IT350 Assignment 1 - Report

Sachin Prasanna

January 24, 2024

1 Motivation

Suppose we are given a user - movie dataset, wherein each instance is a user and each feature is a movie, and the table is filled with the ratings of each user on that particular movie.

The goal is to minimise the number of features needed to represent this data.

2 Dataset

The selected user-movie dataset for this assignment was the **The Movies Dataset** available on Kaggle. The dataset consists of movies released on or before July 2017. Data points include cast, crew, plot keywords, budget, revenue, posters, release dates, languages, production companies, countries, TMDB vote counts and vote averages.

I have used the **ratings_small.csv** file from this dataset for this assignment. It contains 4 features - *userId*, *movieId*, *rating*, *timestamp*.

3 Preprocessing

The *timestamp* is irrelevant for the problem and is dropped. The dataset is transformed such that each instance is a user and each feature is a movie, and the table is filled with the ratings of each user on that particular movie.

The dataset had 5983282 NaN values out of the 6083286 values. Hence, NaN values contributed to **98.35%** of the dataset. These NaN values could mean the rating was 0 or the user had not rated the movie.

Hence, the features and instances with a lot of NaN values were dropped as described in the following subsections.

3.1 Features

The dataset contained 9066 features. Using the figure 1 and figure 2, it was found that only around 1% of the features had more than 19% non NaN values. Hence, after removing the features with more than 81% NaN values, only 80 features remained, which could then be analysed using dimensionality reduction techniques.

