**Recommendation Systems: Collaborative Filtering**

Samhitha D, PES1201701274

Varsha C, PES1201701387

Anish Poddar, PES1201700269

Ananthasharma CN, PES1201802361

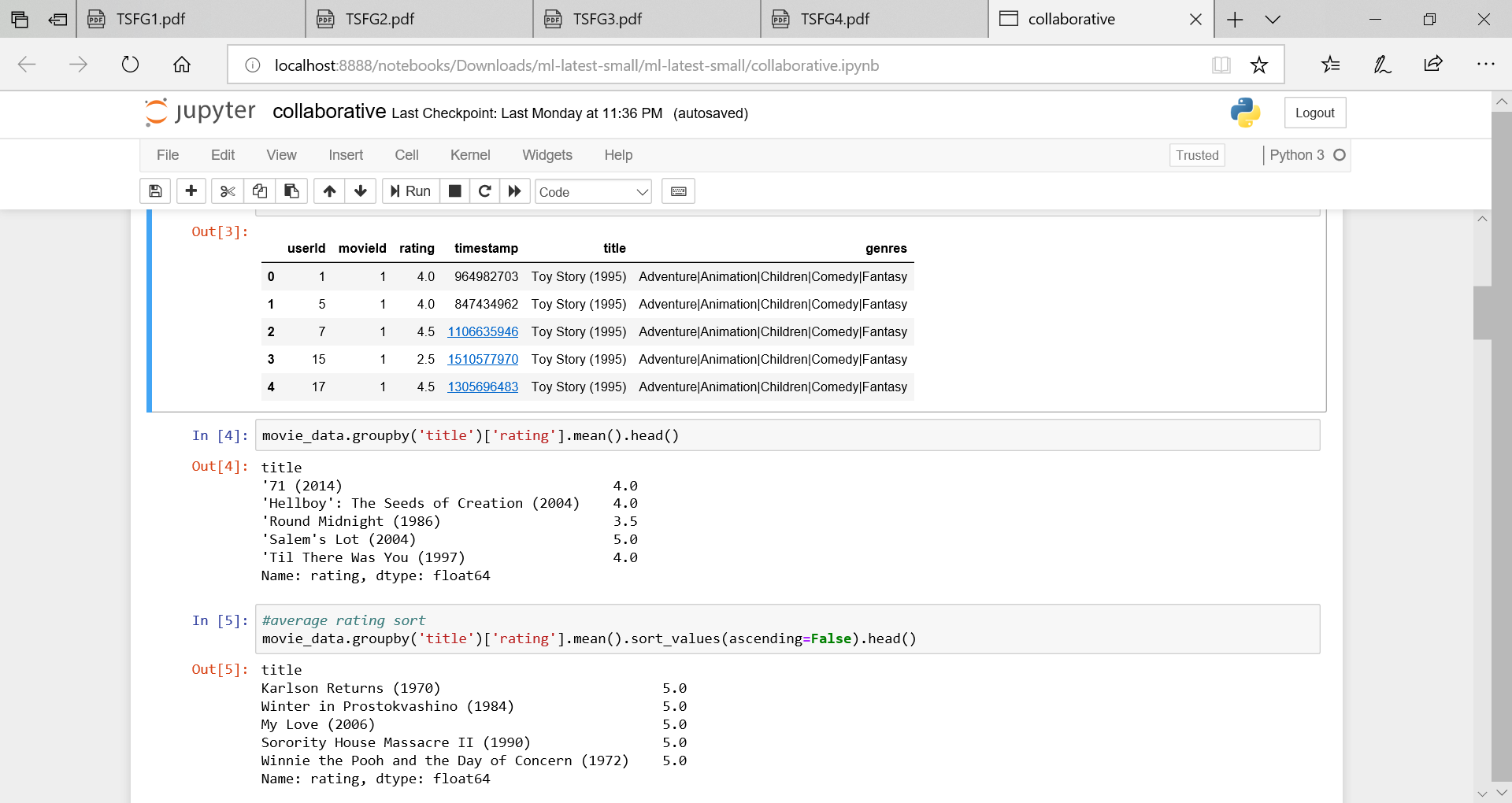
Aditya M.Nair, PES1201802187

A Recommender System employs a statistical algorithm that seeks to predict users' ratings for a particular entity, based on the similarity between the entities or similarity between the users that previously rated those entities. The intuition is that similar types of users are likely to have similar ratings for a set of entities. There are 2 types of recommendation systems- content-based and collaborative based.

