# **THE NETFLIX DATASET CHALLENGE**

Movie Rating Prediction based on the Netflix Dataset

**GROUP: 3 TEAM MEMBERS:**

Jaishree Bangalore

Varsha Tekale

Chandan Jagadesh

Krishna Bang

Pingali Venkata Praveen

**Contents**

**Page no**

1. **Executive Summary……………………………………………………………………………………………………….3**
2. **Project Motivation………………………………………………………………………………….4**

**2.1. Objective...........................................................................................4**

**2.2. Netflix Data Description…………………………………………………………………..4**

**2.3. Descriptive statistics………………………………………………………………………..5**

**2.4. Insights of descriptive statistics………………………………………………………..7**

1. **Introduction to BI topics chosen……………………………………………………………..8**

**3.1. Linear modelling…………………………………………..………………………………….8**

**3.2. Collaborative filtering………………………..…………………………………………….8**

**3.2.1. Item based collaborative filtering………………………………….…….9**

**3.2.2. Prediction of ratings using similarity index…………………………..9**

1. **Business Applications & Value……………………………………………………………….10**
2. **BI Techniques used………………………………………………………………………………..12**

**5.1 Linear Regression Models………………………………………………………………..12**

* 1. **Item based Collaborative filtering…………………………………………………….14**

1. **Challenges…………………………………………………………………………………………….19**
2. **Concise case study………………………………………………………………………………..20**
3. **Conclusion…………………………………………………………………………………………….21**
4. **References……………….…………………………………………………………………………..221.** **EXECUTIVE SUMMARY**

The Netflix Dataset challenge was an open challenge that concluded in 2009. The objective of the challenge was to come up with an algorithm that could improve on the predictions of Netflix’s “Cinematch”.

The data provided for the rating prediction consists only of the movie identification number, a number of identifying the users who have rated the movies, the date on which a certain movie was rated and the rating itself. The two major approaches were employed in the rating prediction: Linear Regression and Item Based Collaborative Filtering.

**Linear Models**:

* Since the dataset doesn’t contain any explanatory variables (but only 2 nominal variables: MovieID and UserID/CustID) linear regression cannot directly be used. For this we derived 3 derived variables each attributed to either the movie or the user. The average rating of each movie and the rating frequency tell us about how good the movie is and how popular the movie is.
* The average rating of each user tell us about how stingy or lenient a user is while rating movies. Both the averages have been subjected to power transformations to approximate normality to aid in more robust estimation of the regression parameters. Interaction effects were also specified between the average of the user ratings and the frequency of rating. This was done on a 20% dataset (i.e. 20 million rows) and the model yielded a RMSE of 0.9145.

**Item Based Collaborative Filtering**:

* This is a model based algorithm used mostly by recommender systems. A very small sample of the top 10 movies and the top 500 users was used to run this algorithm due to the numerical complexity and the limited computational capacity available. The most heartening results were achieved when the similarity index was calculated using cosine similarity and the predicted rating was computed using the similarity weighted average of the ratings.
* Even considering the very dense sample we used for this algorithm the model yielded a fairly low RMSE of 0.8875. Other approaches such as Euclidean similarity and Pearson correlation coefficient were used for calculating the similarity index between the movies and kNN method was used to predict the ratings, though they yielded significantly higher RMSE.

