Movie Recommendation System

Kregg Jackson

Business Context

- This project was built for a app named Letterboxd and is designed to make personalized movie recommendations to the user based off their ratings of movies
- When done correctly the system will support and improve the quality of the decisions users make while searching for movies

Movie Review Data

The movies dataframe has 9,700 rows of three columns :

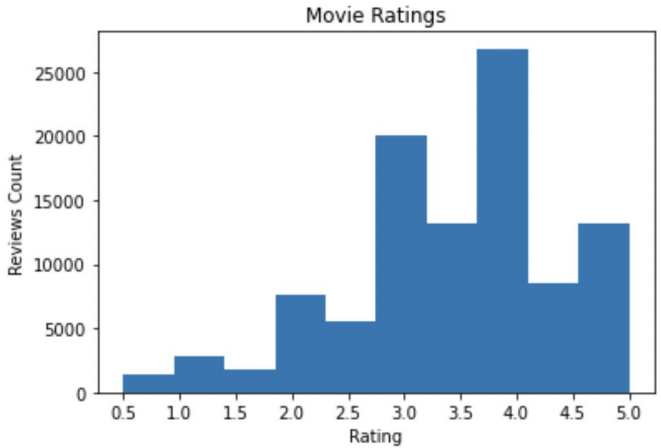
- ‘movieId’
- ‘title’
- ‘genre’

The ratings dataframe originally had over 100,000 rows of reviews with four columns of:

- ‘userId’
- ‘movieId’
- ‘rating’
- ‘timestamp’

Ratings Scale

- After watching movies users rate them on a scale from 0.5 - 5 stars
- User ratings are made on a 5-star scale, with half-star increments
- The ratings are skewed to the left



How is success determined?

- For my success determinant I am using a combination of RMSE and MAE
- Root mean squared error (RMSE)
    - standard deviation of the residuals (prediction errors)
- Mean Absolute Error (MAE)
    - average absolute error between actual and predicted values
- RMSE is sensitive to outliers and MAE takes more of an overall average approach

Process Steps

My next steps were to:

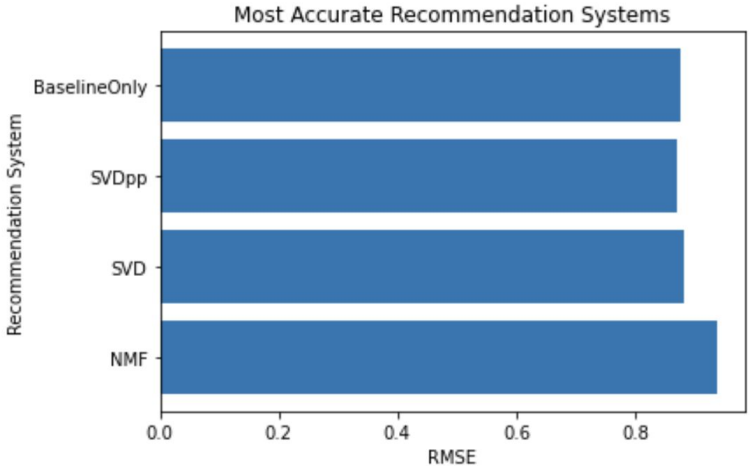
- Build and tune the recommendation systems
- Find the best parameters for the lowest Root means squared error (RMSE)
- Obtain a prediction for a specific user for a particular item
- Create a function that will return the top 5 movie recommendations for a user

The best performing systems were:

- SVD ++
- SVD
- BaselineOnly

Results

- The SVD system was able to predict rating with a RMSE score of 0.88 and the MAE is at 0.68
- The BaselineOnly recommendation system predicted rating with a RMSE of 0.87 and a MAE score of 0.66
- The best performing system was the SVD ++, and was able to predict user ratings with a RMSE of 0.86 and the MAE is 0.65



SVD ++

- Best performing model uses collaborative filtering

- User ratings is the input factor because explicit feedback is the most convenient feature when making recommendations

Conclusive Evaluation

- The best performing recommendation system is the SVD ++ iteration
- SVD ++ system predicts user preference within 1 point of the actual rating
- The movie suggestion website will be able to take users previous ratings and recommend movies the user has not seen

Future Improvements

- Have recommendation system not only suggest singular movies but movie genres
- Running gridsearch on SVD++ was very taxing on my machine's CPU with more time I could test more parameters to see if I could lower RMSE and MAE
- Have users rate the recommended movies to know if the model recommendations are successful

Thank You

Email: kreggthegoat@gmail.com

Github: @kreggthegoat