# SVD-based incremental approaches for recommender systems

CSDS 313 - Introduction to Data Analysis

Group Member: Lei Ruan, Sohan Bhawtankar

# Index

1. Background
2. Analysis
3. Improvement
4. Experiment Result
5. Conclusion
6. Future Work

Paper: *Xun Zhou, Jing He, Guangyan Huang, Yanchun Zhang, SVD-based incremental approaches for recommender systems, Journal of Computer and System Sciences, Volume 81, Issue 4, 2015,*

Link: *https://doi.org/10.1016/j.jcss.2014.11.016*

Journal of Computer and System Sciences 81 (2015) 717–733

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

ELSEVIER

## SVD-based incremental approaches for recommender systems

CrossMark

Xun Zhou [a,d], Jing He [b], Guangyan Huang [c], Yanchun Zhang [b,d,*]

[a] *UCAS-VU Joint Lab for Social Computing and E-Health Research, University of Chinese Academy of Sciences, Beijing, China*
[b] *Centre for Applied Informatics, College of Engineering and Science, Victoria University, Australia*
[c] *School of Information Technology, Deakin University, Australia*
[d] *School of Computer Science, Fudan University, Shanghai, China*

ARTICLE INFO

*Article history:*
Received 16 July 2014
Received in revised form 30 October 2014
Accepted 6 November 2014
Available online 16 December 2014

*Keywords:*
Singular value decomposition
Incremental algorithm
Recommender system
Experimental evaluation

ABSTRACT

Due to the serious information overload problem on the Internet, recommender systems have emerged as an important tool for recommending more useful information to users by providing personalized services for individual users. However, in the "big data" era, recommender systems face significant challenges, such as how to process massive data efficiently and accurately. In this paper we propose an incremental algorithm based on singular value decomposition (SVD) with good scalability, which combines the Incremental SVD algorithm with the Approximating the Singular Value Decomposition (ApproSVD) algorithm, called the Incremental ApproSVD. Furthermore, strict error analysis demonstrates the effectiveness of the performance of our Incremental ApproSVD algorithm. We then present an empirical study to compare the prediction accuracy and running time between our Incremental ApproSVD algorithm and the Incremental SVD algorithm on the MovieLens dataset and Flixster dataset. The experimental results demonstrate that our proposed method outperforms its counterparts.



### 1. Introduction

#### *1.1. Background*

With the popularity of the Internet and advances in information technology, information from websites tends to be too general and people require more personalized information. In order to meet users' demand for personalized services, personalized recommender systems are a powerful tool to solve the information overload problem. Collaborative filtering is one of the most important techniques used in recommender systems. Its principle is to recommend likely new information to an active user by considering other similar users' interests. It is based on the assumption that if two users have similar interests then the two users will probably share the same information. The advantages of collaborative filtering are as follows: first, it is independent of the contents of recommended items; second, it can be closely integrated with social networks; third, it has good accuracy in terms of recommendations.

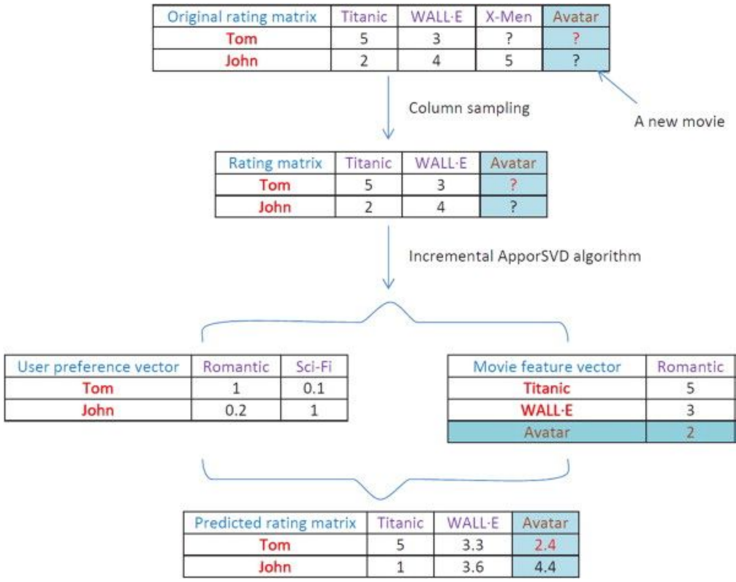
The common challenge of collaborative filtering and other types of recommender systems is how to deal with massive data to make accurate recommendations. There are three difficulties [1]: (1) the huge amount of data, which requires the algorithm to respond quickly; (2) the sparsity of data, the ratings provided by the users or information which can be used to indicate interests are actually very sparse, compared with the large number of users and items in a recommender system; (3) the dynamic nature of data, which requires the algorithm to update quickly and accurately. The recommender system