1. **PROJECT MOTIVATION**

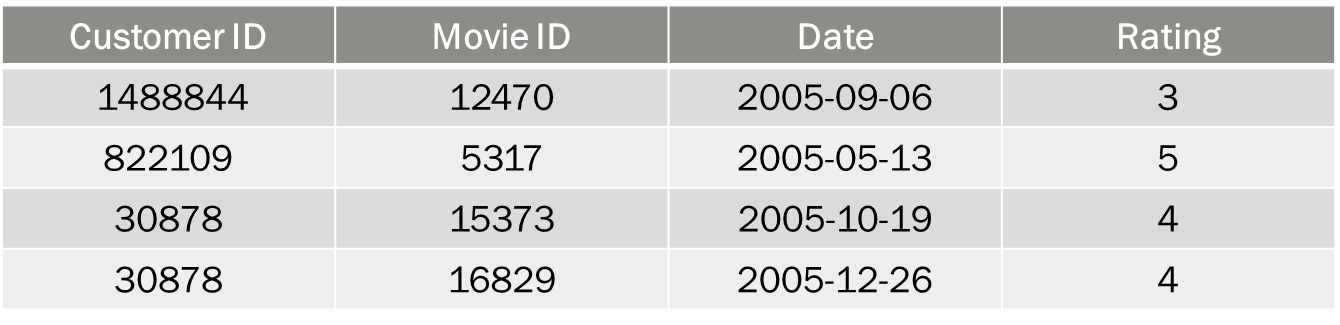
In this project the objective is to predict the movie ratings using Netflix dataset. For predicting the ratings different algorithm are used such as statistical linear models and collaborative filtering systems. The purpose of recommendation systems (also known as collaborative filtering systems) is to recommend movies which a customer is likely to order. These recommendations are based on statistics of a customer’s purchasing history and the histories of customers with similar interests. Examples of recommendation systems can be found at amazon.com and netflix.com. Apart from the usefulness consideration, we had a personal motivation to work on this project because Netflix offers a million $1 to the person(s) who can beat their baseline system by 10% and in future we can actually take part in such competition which will increase our knowledge of problem solving.

**2.1. OBJECTIVE:**

To predict the movie ratings using Netflix dataset by implementing various statistical linear models and collaborative filtering methods.

**2.2. NETFLIX DATA DESCRIPTION:**

Following is the brief description of the data set:



* There are 17770 movies.
* There are 480189 users. Customer IDs range from 1 to 2649429, with gaps.
* Ratings are based on a scale of 1 to 5.
* Date (Year of Release) range from 1890 to 2005.
* Training set consists of 100 million records.
* Qualifying dataset size is 2817131. It contains from 1-9999 movies ids. Prediction needs to be submitted on this dataset.
* Probe dataset size is 1408395. It contains from 1-9999 movies ids. This dataset is meant to be used for checking the RMSE before proceeding for qualifying dataset prediction.

**2.3. DESCRIPTIVE STATISTICS**

Some interesting findings from the provided data sets.

