Recommender systems

Roozbeh Ehsani

Spring 2023

Motivations

- Recommender systems provide personalized recommendations to users, which can lead to increased user satisfaction and engagement.

- They can help users discover new and relevant items or products that they may not have found otherwise.

- Recommender systems can improve the efficiency of online shopping or product search by reducing the amount of time users spend searching for items they want.

- They can also help businesses by increasing sales and revenue through targeted recommendations and personalized marketing.

Methods

1. User-based collaborative filtering: This algorithm recommends items based on similarities between users' past behavior, such as ratings or purchases.

2. Item-based collaborative filtering: This algorithm recommends items based on similarities between items themselves, such as genre, actors, or tags.

3. Matrix factorization: This algorithm involves decomposing a user-item matrix into two lower-dimensional matrices, which can be used to make predictions about users' preferences.

4. Many other methods…

Dataset

The 100k MovieLens dataset is a well-known benchmark dataset for recommender systems. It consists of 100,000 ratings on a scale of 1-5 provided by 943 users for 1,682 movies. The dataset also includes information about the movies such as their titles, release years, and genres.

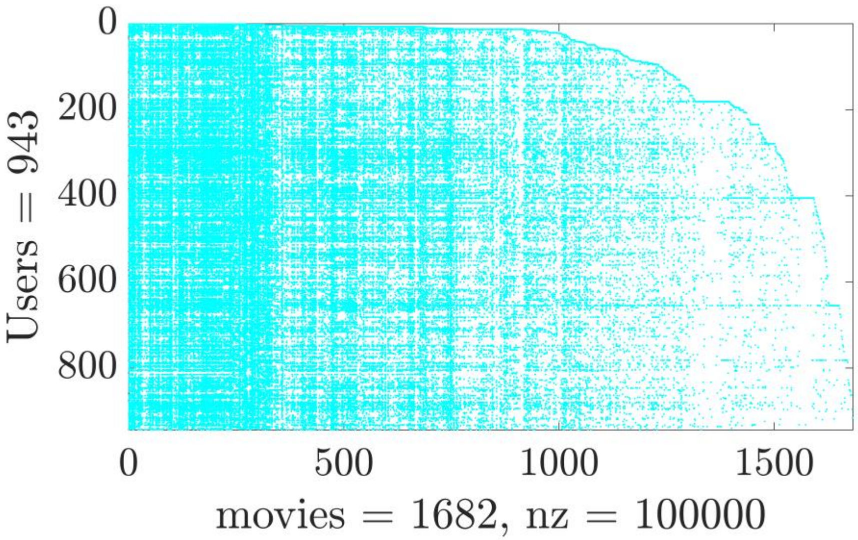
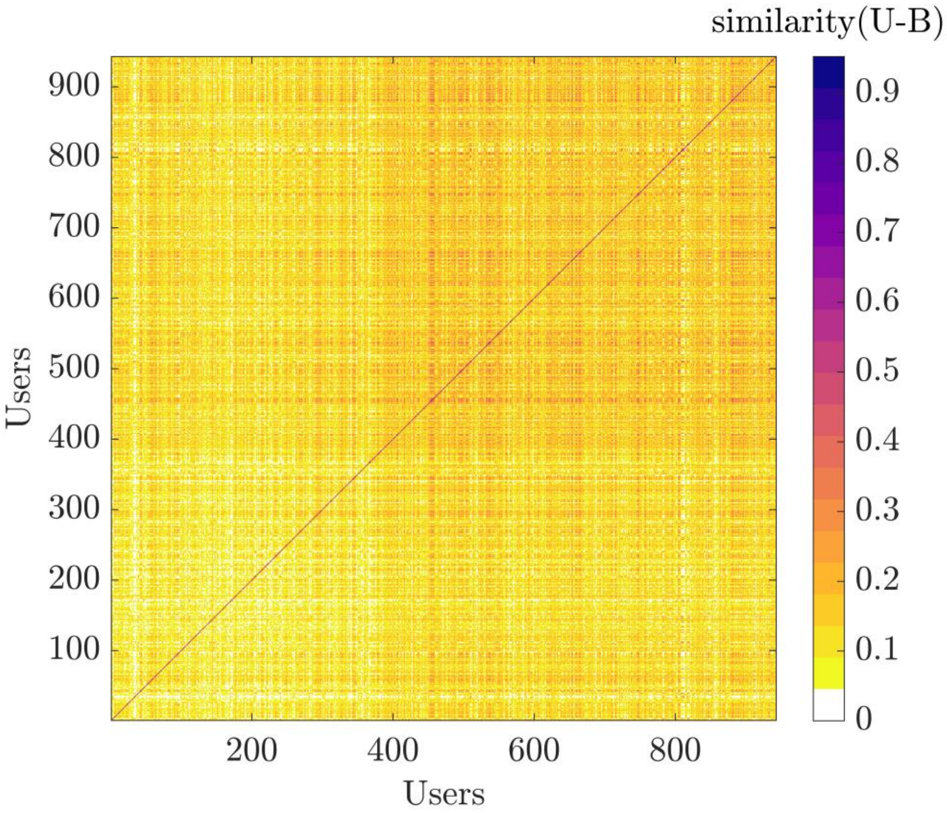