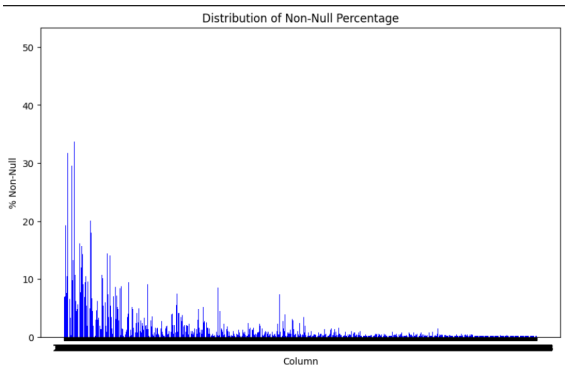

Figure 1: Distribution of Non NaN percentages

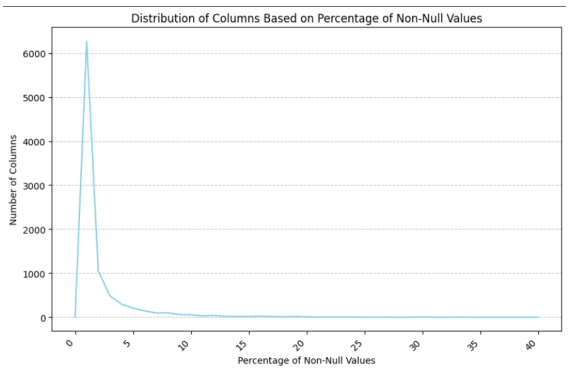

Figure 2: Distribution of Columns Based on Percentage of Non-Null Values

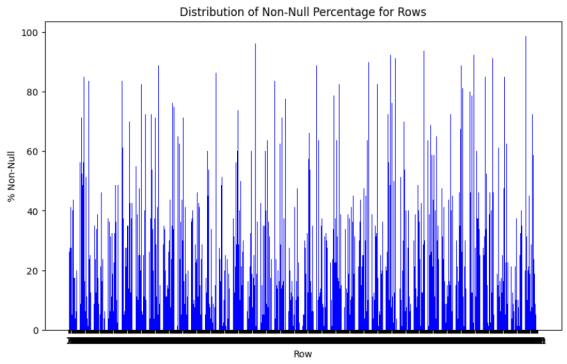

Figure 3: Distribution of Non NaN percentages for Rows

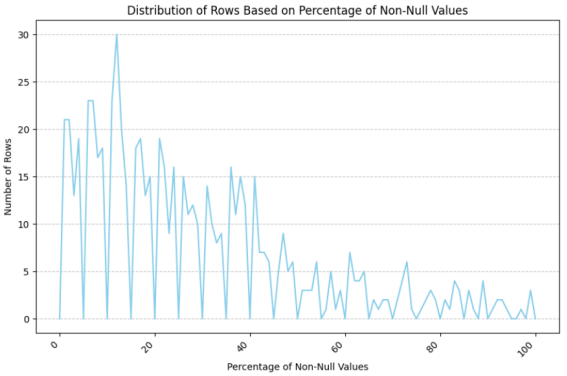

Figure 4: Distribution of Rows Based on Percentage of Non-Null Values

3.2 Instances

The dataset contained 671 instances. Using the figure 3 and figure 4, it was found that only around 50% of the instances had more than 21% non NaN values. Hence, these after removing features with more than 79% NaN values, only 364 instances remained.

This was mainly done to make sure that a user is a dependable user who has rated a decent amount of movies and their rating has some level of truth to it.

3.3 Filling NaN Values

After dropping of features and instances the number of NaN values that remained were 16675. These were imputed with 0, signifying the user did not like the movie at all.

4 Need for PCA and SVD

A correlation heat map was plotted to see the correlation between the features (Figure 5). This also essentially serves as the multiple dimension visualisation. Using the colour coding,

Statistic	Value
Mean Covariance	0.5448
25th Percentile	0.0969
50th Percentile (Median)	0.5093
75th Percentile	0.9335
Standard Deviation	0.6179

Table 1: Covariance Statistics

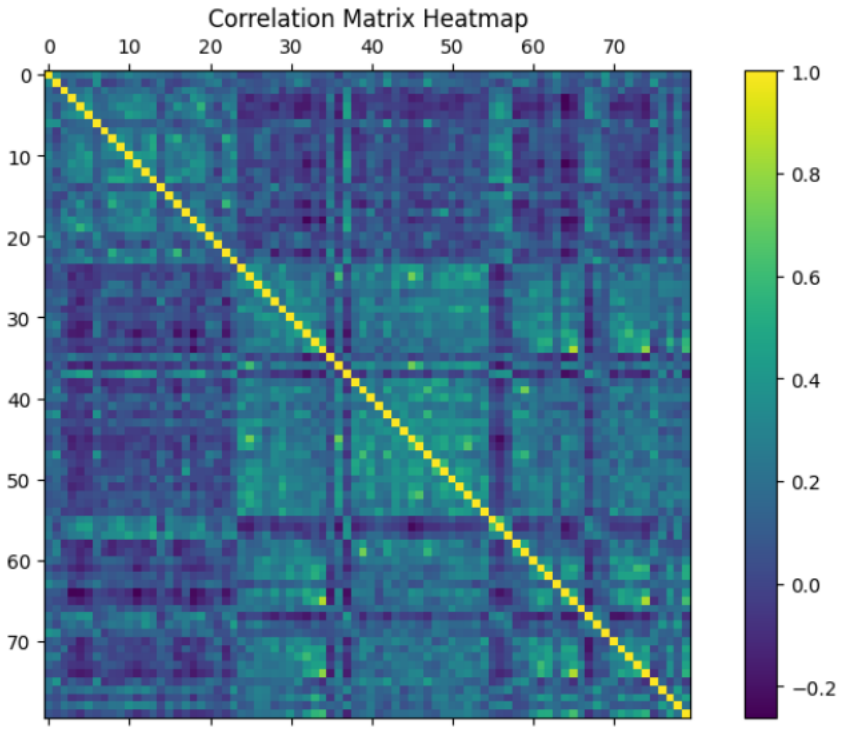

Figure 5: Correlation Heat Map

it was clear that there was a good amount of correlation between many columns. Hence, dimensionality reduction techniques like SVD and PCA could be used to take advantage of it.