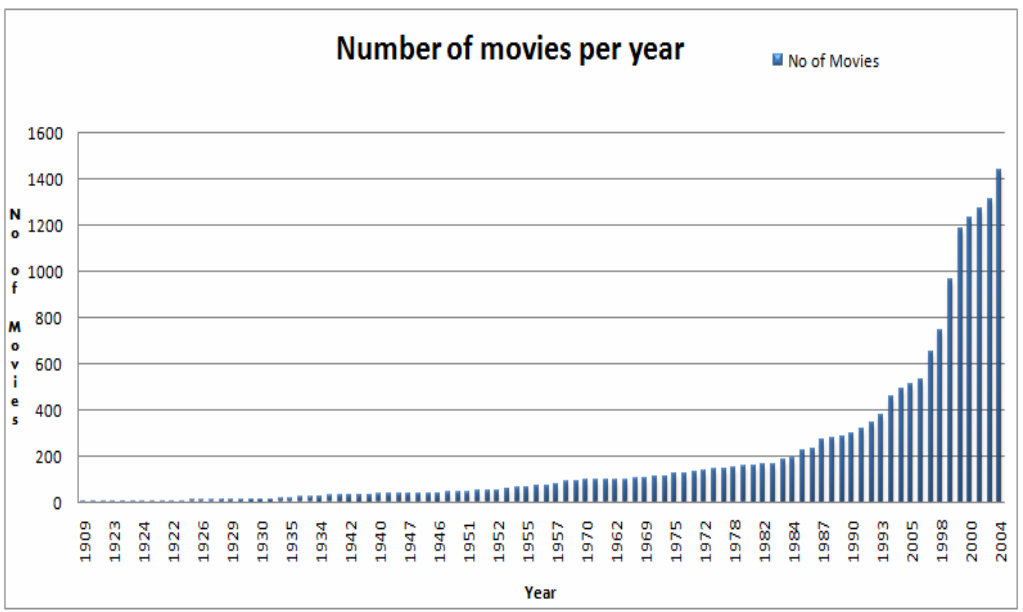


Fig.1: Number of movies per year from 1909 – 2004

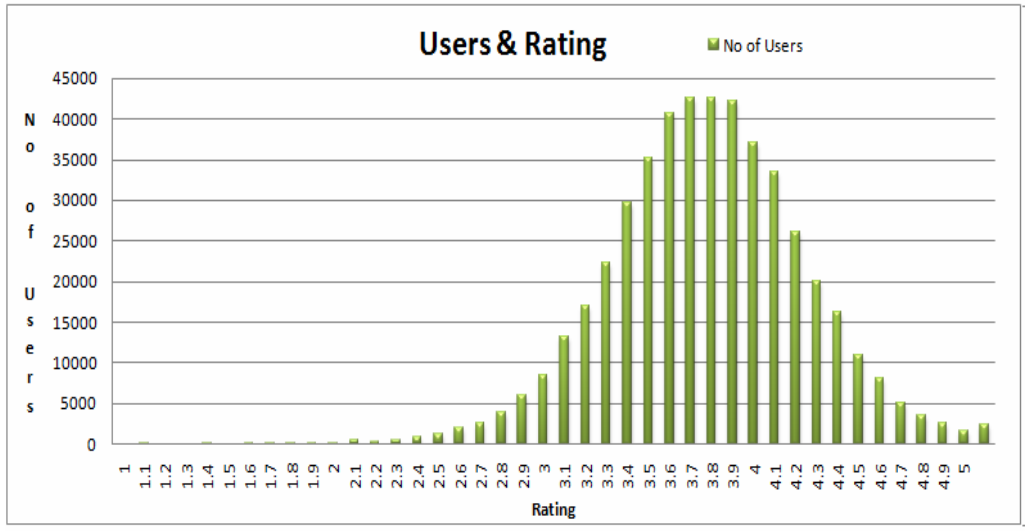


Fig.2: Number of users ratings ranging from 1 – 5

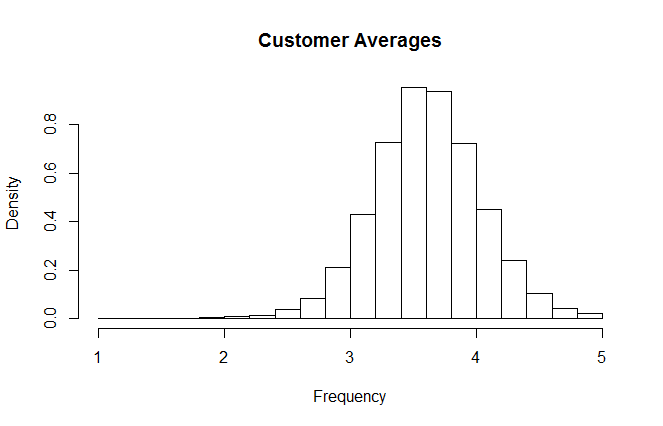


Fig.3: Average customer ratings from 1 – 5

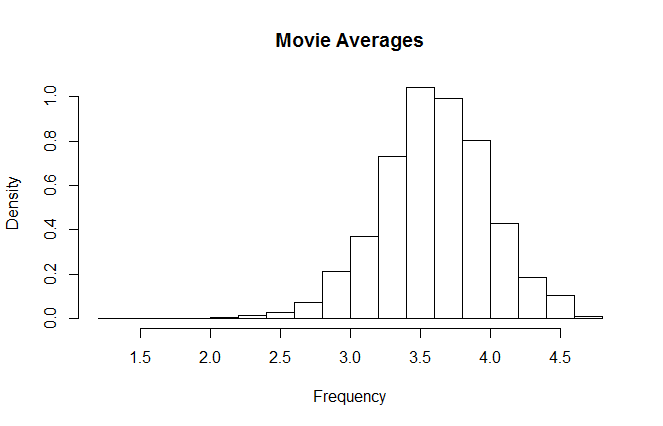


Fig.4: Average movie ratings from 1 – 5

**2.4. INSIGHTS OF DESCRIPTIVE STATISTICS**

* The total number of movies on Netflix increased in the year 2000 to 2004 and decreased in the year 2005 as shown in Fig.1
* There are very less users who gave ratings below 3 and above 4.5. Maximum above 40,000 users gave rating 3.7 and 3.8
* Average movie rating and average user rating is 3.6 – 3.7.

