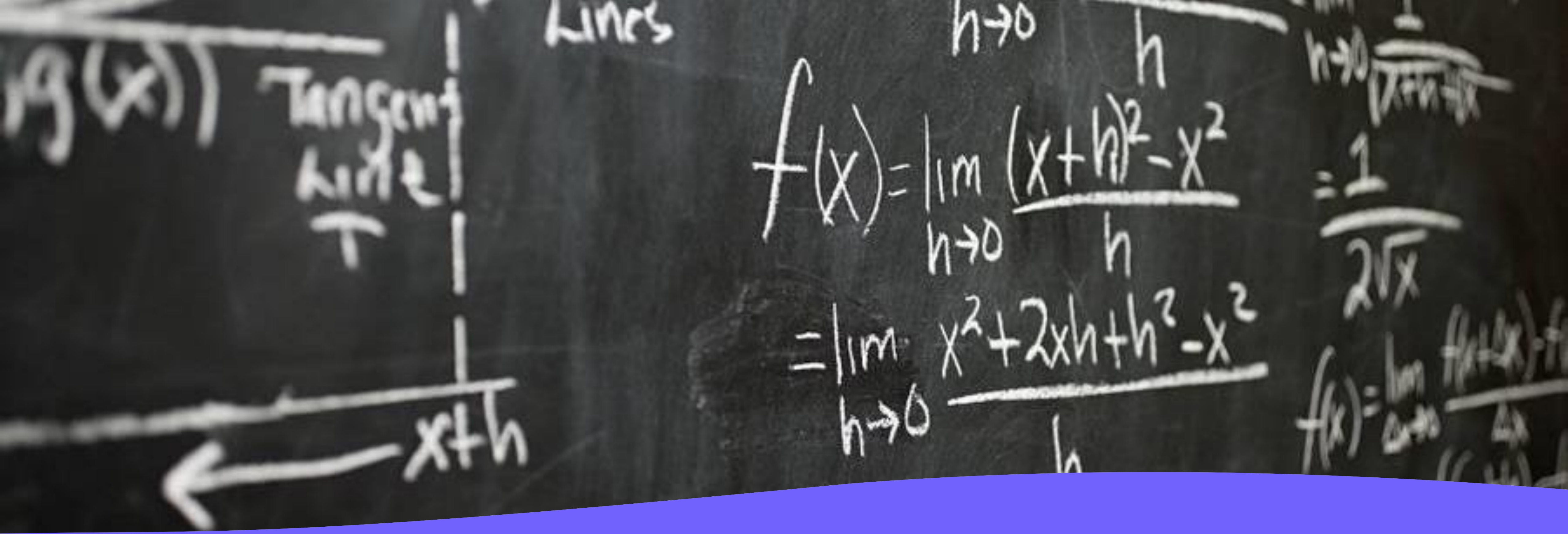


# Recommender Systems

Assignment 3 – Practical Data Science with Python (COSC2670)
2024

Campbell Timms
(s3720784)

# Outline

Task 1: kNN-based Collaborative Filtering

Task 2: Matrix Factorization-based Recommendation

Task 3: Ranking-based Evaluation and Comparison

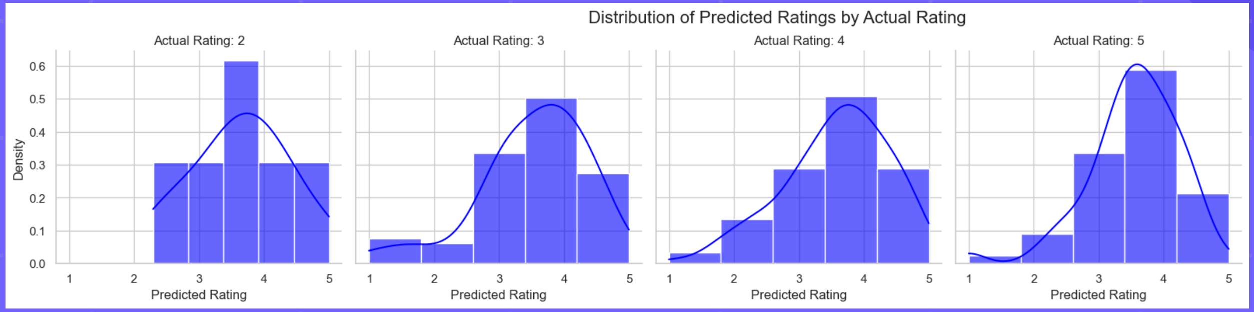
# Task 1 : kNN-based Collaborative Filtering