Also, the value of covariance between every every 2 feature was taken and analysed. The results are captured in Table 1. It was clear that the features in the dataset were related to one and other and could be reduced.

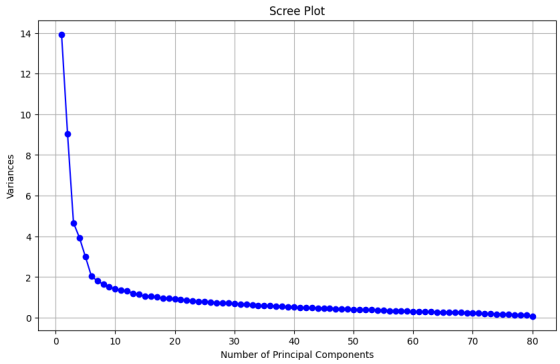

Figure 6: Scree Plot

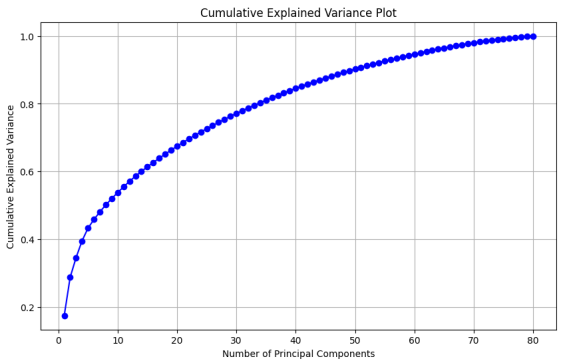

Figure 7: Cumulative Variance Plot

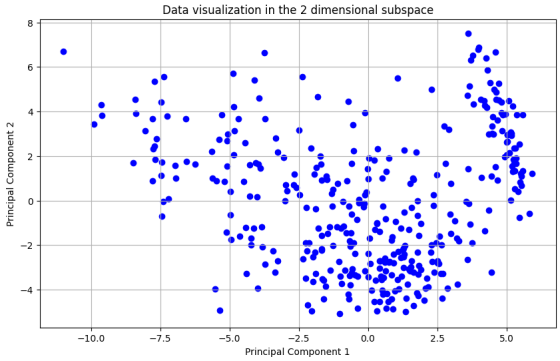

Figure 8: Projection in 2 dimensions

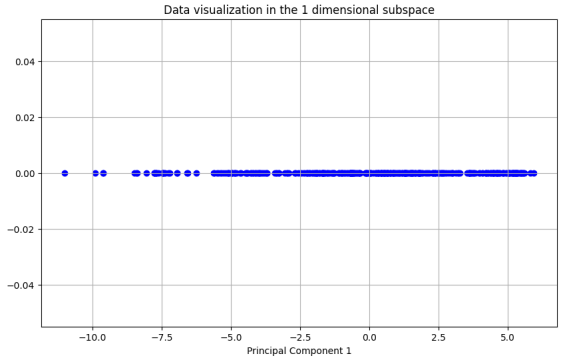

Figure 9: Projection in 1 dimension

5 Principle Component Analysis (PCA)

5.1 Dimensionality Reduction

Principle Component Analysis was implemented and the top k dimensions were selected using the Kaiser criterion. In this method, the components corresponding to the eigenvalues which are greater than the average eigenvalues are retained. Another method can be selecting those components which contribute to 80% variance of the total data. Another approach is the scree plot, which involves plotting the eigenvalues against the component number. The "elbow" of the plot is examined, and components before the elbow are retained. This method helps in visually identifying the point where additional components contribute less to the overall variance. Using the Kaiser Criterion, the number of components selected was 17.

The Scree Plot and Cumulative Variance plots are shown in Figure 6 and Figure 7.

5.2 Projection of Data

The Projected data onto 2 and 1 dimension can be found in Figure 8 and Figure 9 respectively.