* Corresponding author at: Centre for Applied Informatics, College of Engineering and Science, Victoria University, Australia.
*E-mail address:* yzhangvu@gmail.com (Y. Zhang).

http://dx.doi.org/10.1016/j.jcss.2014.11.016

# 1. Background

- Incremental ApproSVD




| Original rating matrix | Titanic | WALL·E | X-Men | Avatar |
|---|---|---|---|---|
| Tom | 5 | 3 | ? | ? |
| John | 2 | 4 | 5 | ? |

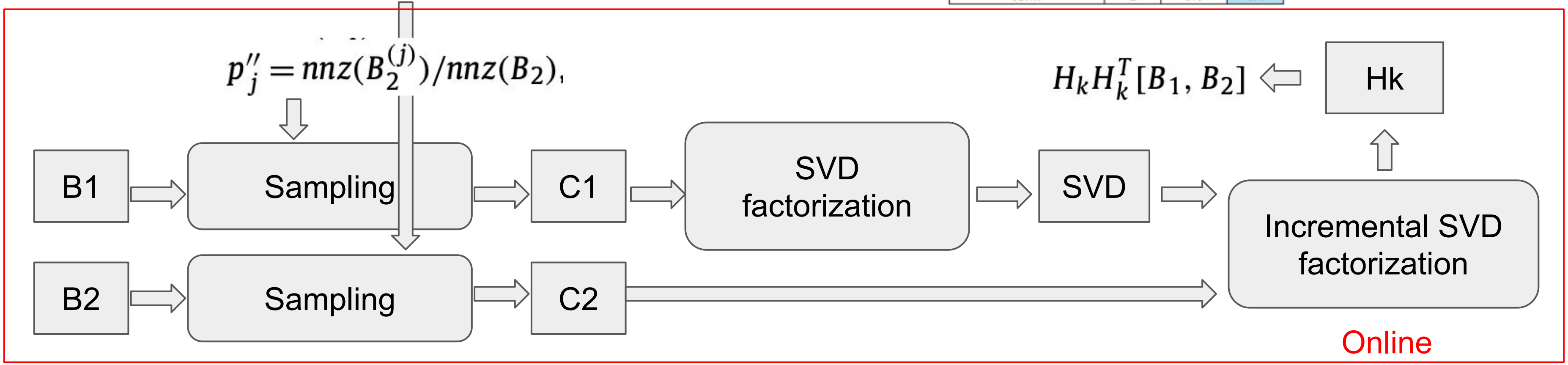
| Rating matrix | Titanic | WALL·E | Avatar |
|---|---|---|---|
| Tom | 5 | 3 | ? |
| John | 2 | 4 | ? |

| User preference vector | Romantic | Sci-Fi |
|---|---|---|
| Tom | 1 | 0.1 |
| John | 0.2 | 1 |

| Movie feature vector | Romantic | Sci-Fi |
|---|---|---|
| Titanic | 5 | 0 |
| WALL·E | 3 | 3 |
| Avatar | 2 | 4 |