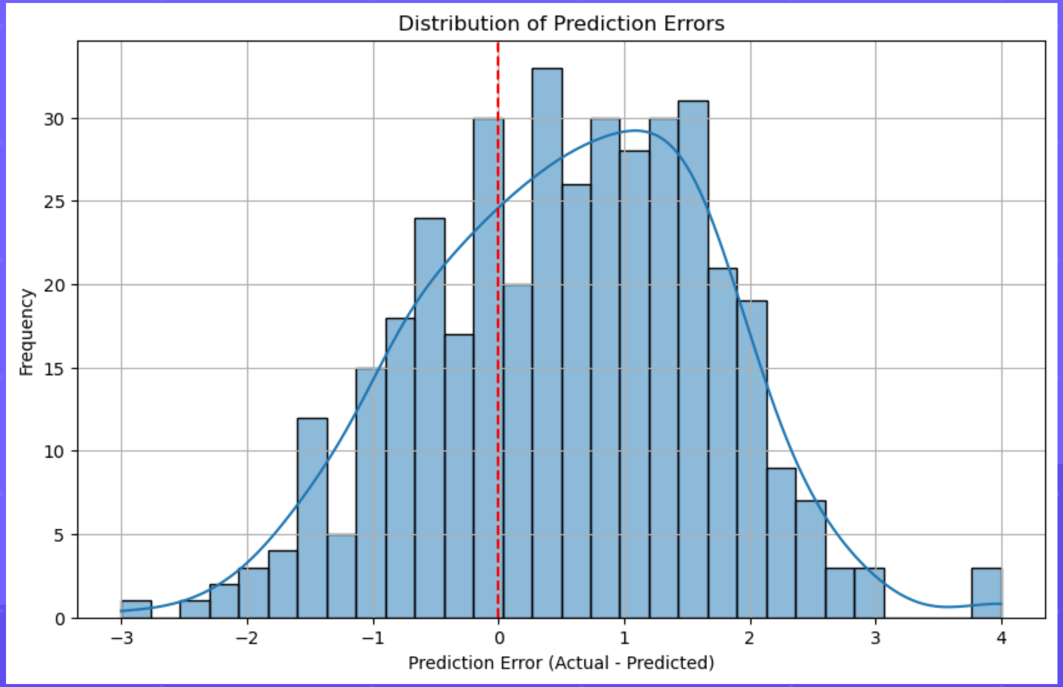
1. kNNCF User – User based collaborative filtering model involves generating predicted ratings for a user based on the nearest k other similar user's ratings.
2. Similarity was determined using either Cosine similarity or by their Euclidean distances.
3. A set of k values was tested to determine the best value for k.
4. RMSE (Root mean squared error) was used to measure the accuracy of the model.

Table 1: The average RMSE of n = 50 users at different k values. k of 25 using the Euclidean distance similarity was determined to be the best.

| k | Ave Cosine RMSE | Ave Euclidean RMSE |
|---|---|---|
| 5 | 1.415 | 1.333 |
| 10 | 1.383 | 1.333 |
| 15 | 1.374 | 1.331 |
| 20 | 1.365 | 1.329 |
| 25 | 1.359 | 1.326 |