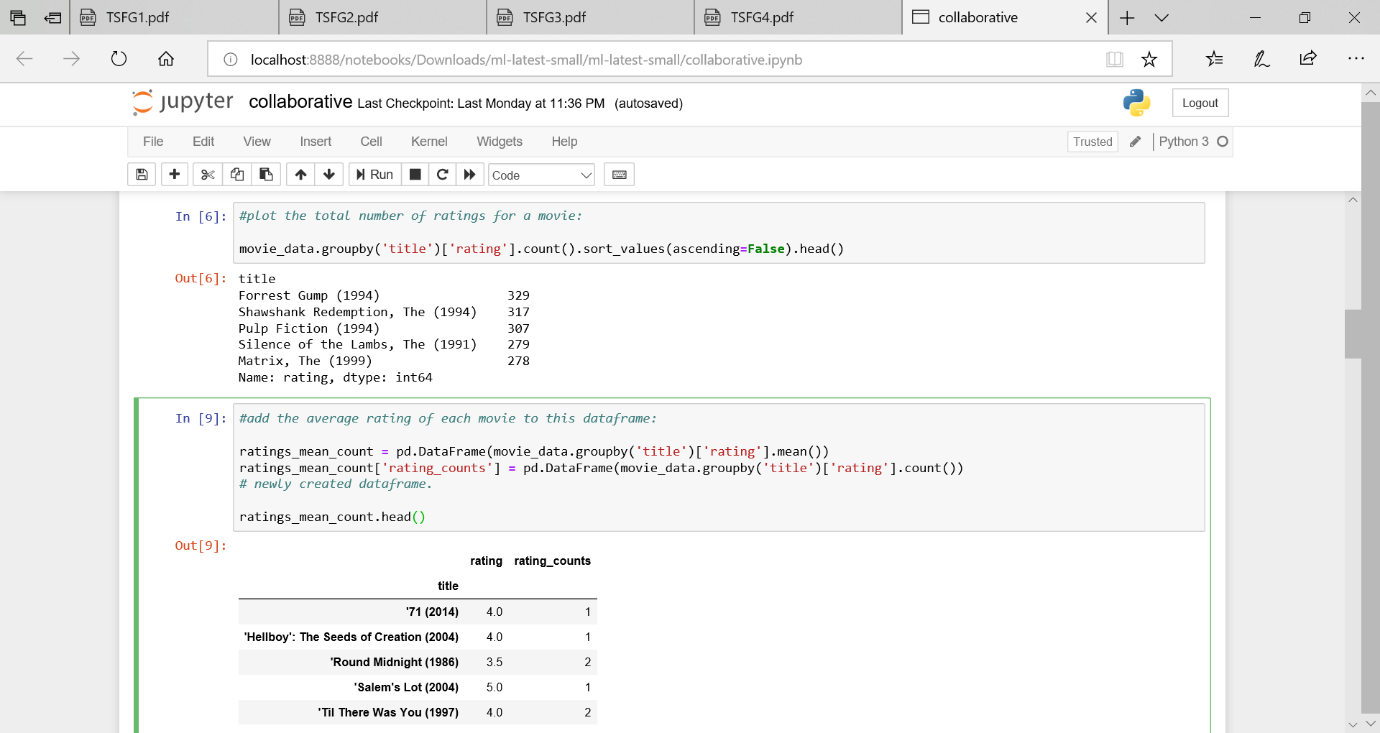
**Collaborative Filtering**

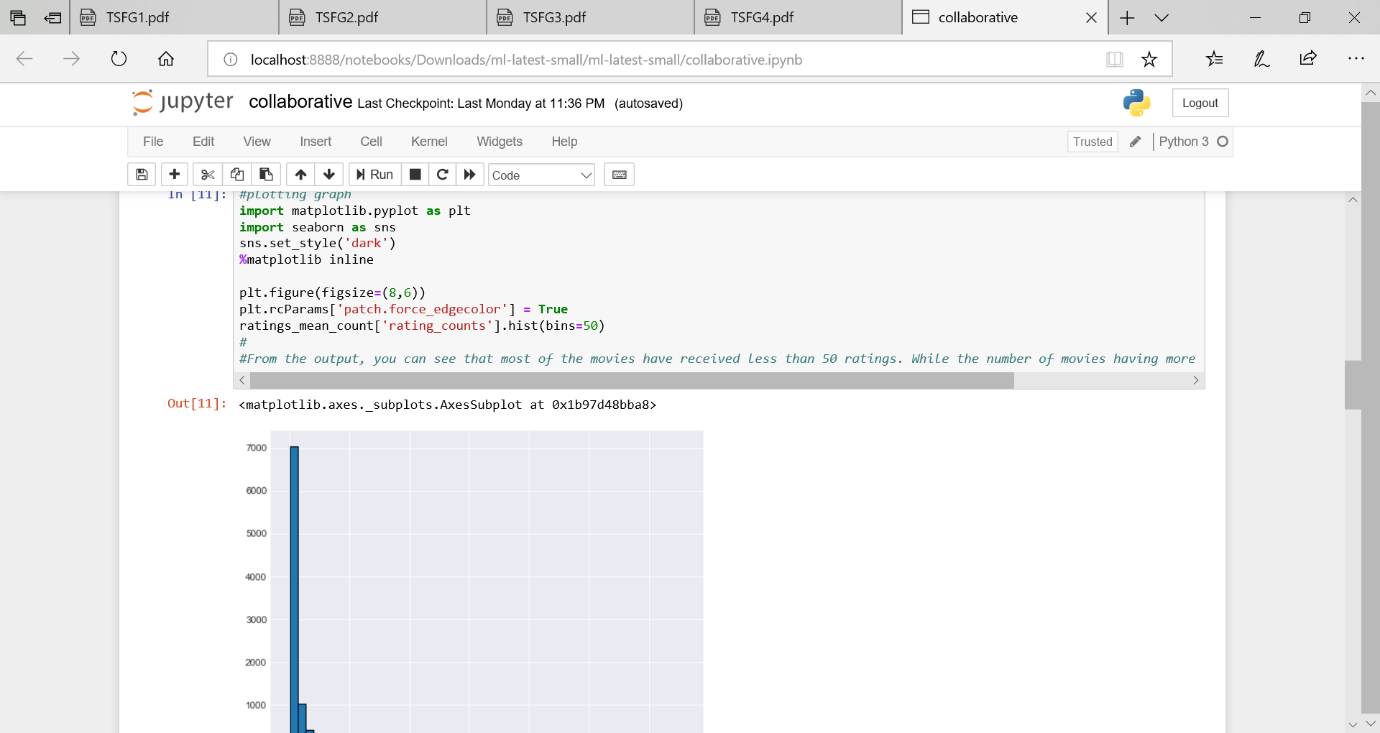
Collaborative filtering leverages the power of the crowd. The intuition behind collaborative filtering is that if a user A likes products X and Y, and if another user B likes product X, there is a fair bit of chance that he will like the product Y as well.