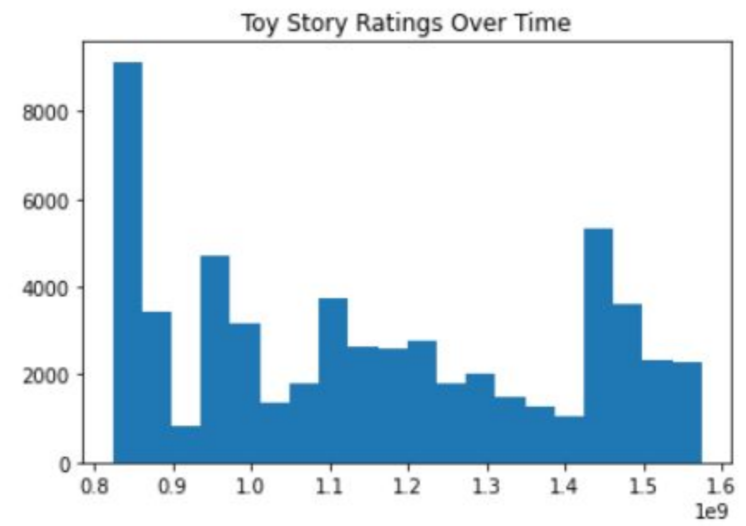
| Predicted rating matrix | Titanic | WALL·E | Avatar |
|---|---|---|---|
| Tom | 5 | 3.3 | 2.4 |
| John | 1 | 3.6 | 4.4 |

Offline $p'_i = nnz(B_1^{(i)})/nnz(B_1)$



# 2. Analysis

- Undesired **blockbuster effects**
  - Not in every domain the algorithm that achieves the lowest RMSE leads to the best results with respect to the business goals
  - 3 aspects of undesired blockbuster effects that may affect the algorithm performance:
    - **Popularity Bias**
    - **Concentration Biases**
    - **Reinforcement Effect**
- Incomplete information
  - SVD factorization drops some information
  - Time information is not utilized
- Threshold

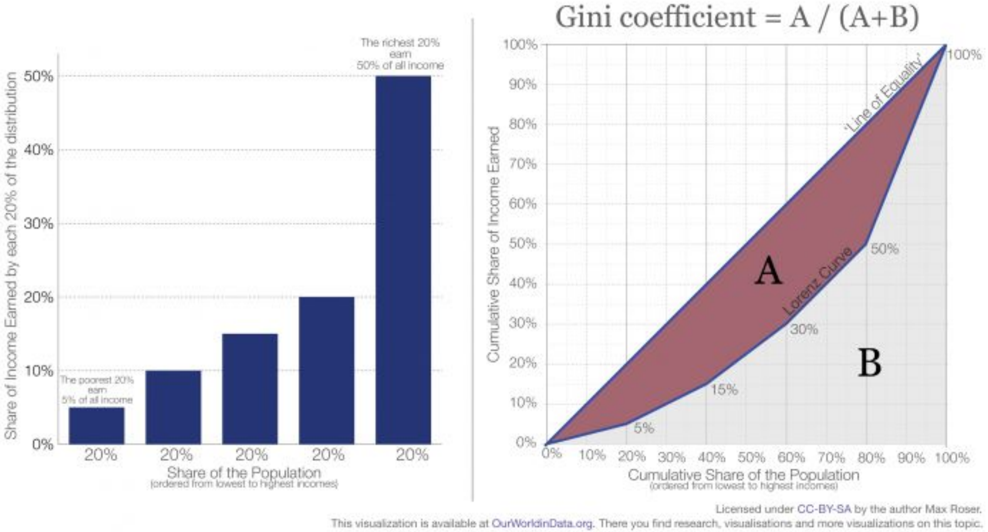


# 2. Analysis

- **Whether the new algorithm increased the Popularity Biases**
  - **Experiment 1**: validate that artificial popularity bias can enhance the accuracy
    - Method: filter those items rated by less than ***p*** users (post-processing)
    - Metrics: Precision & Recall (TS)
    - Analysis: if the effect is validated
      - Compare popularity bias of SVD and ApproSVD
  - **Experiment 2**: calculate the degree of popularity bias
    - Method:
      - Suppose we recommend 10 items for each user
      - Use t-test to compare the degree of popularity bias
    - Metrics:
      - Average rating
      - # of ratings
    - Analysis: if the popularity bias for Incremental ApproSVD is higher, this can be a caveat