# Task 1: kNN Collaborative Filtering results



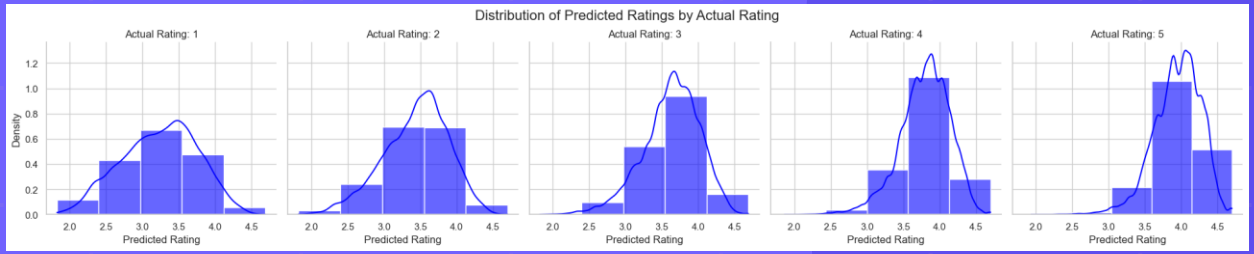

# Task 2: Matrix Factorization-based Recommendation

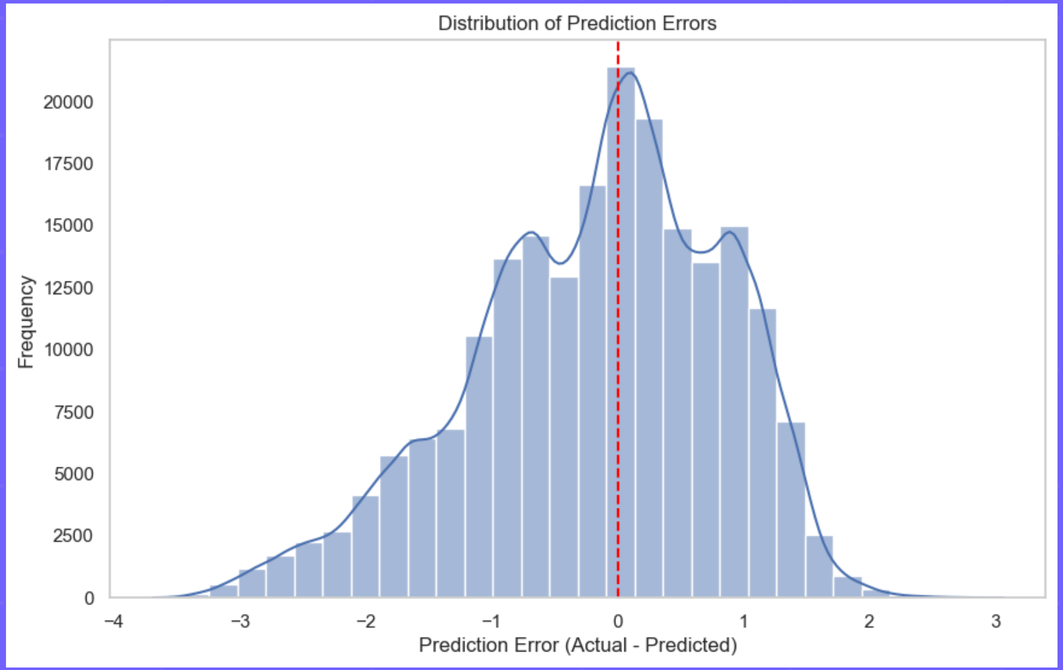
1. Matrix factorization breaks down the user-item matrix into the product of two lower-dimensional matrices, capturing user information and its latent factors alongside item information and its latent factors.
2. 5 Movies were chosen, and all user's predicted ratings were found using the following.
3. SVD (single value decomposition) was the matrix factorization technique of choice.
4. SVD was improved upon by using SVD++ with user demographic information stored as latent factors.