Here, we are taking the example of a movie recommender system. Suppose a huge number of users have assigned the same ratings to movies X and Y. A new user comes who has assigned the same rating to movie X but hasn't watched movie Y yet. Collaborative filtering system will recommend him the movie Y.

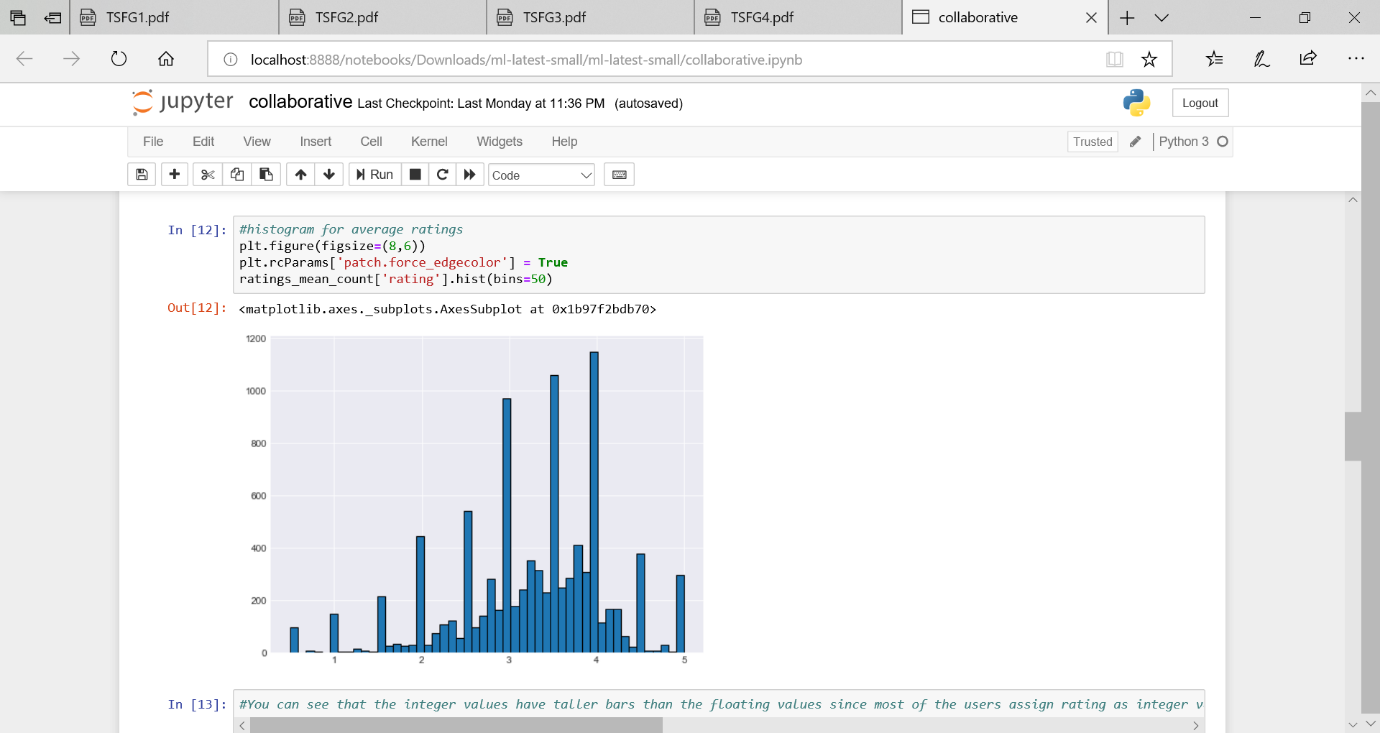


The "ratings.csv" file contains the userId, movieId, ratings, and timestamp attributes. Each row in the dataset corresponds to one rating. The userId column contains the ID of the user who left the rating. The movieId column contains the Id of the movie, the rating column contains the rating left by the user. Ratings can have values between 1 and 5. And finally, the timestamp refers to the time at which the user left the rating.But it contains the IDs of the movies but not their titles. We'll need movie names for the movies we're recommending. The movie names are stored in the "movies.csv" file.