# 2. Analysis

- **Whether the new algorithm increased the Concentration Biases**
  - Method: Count the times each element is recommended to a user
  - Metrics: Gini Coefficient
  - Analyze: Compare the Gini coefficients for Incremental SVD and Incremental ApproSVD



# 2. Analysis

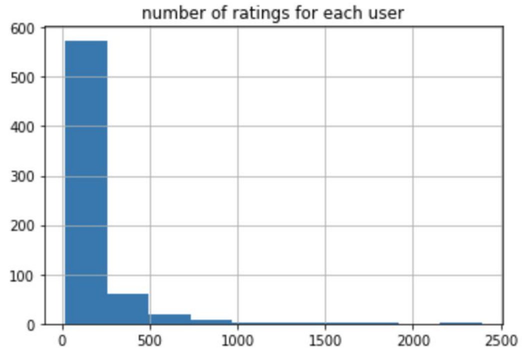
- **Whether the new algorithm increased the Popularity Reinforcement Effect**
  - Method: Emulate evolution of the database
  - Metrics: Gini Coefficient and the number of recommended items w.r.t. iteration
  - Analyze: Compare the Popularity Reinforcement Effect

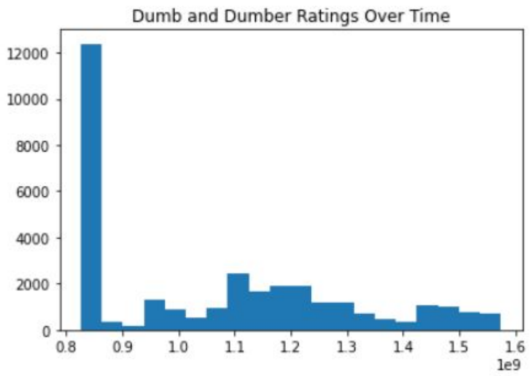
# 3. Improvement

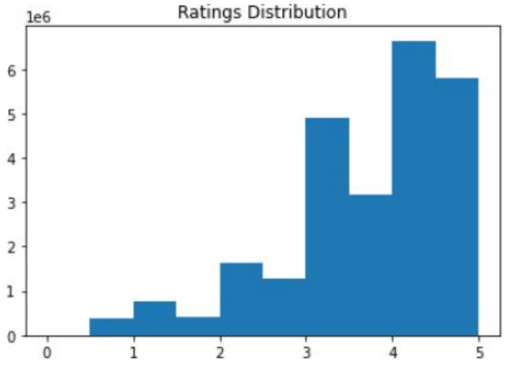
- **Possible countermeasures:  Post-processing**
  - Undesired blockbuster effects → Adjusting relevance scores based on popularity biases
  - Plan 1: add a user-irrelevant score to the prediction rating score
  - Plan 2: add a **user-related** score to the prediction rating score
    - Rules:
      - The added score should be relatively small compared to the rating score and related to user's original rating bias
    - Methodologies:
      - Use **L-2 regularization** to limit the score to be relatively small
      - Use **gradient descent** approach to optimize the personalized bias
    - Metrics:
      - **RMSE, MSE**, and **Precision & Recall** (TS/All)
      - Popularity Biases: **Average popularity biases**
      - Concentration Biases: **Gini coefficients**

