprise.**
2. **Train the SVD model: -** We have performed the recommendation for one user
3. E**valuate the Model:** - We evaluate the performance of the trained SVD model using metrics such as RMSE (Root Mean Squared Error) and MAE (Mean Absolute Error).
4. **Make Predictions: -** We can predict the rating a specific user would give to a specific item.

# Assumptions

The following assumptions were made in order to create the model for Walmart Sales project.

1. **No Missing Values**: The dataset is assumed and checked to be complete with no missing values, or any missing values have been appropriately handled through imputation or other methods.
2. **Sufficient Data**: The dataset contains a sufficient number of user-item interactions to provide meaningful patterns and ensure the reliability of the SVD model.
3. **Sparse Matrix**: The user-item interaction matrix is sparse, meaning that most users have rated only a small subset of the available items. SVD is well-suited to handle such sparsity.
4. **Gaussian Noise**: The error terms (residuals) in the approximated user-item interaction matrix are assumed to be normally distributed, allowing for effective minimization in the SVD computation.

# Model Evaluation and Technique

The following techniques and steps were involved in the evaluation of the model SVD

**1. Trimming Data**

**Purpose**: To build the SVD model faster we have taken only top 200000 rows.

**2. Performance Metrics**

**Root Mean Squared Error (RMSE)**: The square root of MSE, providing error magnitude in the same units as the data using cross\_validate function. Below is the output

{'test\_rmse': array([0.99451907, 0.99540665, 0.99564404]),

'fit\_time': (1.505462408065796, 1.5318565368652344, 1.534627914428711),

'test\_time': (0.562584638595581, 0.5317983627319336, 0.5320923328399658)}

**3. Recommendation Accuracy**

* **Purpose**: To ensure the model makes correct movie recommendation.

**Estimation score has been used for each movie a user can provide and based on that movie will be recommended.**

# Inferences from the Project

### **User Preferences**

* **Personalized Viewing Patterns**: The SVD model can identify latent factors that represent underlying viewing preferences, helping to understand what types of content individual users prefer (e.g., genre, director, cast).
* **User Segmentation**: Users can be segmented into distinct groups based on their viewing patterns, allowing for more targeted and personalized recommendations.