1. **INTRODUCTION TO THE CHOSEN BI TOPIC**

Cinematch (Netflix’s algorithm) does straightforward statistical linear models with a lot of data conditioning. But a real-world system is much more than an algorithm, and Cinematch does a lot more than just optimize for RMSE.

The winning team of Netflix competition used the collaborative filtering algorithm to predict the movie rating.

Our approach to this project involves both the methods, linear modelling as well as collaborative filtering method.

**3.1. LINEAR MODELLING:**

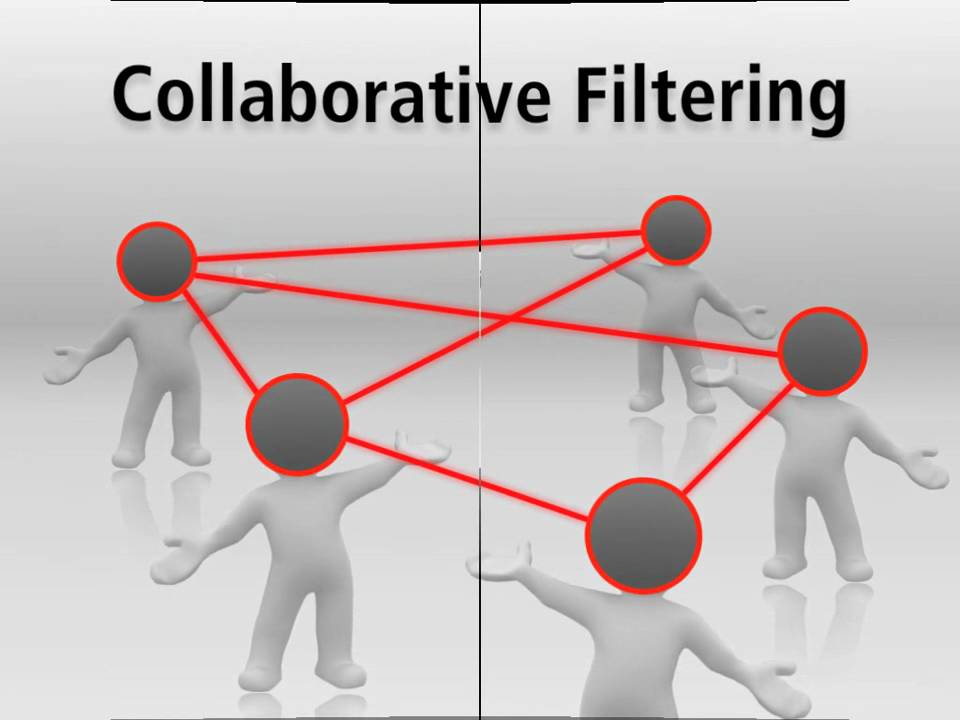
Our approach to this problem involves test & trial approach where we first test a few simple approaches and progress to more advanced ones, and report the findings.

As our data set contains only four variables and two of them (customer ID and movie ID) are nominal, we created three derived variables which are used to run the regression model.

Derived Variables are:

* Average rating of the user
* Average rating of the movie
* Frequency of rating
* Interaction variables between average rating of the user and average rating of the movie
* Box-cox transformed variable of average rating of the user
* Box-cox transformed variable of average rating of the movie

**3.2. COLLABORATIVE FILTERING:**

****

Collaborative filtering (CF) is the method of making automatic predictions (filtering) about the interests of a user by collecting taste information from many users (collaborating). The underlying assumption of CF approach is that: those who agreed in the past tend to agree again in the future. A lot of research has been done in this field giving birth to item-based filtering, user based filtering, content-based predictions and many mixture models. For example, in user based filtering, the user of the system provides ratings of some artifacts or items. The system makes informed guesses about other items the user may like based on ratings other users have provided. This is the framework for social filtering methods. In item-based filtering, the system accepts information describing the nature of an item and based on a sample of the user’s preferences and learns to predict which items the user will like, usually called content based filtering as it does not rely on social information, in the form of other users’ ratings.We aim to approach this problem using item based collaborative filtering.