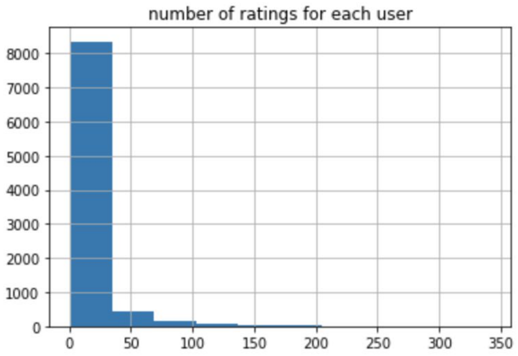
# 4. Experiment

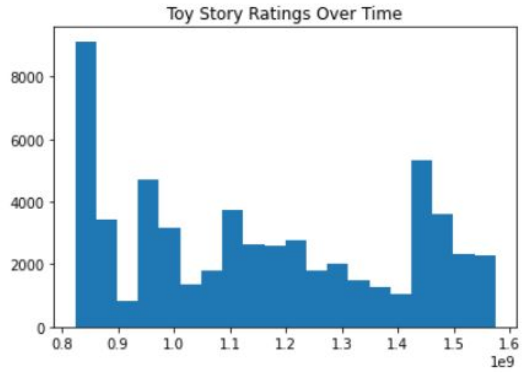
- Dataset: MovieLens 100K
  - N_user: 943
  - N_item: 1682
  - # of ratings: 100k
  - Sparsity: about 6%



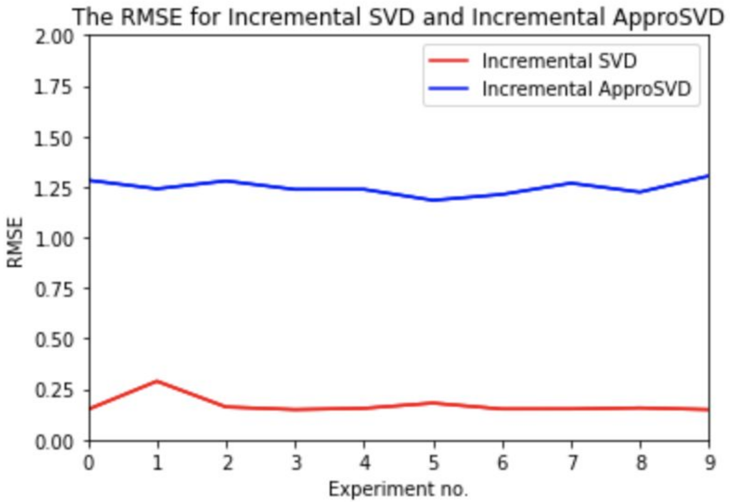



# 4. Experiment Result

- **Experiment 1**: Reproduction of Incremental Appro-SVD algorithm
  - Design
    - Metric: **RMSE**
    - Method: **5-fold Cross validation**
  - Result
    - The accuracy is significantly worse than normal Incremental SVD algorithm
    - The RMSE for the new model is 5 times of original model
    - The rise in time performance cannot make for the degradation in accuracy

| | 0 | 1 |
|---|---|---|
| 0 | 0.149504 | 1.282847 |
| 1 | 0.289194 | 1.240438 |
| 2 | 0.161929 | 1.279130 |
| 3 | 0.148917 | 1.237297 |
| 4 | 0.155333 | 1.237465 |
| 5 | 0.181060 | 1.183592 |
| 6 | 0.151914 | 1.212559 |
| 7 | 0.152003 | 1.268118 |
| 8 | 0.157498 | 1.224162 |
| 9 | 0.148847 | 1.304091 |