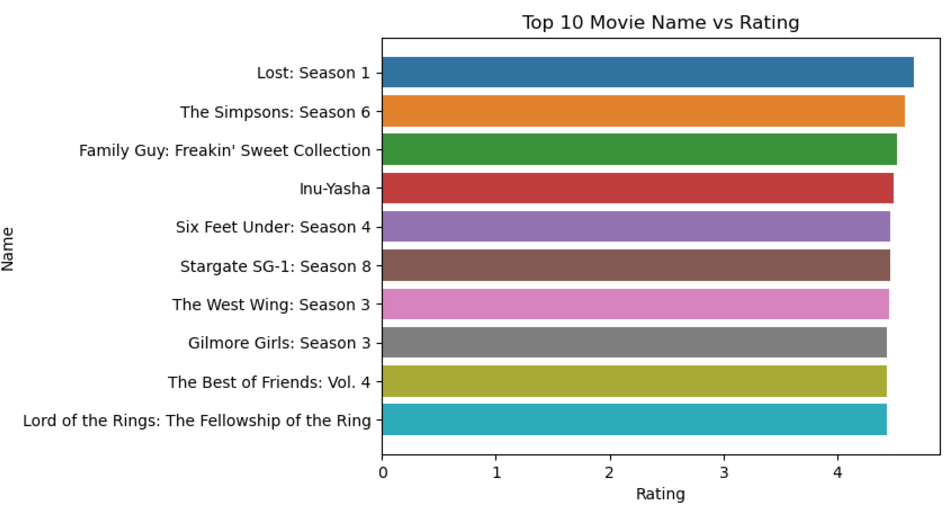
### **Content Popularity**

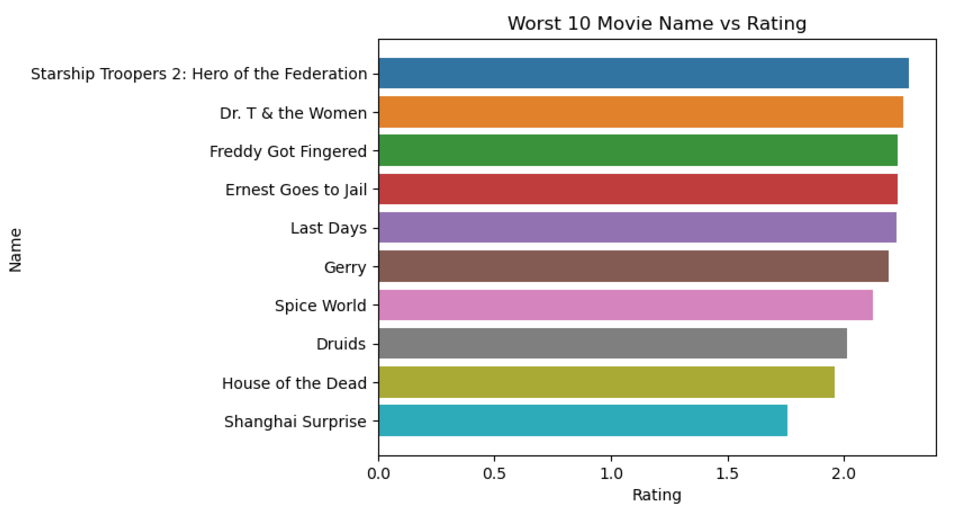
* **Popular Genres and Titles: Insights into which genres and specific titles are most popular among different user segments can help in content acquisition and production decisions.**
* **Trending Content: Identification of trending content based on current viewing patterns, aiding in timely promotions and marketing efforts.**

### **Recommendation Quality**

* **Accuracy of Predictions: Evaluation metrics such as RMSE and MAE provide an understanding of how accurately the model predicts user ratings, indicating the effectiveness of the recommendation system.**
* **User Engagement: Analysis of user interaction with the recommendations (e.g., click-through rates, watch completion rates) can indicate the relevance and attractiveness of the suggested content.**

**Rating for top 10 and worst 10 movies visualized**





# Future Possibilities

**Real-Time Recommendations**:

* **Streaming Data**: Implement real-time recommendation systems that update predictions based on streaming data, providing users with the most current and relevant recommendations as they interact with the platform.
* **Dynamic Adaptation**: Develop models that adapt in real-time to changes in user preferences and new content, ensuring the system remains responsive to emerging trends.

**Personalization at Scale**:

* **User Profiling**: Create detailed user profiles by integrating data from various sources (e.g., viewing history, search queries, social media interactions) to provide more personalized and accurate recommendations.
* **Context-Aware Recommendations**: Incorporate contextual factors such as time of day, device type, and user location to tailor recommendations to the user's current context.

**Leveraging External Data**:

* **Social Media Integration**: Use data from social media platforms to gauge user sentiment and trends, enhancing the recommendation system with real-time insights from external sources.
* **Cross-Platform Recommendations**: Explore recommendations across different content platforms (e.g., music, books) to provide a holistic entertainment experience based on diverse user interests.

**Enhanced User Interaction**:

* **Interactive Interfaces**: Develop more interactive recommendation interfaces that allow users to provide feedback (e.g., thumbs up/down, ratings) to refine and improve future recommendations.
* **Exploratory Recommendations**: Offer users exploratory recommendation options that encourage them to discover new content outside their usual preferences.

# Conclusion

This project aimed to develop a Netflix recommendation system using Singular Value Decomposition (SVD) to provide personalized content recommendations to users. Through a systematic approach involving data pre-processing, model building, evaluation, and analysis, several key insights and outcomes were achieved.

#### Key Insights:

1. **User Preferences**: The SVD model successfully identified latent factors representing user preferences and content attributes, allowing for highly personalized recommendations.
2. **Content Popularity**: Analysis revealed popular genres and titles, highlighting trends and preferences across different user segments, which can inform content acquisition and production strategies.
3. **Recommendation Quality**: The model demonstrated good prediction accuracy, with evaluation metrics such as RMSE and MAE indicating reliable performance. User engagement metrics further validated the relevance of the recommendation.