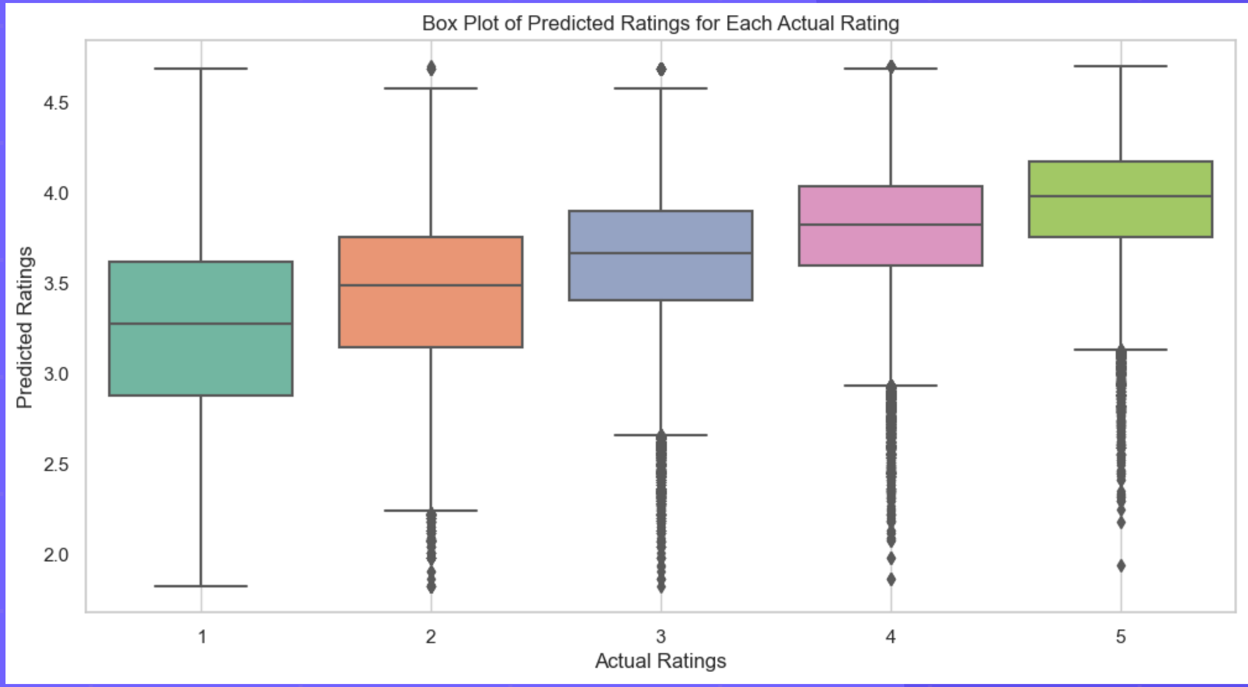
Table 2: RMSE was found using all users predicted ratings from a random 5 movies. SVD++ showed a small improvement over SVD.

| Factorization Technique | RMSE |
|---|---|
| SVD | 1.0435 |
| SVD ++ | 1.0171 |