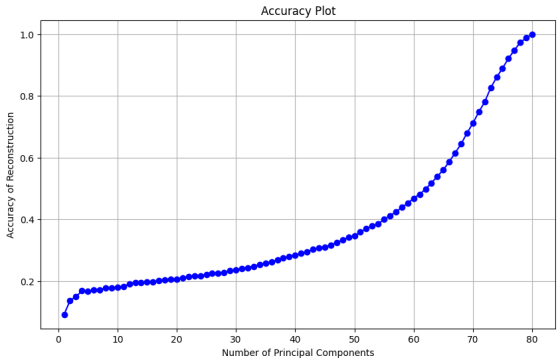

Figure 10: Accuracy Plot

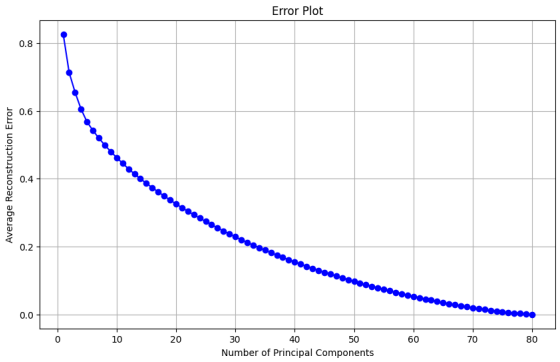

Figure 11: Error Plot

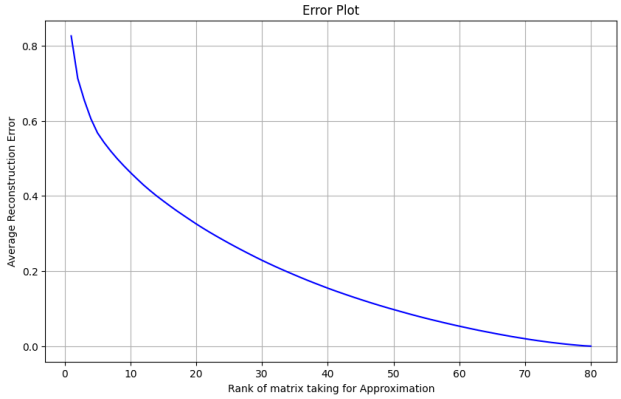

Figure 12: Error using Rank Approximations

5.3 Reconstruction of Data

The projected data in the lower dimensions was reconstructed. Then the reconstructed data was multiplied by the initial standard deviation and the original mean was added. The accuracy and error were then analysed and the graphs are shown in Figure 10 and Figure 11 respectively.

6 Singular Value Decomposition (SVD)

6.1 Rank Approximation

The original data can be approximated using different rank matrices. The error of approximation is detailed in Figure 12. As the rank of the matrix increased, the approximation error decreased. The optimal rank is found through a method called cumulative energy retention using a threshold of 0.80. The optimal rank using this method was 34. This means that 80% of the energy/information is retained by using a 34 rank matrix to approximate the data.

