

Recommendation System Using SVD Method

Student ID and Name:

P Saran Pandian(202018004)

Akash Gupta(202018015)

Bhavya Jain(202018023)

Sharvari Gokhale(202018038)

Project Summary:

Dataset:

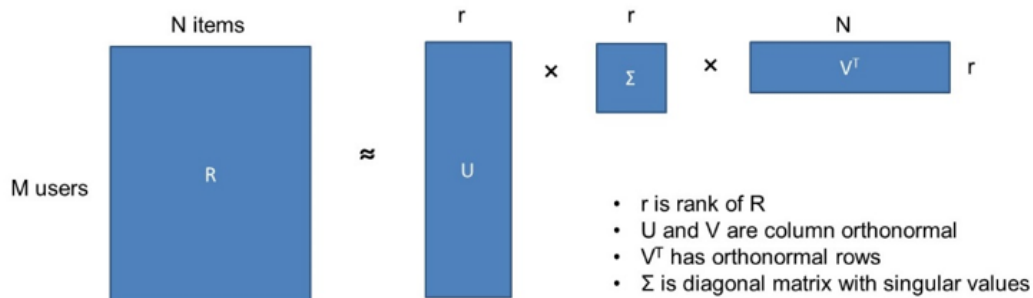
The dataset is collected from MovieLens. It contains 100836 ratings and 3683 tag applications across 9742 movies. These data were created by 610 users between March 29, 1996 and September 24, 2018. This dataset was generated on September 26, 2018.

The data are contained in the files links.csv, movies.csv, ratings.csv and tags.csv. The dataset is converted to a Utility matrix which has users as rows and movies as columns. Null values are converted to 0 which makes the utility matrix sparse.

SVD Decomposition:

The sparse matrix is decomposed into three matrices using the SVD method (say U,S,V). The U matrix contains the latent factors of users. The V matrix contains latent factors of movies. S matrix contains the strength of each latent factor. We reduce the dimension of column and row of the U and V matrix to a value k. Let's assume the number of users is M and number of movies as N. then the reduced dimension of U and V will be $M \times k$ and $k \times N$ respectively from $M \times \min(M,N)$ and $\min(M,N) \times N$. This k value is chosen by finding the number of elements in S which constitute 95% of the total sum of elements in S.

$$\sum_{i=1}^k S_i \geq 0.95 \times \sum_{j=1}^{\min(M,N)} S_j$$



Recommendation engine 1:

User-Item Collaborative Filtering

For *User-Item Collaborative Filtering* the similarity values between users are measured by observing all the items that are rated by both users.

The U matrix is used for finding the similar users for a given user. Each row of the U matrix represents the latent vector of each user. Users with similar interests towards a set of movies have most similar latent vectors. To find the similar users we first multiply U with S matrix and then we use cosine similarity values between latent vectors of users. On this basis the users with top 5 cosine similarity scores are picked And recommend him with movies with good ratings to that user which he has never watched by him.

Recommendation Engine 2:

Item-Item Collaborative Filtering

The similarity values between items(movies) in *Item-Item Collaborative Filtering* are measured by observing all the users who have rated both movies. This type of recommendation engine has an advantage that the movies don't change their genres and content unlike the user's preferences. So the recommended movies would seldom not fit in the same category again.

Here, the V matrix is used for finding the movies which are similar to each other. V matrix represents the latent features vectors for the movies. Cosine Similarity is used for identifying the top 10 movies. Movies having cosine similarity close to the required movie. The top 10 movies recommended will be lying in the same category, genres, etc, of the movie the user has watched.