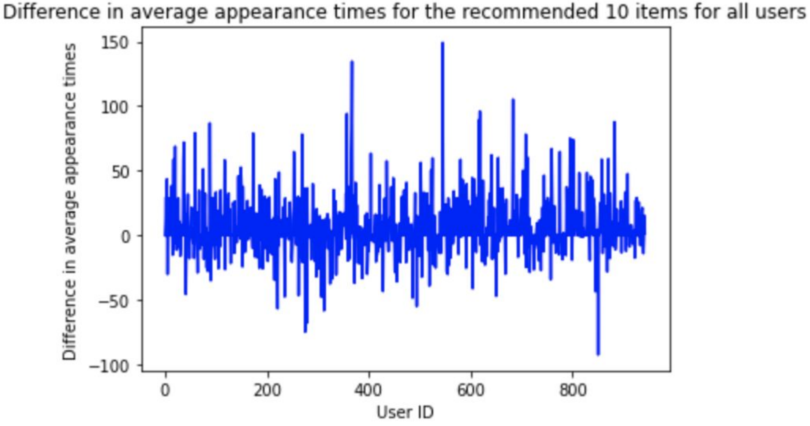
# 4. Experiment Result

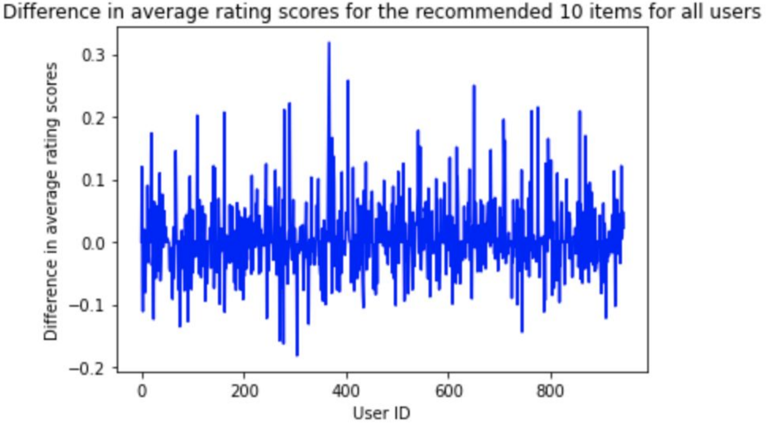
- **Experiment 2**: Degree of popularity bias
  - Design
    - Methodology:
      - Recommend 10 new items for each user
      - Compare the popularity bias for Incremental Appro-SVD and Incremental SVD
    - Metrics:
      - Average rating for the top 10 items of each user
      - Number of ratings for the top 10 items of each user
    - Hyperparameter Analysis
      - Number of columns selected: **n1, n2**
      - Number of columns sampled in the Incremental Appro-SVD algorithm: **c1, c2**
      - Change the approximation dimension: **k**
      - Number of items recommended: **q**

# 4. Experiment Result

*= Popularity bias for Incremental SVD - popularity bias for Incremental ApproSVD*

- **Experiment 2**: calculate the degree of popularity bias
  - Result
    - Fixed parameters:
      - n1=900, n2=100, c1=500, c2=50
      - k=10,q=10
    - **Single-side T-test** result
      - Null Hypothesis: the popularity bias using Incremental Appro-SVD is the same as Incremental SVD
      - Metric: average rating score
        - T score: 0.64
        - P value: 0.27
      - Metric: # of appearance
        - T score: 2.17
        - P value: 0.015
        - Incremental ApproSVD decreases the degree of popularity bias



# 4. Experiment Result

- **Experiment 2**: calculate the degree of popularity bias
  - Hyperparameter analysis
    - Result & conclusion:
      - The Incremental Appro-SVD algorithm has lower popularity bias
      - This effect is more significant when the approximation rank is high

| $n_1$ | $n_2$ | $c_1$ | $c_2$ | k | q | t | p |
|---|---|---|---|---|---|---|---|
| **900** | **100** | **500** | **50** | **10** | 10 | 2.17 | 0.015 |
| **800** | **200** | 500 | 50 | 10 | 10 | 2.42 | 0.008 |
| **700** | **300** | 500 | 50 | 10 | 10 | 2.12 | 0.17 |
| 900 | 100 | **400** | **40** | 10 | 10 | 1.618 | 0.05 |
| 900 | 100 | **600** | **60** | 10 | 10 | 1.63 | 0.05 |
| 900 | 100 | 500 | 50 | **15** | 10 | 2.739 | 0.003 |
| 900 | 100 | 500 | 50 | **5** | 10 | 1.53 | 0.06 |