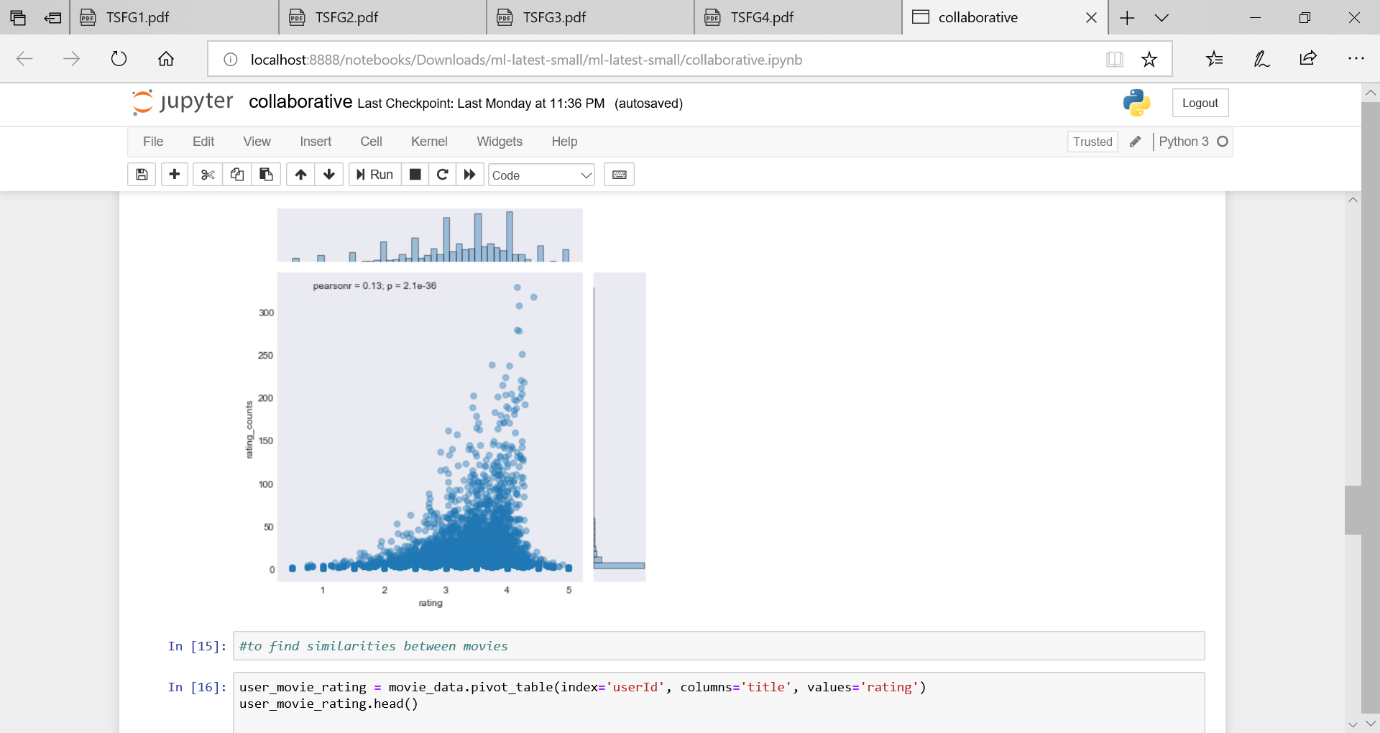
We group the dataset by the title of the movie and then calculate the mean of the rating for each movie. We will then display the first five movies along with their average rating using the head() method and plot