# Task 2: Matrix Factorization-based Recommendation results



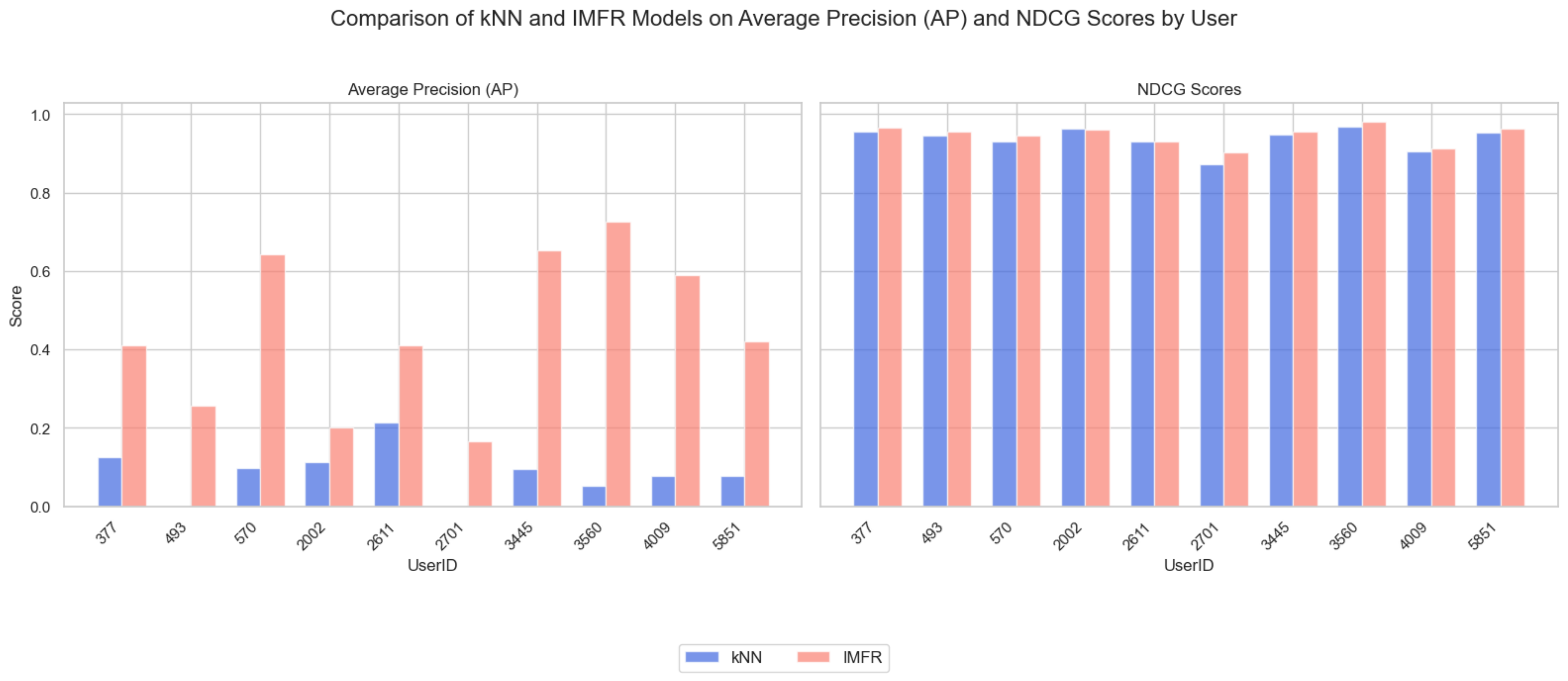
# Task 3: Ranking-based Evaluation and Comparison

1. Complete a ranking based evaluation on the two models produced in tasks 1 and 2.
2. 10 random users who have rated more than 100 movies were selected and recommended the top 20 movies for each of them.
3. Average Precision and Normalised Discounted Cumulative Gain was used for the ranking evaluation.
4. For AP if they movie recommended has been rated as a 4 or above it is seen as a success.
5. For NDCG the actual ratings were compared to the predicted ratings of each model.

Table 3: The RMSE for each of the models used from task 1 and task 2. IMFR is shown to have a much lower error and more accurate predictions.

| Model | RMSE |
|---|---|
| kNNCF | 1.4344 |
| IMFR | 1.1078 |

# Task 3: Ranking-based Evaluation and Comparison Results



# Discussion

## Findings

- IMFR was found to be more accurate and scales better than kNNCF. kNNCF doesn't handle sparce data sets as well. Our data set was shown to be 89.59% sparce.
- kNNCF was faster initially but as the data set grew it became much slower to run than IMFR.
- IMFR with user information can capture more latent features and make more personal recommendation for each user. kNNCF frequently recommends only popular films.
- kNNCF is based on the user's previous ratings and is only compared to other users with similar ratings. In terms of recommendation more highly rated films are predicted to have a high rating for the user and less variety is shown. Including more information about movies and users may improve performance.
- IMFR can also adjust the importance of demographic information. A weighting of 3 was used for user information which showed more variety.
- IMFR can also be improved. Greater parameter tuning of the latent user features and the addition of movie information could improve performance.