**3.2.1. ITEM BASED COLLABORATIVE FILTERING**

Item based collaborative filtering is a model-based algorithm for recommender engines. In item based collaborative filtering similarities between items are calculated from rating-matrix. And based upon these similarities, user’s preference for an item not rated by him is calculated.

Item-Item based similarity matrix was generated using 3 methods:

1. Using Euclidean Distance
2. Correlation Coefficient
3. Cosine Similarity Matrix

**3.2.2. PREDICTION OF RATINGS USING SIMILARITY INDEX**

For each user, we next predict his ratings for items that he had not rated. To calculate we weigh the just-calculated similarity-measure between the target item and other items that user has already rated. The weighing factor is the ratings given by the user to items already rated by him. We further scale this weighted average with the average of similarity-measures so that the calculated rating remains within a predefined limit. We used two different methods to predict the ratings of the customers.

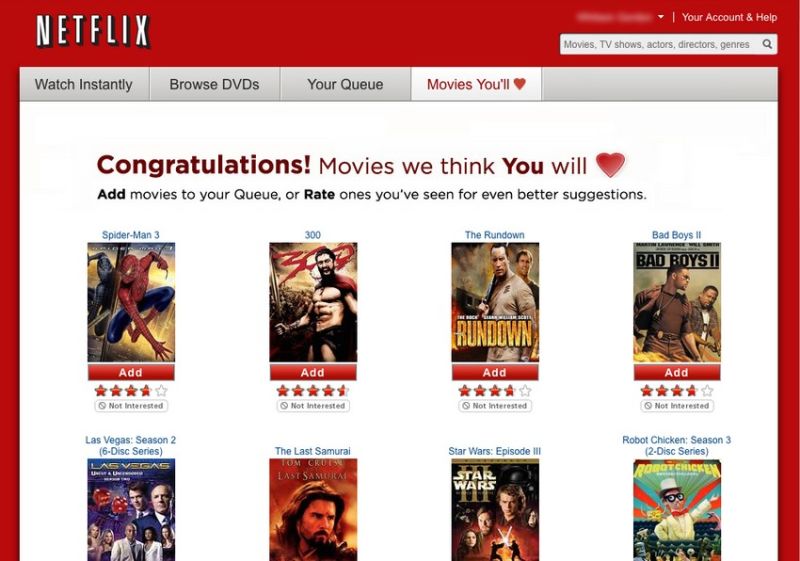
1. kNN : Average of k – nearest neighbours
2. Similarity weighted average of rating
3. **BUSINESS APPLICATIONS & VALUE**

*Top Recommendation business applications*

**

**Movie Recommendations:**

Netflix offers its customers recommendations of movies they might like. These recommendations are based on ratings provided by users. The importance of predicting ratings accurately is so high, that Netflix offered a prize of one million dollars for the first algorithm that could beat its own recommendation system by 10%.1 The prize was finally won in 2009, by a team of researchers called “Bellkor’s Pragmatic Chaos,” after over three years of competition.



**Product Recommendations:**

Perhaps the most important use of recommendation systems is at on-line retailers. We have noted how Amazon or similar on-line vendors strive to present each returning user with some

Suggestions of products that they might like to buy. These suggestions are not random, but are based on the purchasing decisions made by similar customers or on other techniques.



**News Articles:**

News services have attempted to identify articles of interest to readers, based on the articles that they have read in the past. The similarity might be based on the similarity of important words in the documents, or on the articles that are read by people with similar reading tastes. The same principles apply to recommending blogs from among the millions of blogs available, videos on YouTube, or other sites where content is provided regularly.