Histogram for average ratings

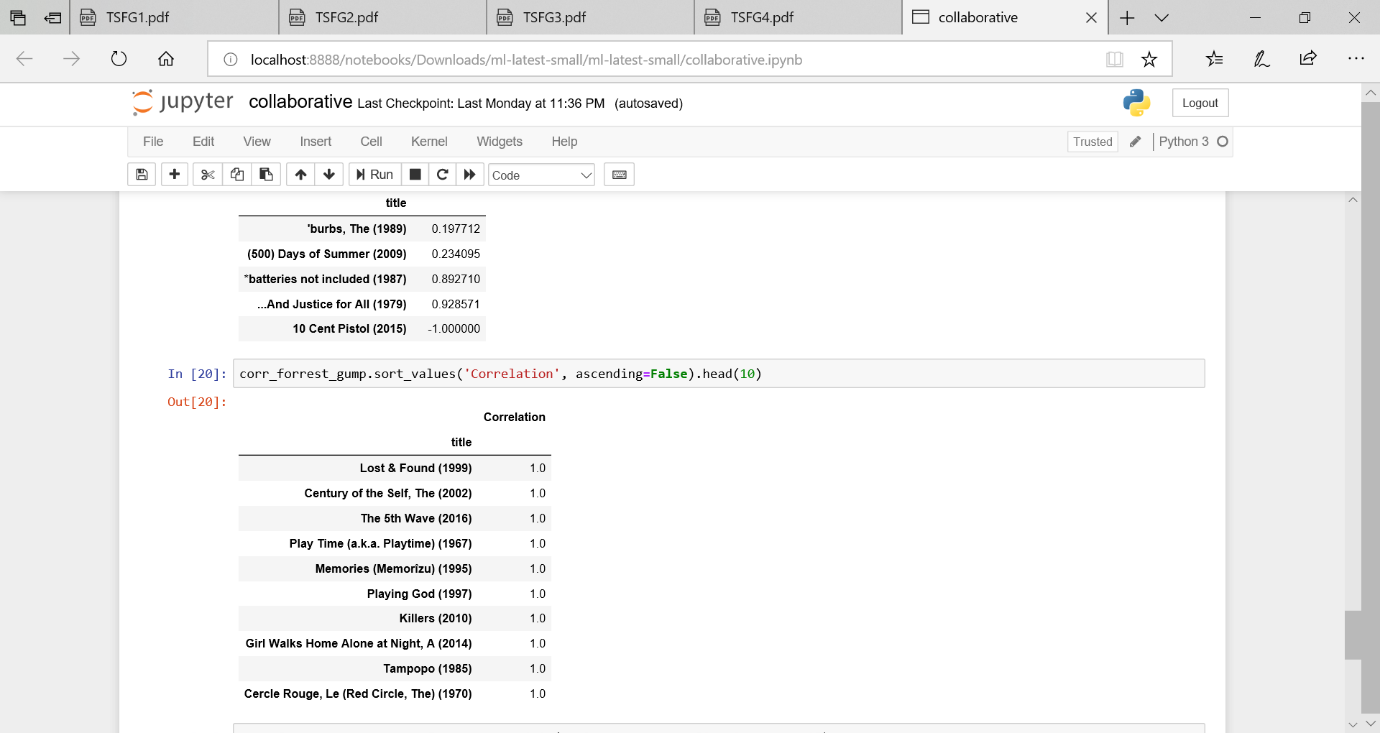


From the above output, we can see that most of the movies have received less than 50 ratings. While the number of movies having more than 100 ratings is very low.