Table 4: The top 5 movie recommendations of the IMFR model includes several highly rated films but the position changes based on user. This is an improvement on the kNN model which isn't personalised as much.

| UserID | Recommendation 1 | Recommendation 2 | Recommendation 3 | Recommendation 4 | Recommendation 5 |
|---|---|---|---|---|---|
| 377 | Sanjuro (1962) | Shawshank Redemption, The (1994) | Pather Panchali (1955) | Schindler's List (1993) | Godfather, The (1972) |
| 493 | Sanjuro (1962) | Rear Window (1954) | Godfather, The (1972) | Shawshank Redemption, The (1994) | Seven Samurai (The Magnificent Seven) (Shichin... |
| 570 | Godfather, The (1972) | Rear Window (1954) | Sanjuro (1962) | Schindler's List (1993) | Shawshank Redemption, The (1994) |
| 2002 | Shawshank Redemption, The (1994) | Schindler's List (1993) | Sanjuro (1962) | Silence of the Lambs, The (1991) | Life Is Beautiful (La Vita è bella) (1997) |
| 2611 | Seven Samurai (The Magnificent Seven) (Shichin... | Schindler's List (1993) | Shawshank Redemption, The (1994) | Sanjuro (1962) | Close Shave, A (1995) |
| 2701 | Schindler's List (1993) | Sanjuro (1962) | To Kill a Mockingbird (1962) | Shawshank Redemption, The (1994) | Casablanca (1942) |
| 3445 | Shawshank Redemption, The (1994) | Godfather, The (1972) | Sanjuro (1962) | Schindler's List (1993) | Star Wars: Episode IV - A New Hope (1977) |
| 3560 | Shawshank Redemption, The (1994) | Sanjuro (1962) | Star Wars: Episode IV - A New Hope (1977) | Raiders of the Lost Ark (1981) | Godfather, The (1972) |
| 4009 | Schindler's List (1993) | Shawshank Redemption, The (1994) | Sanjuro (1962) | Wrong Trousers, The (1993) | Star Wars: Episode IV - A New Hope (1977) |
| 5851 | Sanjuro (1962) | Star Wars: Episode IV - A New Hope (1977) | Close Shave, A (1995) | Grand Day Out, A (1992) | Star Wars: Episode V - The Empire Strikes Back... |

# REFERENCES


1. Anwar, Aqeel. "What Is Average Precision in Object Detection & Localization Algorithms and How to Calculate It?" Medium, May 13, 2022. https://towardsdatascience.com/what-is-average-precision-in-object-detection-localization-algorithms-and-how-to-calculate-it-3f330efe697b.
2. Chen, Denise. "Recommendation System – Matrix Factorization." Medium, July 9, 2020. https://towardsdatascience.com/recommendation-system-matrix-factorization-d61978660b4b.
3. Hug, Nicolas. "Home." Surprise. Accessed October 31, 2024. https://surpriselib.com/.
4. Liao, Kevin. "Prototyping a Recommender System Step by Step Part 2: Alternating Least Square (ALS) Matrix…." Medium, November 19, 2018. https://towardsdatascience.com/prototyping-a-recommender-system-step-by-step-part-2-alternating-least-square-als-matrix-4a76c58714a1.
5. Ranjan, Mukul. "Understanding NDCG." *Medium* (blog), May 2, 2024. https://medium.com/@mukulranjan/understanding-ndcg-885656321b3b.
6. scikit-learn. "Average_precision_score." Accessed November 1, 2024. https://scikit-learn/stable/modules/generated/sklearn.metrics.average_precision_score.html.
7. scikit-learn. "Ndcg_score." Accessed November 1, 2024. https://scikit-learn/stable/modules/generated/sklearn.metrics.ndcg_score.html.
8. scikit-learn. "NMF." Accessed October 31, 2024. https://scikit-learn/stable/modules/generated/sklearn.decomposition.NMF.html.
9. scikit-learn. "Sklearn.Decomposition." Accessed October 31, 2024. https://scikit-learn/stable/api/sklearn.decomposition.html.
10. scikit-learn. "TruncatedSVD." Accessed October 31, 2024. https://scikit-learn/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html.