# 4. Experiment Result

- **Experiment 3**: calculate the degree of concentration bias
  - Metric: Gini Coefficient
  - Result
    - The Incremental Appro-SVD algorithm has lower concentration bias than the Incremental SVD algorithm

| $n_1$ | $n_2$ | $c_1$ | $c_2$ | k | q | $gini_1$ | $gini_2$ |
|---|---|---|---|---|---|---|---|
| **900** | **100** | **500** | **50** | **10** | 10 | 0.096 | 0.106 |
| **800** | **200** | 500 | 50 | 10 | 10 | 0.096 | 0.106 |
| **700** | **300** | 500 | 50 | 10 | 10 | 0.096 | 0.106 |
| 900 | 100 | **400** | **40** | 10 | 10 | 0.096 | 0.102 |
| 900 | 100 | **600** | **60** | 10 | 10 | 0.096 | 0.104 |
| 900 | 100 | 500 | 50 | **15** | 10 | 0.099 | 0.108 |
| 900 | 100 | 500 | 50 | **5** | 10 | 0.087 | 0.092 |

# 4. Experiment Result

- **Experiment 4**: Improve by post-processing
  - Design
    - Add a **user-related** score to the prediction rating score
    - Various **L-2 regularization** factors: 0.05, 0.1, 0.2
    - Learning rate of 0.05 in **gradient descent**
  - Metrics: RMSE, average rating score, and Gini coefficients
  - Fixed parameters:
    - n1=900, n2=100, c1=500, c2=50, k=10,q=10
  - Result & conclusion:
    - As the regularization coefficient increases:
      - Accuracy does not change
      - Popularity bias decreases
      - Concentration bias decreases

*Row 2: control group*
*Column E: control group*

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | **lambda** | **RMSE** | **t** | **p** | **$gini_1$** | **$gini_2$** |
| 2 | **N/A** | 1.282 | 2.17 | 0.015 | 0.096 | 0.106 |
| 3 | **0.05** | 1.341 | 2.3 | 0.091 | 0.096 | 0.113 |
| 4 | **0.1** | 1.379 | 2.38 | 0.038 | 0.096 | 0.12 |
| 5 | **0.2** | 1.348 | 2.81 | 0.003 | 0.096 | 0.158 |

# 5. Conclusion

1. **(General Critics)** This algorithm is not scalable, and there lacks enough comparison with the sota methods, which reduces its applicability and contribution.
2. **(Paper reproduction)** The degradation of recommendation for Incremental Appro-SVD is larger than we expected. The improvement in time performance cannot prove its superiority.
3. **(Analysis)** Opposite to our estimation, the Incremental Appro-SVD algorithm decreases the popularity bias and the concentration bias.
    - **One possible explanation** is, the column selection mechanism only decides which part of the matrix is utilized, but will not directly reduce the recommendation score for those unpopular items
4. **(Exploration)** By post-processing, the popularity and concentration bias for Incremental Appro-SVD algorithm will be further reduced, while the accuracy does not change significantly.

# 6. Future Work

1. To explore whether time analysis will improve the performance of recommendation systems using the Incremental Appro-SVD Algorithm.
2. To explore whether the new algorithm increased the popularity reinforcement effect.
3. To test the performance of this algorithm on larger datasets.

# Thanks!

Colab Source Code link:
https://colab.research.google.com/drive/1oh9VuWEzzspovmLIERnFVBXmjiOBqdOg?usp=sharing
Github Source Code Link:
https://github.com/tristonerRL/CSDS313.git