1. **BI TECHNIQUES USED**

**5.1 LINEAR REGRESSION MODELS:**

**Model 1:**

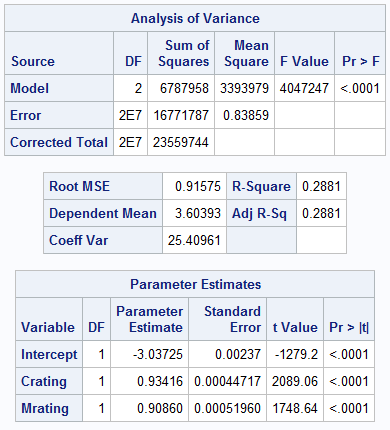
**DATA MINING OBJECTIVE:** To execute Liner regression model on the sample data of 20% (20m records) along with derived variables. The regression will help identify the better model with least root mean square error.

**DATA:** Dataset contains Customer ID, Title, Rating, Average customer rating, and average movie rating.

Training data set can be found in the below link.

[**https://drive.google.com/folderview?id=0B7UDXHgGyQfxLVRHRTdCWWdzTzA&usp=sharing**](https://drive.google.com/open?id=0B7UDXHgGyQfxLVRHRTdCWWdzTzA)

**DATA MINING RESULT:** Model contains following variables: Rating as dependent variable, Average customer rating, average movie rating as independent variables.



**Fig 5.1 ANOVA table of the Model 1**

**The root mean square error for the above model yields RMSE = 0.9157**

**Model 2:**

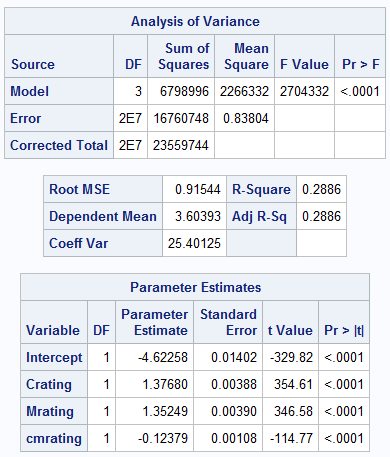
**DATA MINING OBJECTIVE:** To execute Liner regression model on the sample data of 20% (20m records) along with derived variables. The regression will help identify the better model with least root mean square error.

**DATA:** Dataset contains Customer ID, Title, Rating, Average customer rating, average movie rating and interaction variable of Average customer rating and average movie rating as independent variables.

The training data set can be found in the below link.

[**https://drive.google.com/folderview?id=0B7UDXHgGyQfxLVRHRTdCWWdzTzA&usp=sharing**](https://drive.google.com/open?id=0B7UDXHgGyQfxLVRHRTdCWWdzTzA)

**DATA MINING RESULT:** Model contains following variables: Rating as dependent variable, Average customer rating, average movie rating, interaction variable of averages interaction variable between Average customer rating and average movie rating as independent variables.



**Fig 5.2 ANOVA table of the Model 2**

**The root mean square error for the above model yields RMSE = 0.9154**

**Model 3:**

**DATA MINING OBJECTIVE:** To execute Liner regression model on the sample data of 20% (20m records) along with derived variables. The regression will help identify the better model with least root mean square error.

**DATA:** The dataset contains derived variables which are power transformed to be approximately normalized. Average customer rating and average movie rating have under gone box cox transformation to obtain a better normalized distribution.

The training data set can be found in the below link.

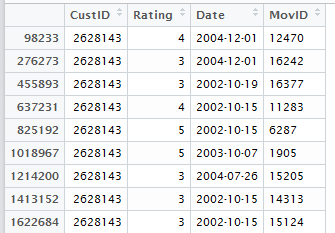
[**https://drive.google.com/folderview?id=0B7UDXHgGyQfxLVRHRTdCWWdzTzA&usp=sharing**](https://drive.google.com/open?id=0B7UDXHgGyQfxLVRHRTdCWWdzTzA)

**DATA MINING RESULT:** Model contains following variables: Rating as dependent variable, Average customer rating, average movie rating, interaction variable of averages, interaction between frequency of total movies rated and average customer movie ting as independent variables.