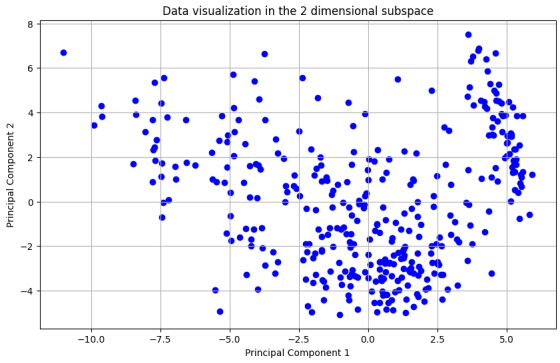

Figure 13: Projection in 2 dimensions

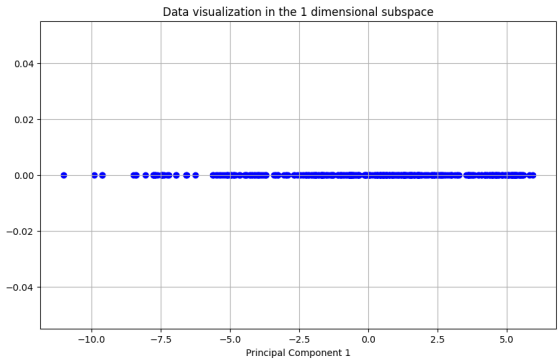

Figure 14: Projection in 1 dimension

6.2 Projection of Data

The Projected data onto 2 and 1 dimension can be found in Figure 13 and Figure 14 respectively.

6.3 Visualising the Matrices generated by SVD

The SVD generates 3 matrices, U, sigma and V. Sigma matrix gives us the strength of our concepts. V matrix maps the concepts to movies U matrix maps the users to concepts. They are visualised in Figure 15, Figure 16 and Figure 17.

- The sigma matrix shows the strength of each concept. They are 80 concepts in total and they are of varying strengths. For example, the first concept has a strength of more than 70, while the last concept has a strength of each than 10. This is due to the fact that the first concept maybe related to something which most movies have, for example Action. Whilst the the concept with the least strength is a combination of several different concepts in a linear combination and hence is not very strong.

- The U matrix maps each user to a concept. Figure 15 is a histogram plot and it brings out the most important concept which each user likes. This is found by getting the highest absolute value for that particular user. We can infer from the histogram plot that each concept has how many users associated to it. It gives a balanced plot.

- The V matrix maps each movie to a concept. Figure 16 is a histogram plot and it brings out the most important concept in each movie. This is found by getting the highest absolute value for that particular movie. We can infer from the histogram plot that some concepts are captured as the main concept in a lot of movies, but there are other concepts which are not really captured as the main concept in any movie.

7 t-sne Plot

The t-sne Plot is also a dimensionality reduction technique which takes into consideration the relation between two variables, rather than looking at a global view like the PCA. The

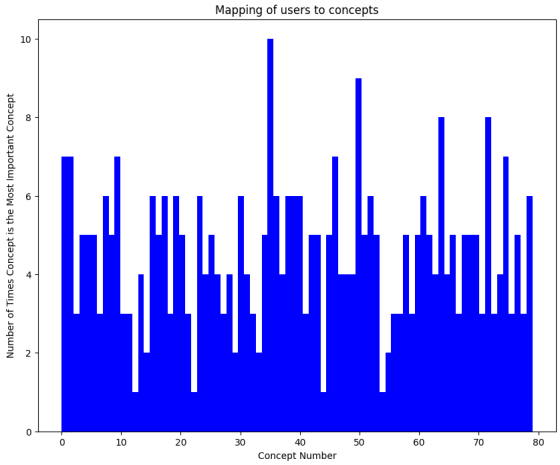

Figure 15: Mapping of users to concepts (U Matrix)

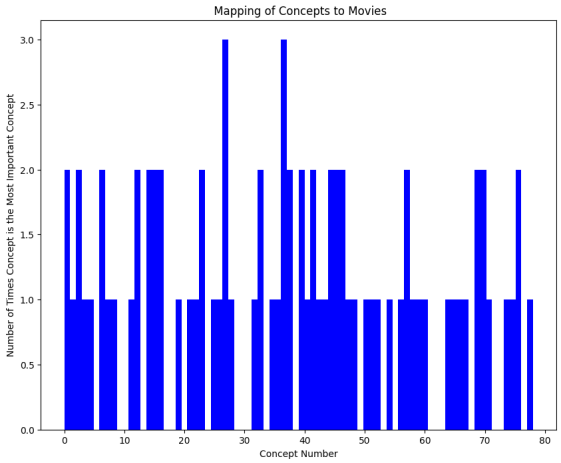

Figure 16: Mapping of concepts to movies (V Matrix)