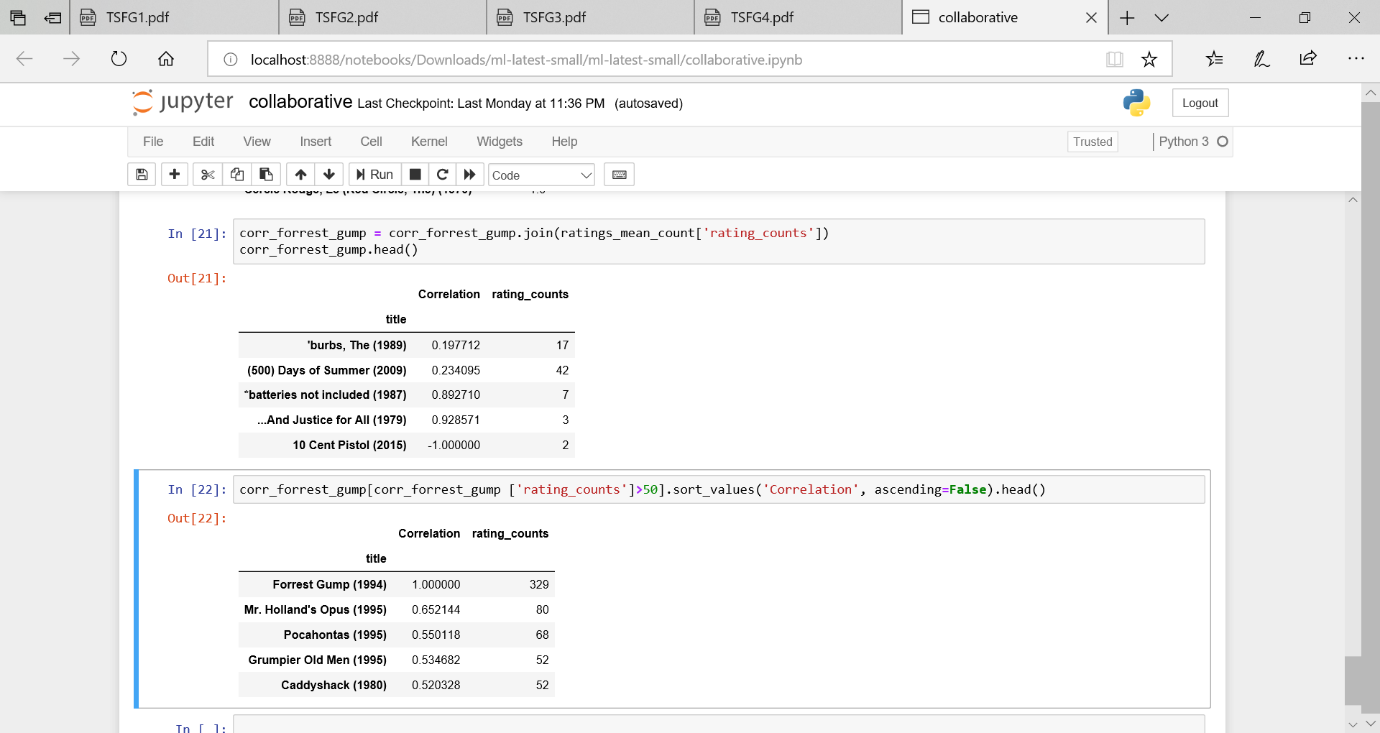


The above graph shows us that, in general, movies with higher average ratings actually have more number of ratings, compared with movies that have lower average ratings.

We know that each column contains all the user ratings for a particular movie. Let's find all the user ratings for the movie "Forrest Gump (1994)" and find the movies similar to it. We chose this movie since it has the highest number of ratings and we want to find the correlation between movies that have a higher number of ratings.



From the output you can see that the movies that have high correlation with "Forrest Gump (1994)" are not very well known. This shows that correlation alone is not a good metric for similarity because there can be a user who watched '"Forest Gump (1994)" and only one other movie and rated both of them as 5. This is why we added "rating\_counts" column. Let's now filter movies correlated to "Forest Gump (1994)", that have more than 50 ratings.



Now we can see from the output the movies that are highly correlated with "Forrest Gump (1994)". The movies in the list are some of the most famous movies Hollywood movies, and since "Forest Gump (1994)" is also a very famous movie, there is a high chance that these movies are correlated.