**The root mean square error for the above model yields RMSE = 0.9145**

**5.2 ITEM BASED COLLABORATIVE FILTERING:**

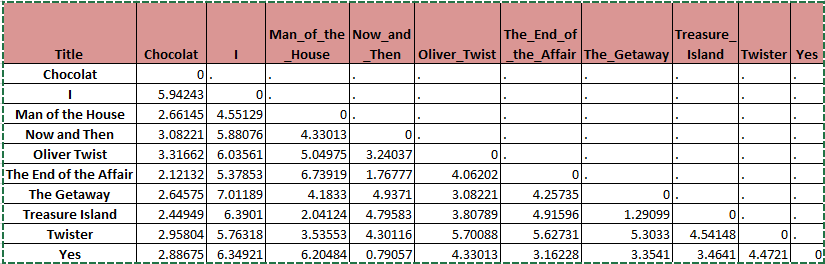
The data on which we are applying item based collaborative filtering is given below. It has four variables namely CustID, Rating, Date and MovID.



Step 1: Similarity Index

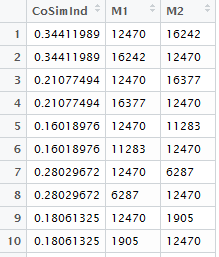
We will now create an item-to-item similarity matrix. The idea is to calculate how similar an item is to another item. There are a number of ways of calculating this:

1. **Using Euclidean Distance**



1. **Cosine Similarity Matrix**

Here is a similarity index between the movies using cosine similarity:



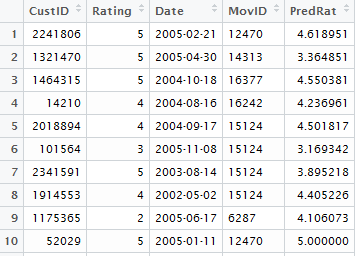
Step 2: Prediction of ratings using similarity index

For each user, we next predict his ratings for items that he had not rated. To calculate we weigh the just-calculated similarity-measure between the target item and other items that user has already rated. The weighing factor is the ratings given by the user to items already rated by him. We further scale this weighted average with the average of similarity-measures so that the calculated rating remains within a predefined limit.

We used two different methods to predict the ratings of the customers.

1. kNN : Average of k – nearest neighbours
2. Similarity weighted average of rating

In our case the training data and validation data are same. The whole objective of the project is to arrive at that prediction algorithm which yields least RMSE (Root Mean Square Error) compared to other prediction models which we are analyzing. Hence we worked on to predict the ratings for which the values are already available. This helps in calculating RMSE which, uses the difference between Actual and predicted values.



The above table consists of original four variables with the addition of one more variable i.e. Predicted Ratings of the movie. The “Rating” column shows the actual ratings which user gave and the “PredRat” column shows predicted ratings. Here, we can easily compare the actual and predicted ratings of the users.

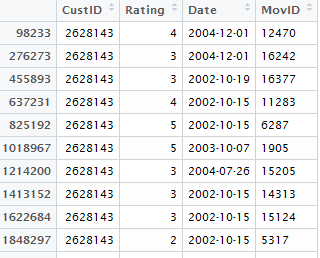
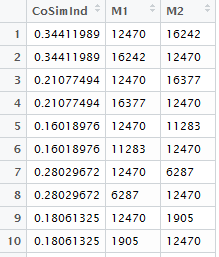
**For Instance: In the first row, for movie ID 12470, the predicted rating is approx. 4.62 and the actual rating is 5.**

In the same way we can compare all the predicted ratings with their actual ratings.

The overall flow of item based collaborative filtering is shown in the below diagram.

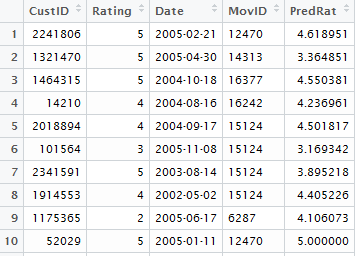
Similarity matrix using cosine similarity

Sample Data (Top 10 movies and top 500 users

Rating prediction using weighted average

Calculating similarity between movies.