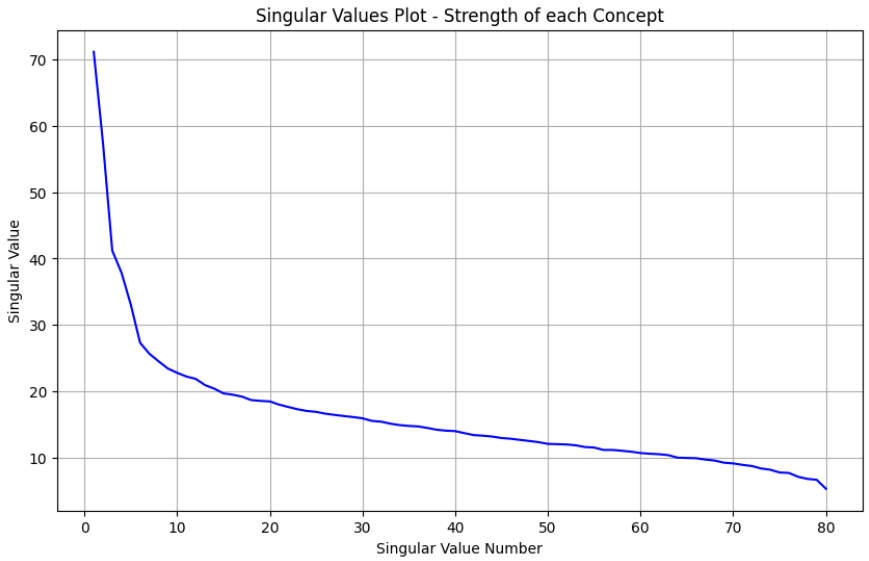

Figure 17: Strength of Each Concept (Sigma Matrix)

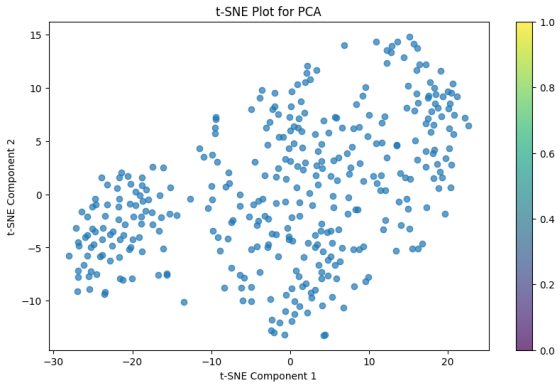

Figure 18: t-sne plot for PCA

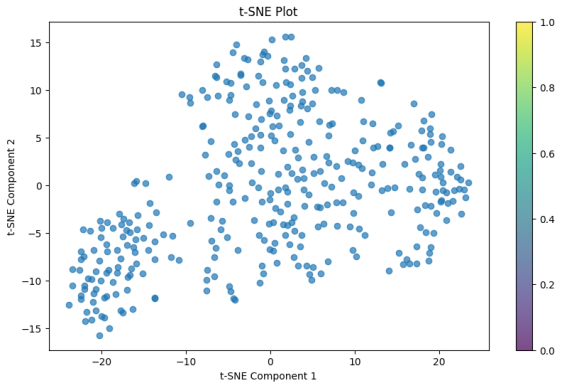

Figure 19: t-sne plot for SVD

t-sne plots were used to visualise the data in 2 dimensions after dimensionality reduction using PCA and SVD.

7.1 PCA

The t-sne plot for the PCA shows a nice spread of the data and a distinct cluster towards the left side. Other clustering is not very clear in this case, which signifies that there is no clear distinction between the data points to distinct clusters. The t-sne plot for the PCA is shown in Figure 18.

7.2 SVD

The t-sne plot for the SVD (Figure 19) also shows a similar trend as the t-sne plot for PCA. There is some level of clustering towards the left of the graph, which shows similarity.