The original vs Predicted movie ratings

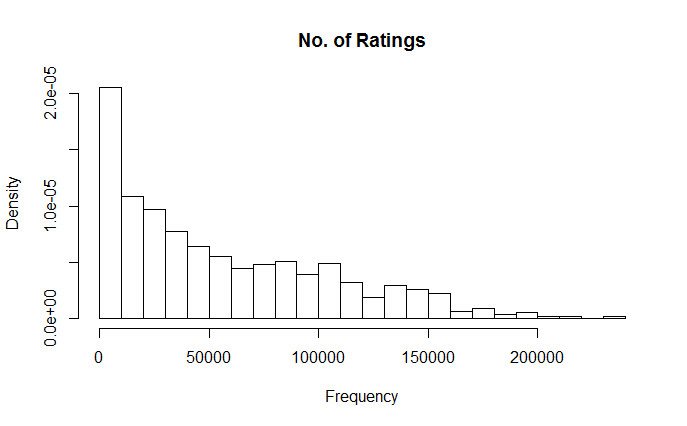
We can calculate the RMSE (Root Mean Square Error) of our model, which shows how accurate our model is.

**The RMSE value of the ratings using item based collaborative filtering on the Netflix’s dataset is 0.8875.**

Note: The million $1 winning team i.e. team BellKor in the Netfix competition had the RMSE value of 0.8643 which is approximately 0.2 away from the RMSE value we got.

1. **CHALLENGES**

* Very few explanatory variables. Except User ID, Movie ID & Date no other parameters available for prediction.
* Sparse data set. Very few ratings for some movies.



* Due to a very large amount of data, some computations required a long time.
* Some other computations were not possible due to memory limitations.
* No in-built functions available in most statistical packages like Base SAS & R for collaborative filtering.

1. **CONCISE CASE STUDY**

Netflix’s history has been one of tremendous success, innovation and now power struggle. First, the Company partnered with both DVD manufacturers and movie studios to bolster the market of DVDs and enable Netflix to dominate the movie rental market previously dominated by Blockbuster. Second, Netflix seamlessly carried over its existing large subscriber base and became the first Company to successfully penetrate the internet video streaming market.

While this transition was once thought of as a story of highly defensible first mover advantage, in actuality the legitimacy of internet streaming invited a host of different ecosystem player to enter as direct rivals. This increased competition shifted power from Netflix to content creators, increasing the cost of content acquisition and eroding the Company’s profitability. Now, Netflix is redefining itself in order to differentiate and capture value by focusing on developing its own original content.

This strategy combined with increased household device penetration and international expansion should secure Netflix’s place as an essential component of the living room and enable the Company to return to profitability and possible to increase prices. However, Netflix should keep its lens wide open and identify the possible adoption chain risks associated with broadband usage and access point channels and use creative strategic decision making to mitigate these risks.

1. **CONCLUSION**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method Used** | **Method Specification** | **Sampling Method** | **Accuracy?**  **RMSE** |
| **Linear Regression** | **Transformed variables with Interaction effects** | **20% random sample**  **(i.e. 20 million observations)** | **0.9145** |
| **Collaborative Filtering** | **Item Based:**  **Using Correlation and kNN average prediction (k=5)** | **Top 10 most popular movies**  **Top 500 users** | **0.9260** |
| **Collaborative Filtering** | **Item Based:**  **Using Euclidean distance and kNN average prediction (k=5)** | **Top 10 most popular movies**  **Top 10 users** | **0.9025** |
| **Collaborative Filtering** | **Item Based:**  **Using Cosine similarity and weighted average** | **Top 10 most popular movies**  **Top 500 users** | **0.8875** |

1. **REFERENCE**
2. The code and data used for the project can be found in the below link

[**https://drive.google.com/folderview?id=0B7UDXHgGyQfxLVRHRTdCWWdzTzA&usp=sharing**](https://drive.google.com/open?id=0B7UDXHgGyQfxLVRHRTdCWWdzTzA)

1. Machine Learning Project: Netflix Competition

[**http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.386.433&rep=rep1&type=pdf**](http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.386.433&rep=rep1&type=pdf)

1. The BellKor Solution to the Netflix Grand Prize

<http://www.netflixprize.com/assets/GrandPrize2009_BPC_BellKor.pdf>

1. Wikipedia

[**https://en.wikipedia.org/wiki/Netflix\_Prize**](https://en.wikipedia.org/wiki/Netflix_Prize)