8 Conclusion - How PCA and SVD have helped

PCA and SVD have helped us in the following ways as described in the points below:

- **Dimensionality Reduction:** PCA and SVD are excellent dimensionality reduction techniques, which help in reducing dimensions. Using PCA along with the Kaiser Criterion, the number of features were reduced from 80 to 17. The average reconstruction error was around 0.4 which says a good amount of information is preserved even when the dimensionality is reduced. Using SVD too, we were able to reduce the dimensions to 17 and observe the same results.

- **Patterns in data:** SVD helped in unveiling interesting patterns in the data as described in Section 6.3. These patterns can be used to map ratings of new users to the concepts they like, or even group users who are similar.

- **Removing Correlated Features:** Initially there was a lot of correlation between the features. When projected into the 2 dimensional dataspace, the correlation is

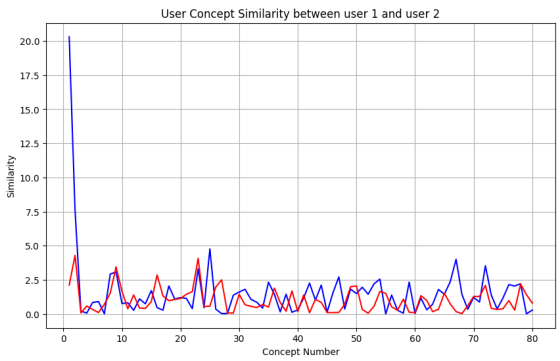

Figure 20: Concept Similarity between user 1 and user 2

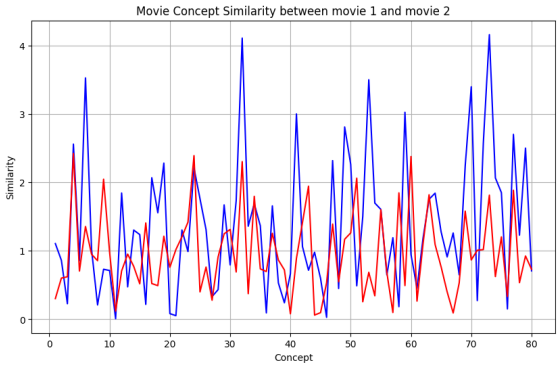

Figure 21: Concept Similarity between movie 1 and movie 2

maximised in both directions, indicating that we have recovered maximum possible information using just 2 dimensions.

- **Rank Approximation:** SVD can approximate the real data using the U, sigma and V matrices. Using the energy method, 80% approximation is obtained using a matrix of rank 34 (Section 6.1).

9 Extra Work Given by Sir

9.1 Add users to see how they map to concepts

Suppose we add 2 new users, *user1* and *user2* to the dataset and what are their movie preferences. Suppose the two user rating vectors are orthogonal to each other, then by simple dot product similarity, their similarity will be 0. But this may not be the case. SVD captures the relation between these users by finding the strength of each concept of the movies the users like and then we can draw a simple comparison. Hence it may not be the case that the similarity is 0 if their dot product similarity is 0. Figure 20 illustrates the same. Clearly, there is a some match in the graphs. So this means there is some relation in the concepts of the movies user 1 and user 2 prefer.

9.2 Add movies to see how they map to concepts

Suppose we add 2 new movies, *movie1* and *movie2* to the dataset with their ratings given by all 364 users. Suppose the two movie rating vectors are orthogonal to each other, then by simple dot product similarity, their similarity will be 0. This means by just the ratings of the users, we have to say the movies are not related at all. But this is not the case. SVD captures the relation between these movies by using the user to concept matrix and the ratings given by each user to the movies. Hence it may not be the case that the similarity between the 2 movies is 0 if their dot product similarity is 0. Figure 21 illustrates the same. Clearly, there is a some match in the graphs. So this means there is some relation in the concepts of the movie 1 and movie 2.