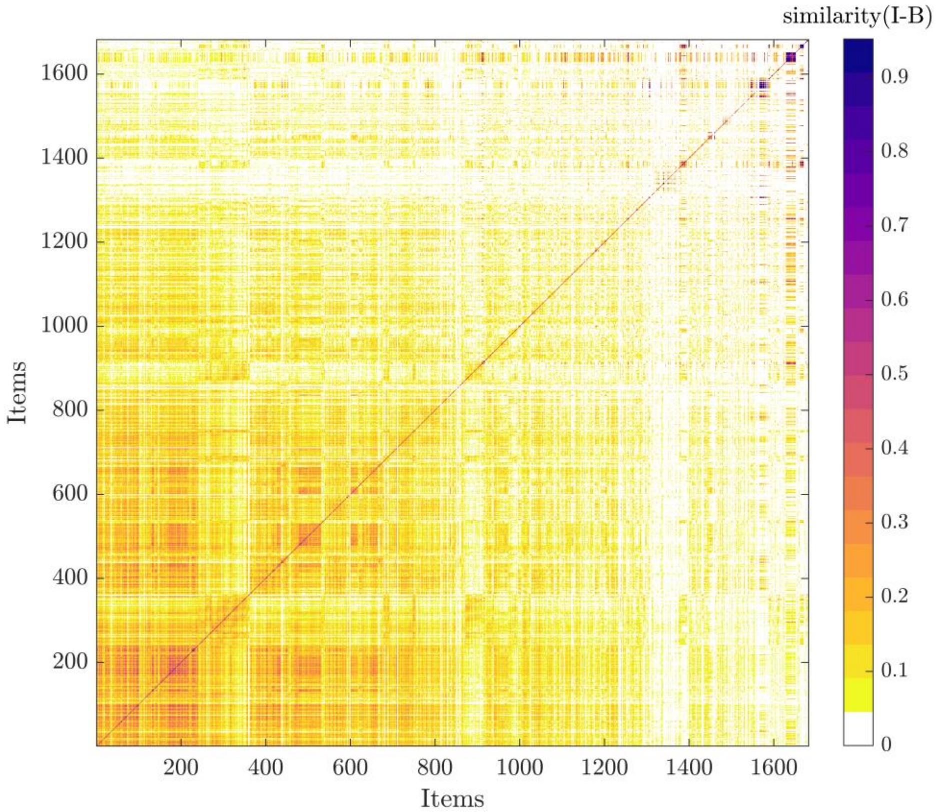
Figure 1: Spy illustration of the rating matrix R

User-based Collaborative filtering



$$\hat{r}_{ui} = \sum_{u'}^{K} Sim(UB)_{(u,u')} r_{u'i} \Big/ \sum_{u'}^{K} Sim(UB)_{(u,u')}$$

Item-based Collaborative filtering



$$\hat{r}_{ui} = \sum_{i'}^{K} Sim(IB)_{(i,i')} r_{ui'} \bigg/ \sum_{i'}^{K} Sim(IB)_{(i,i')}$$

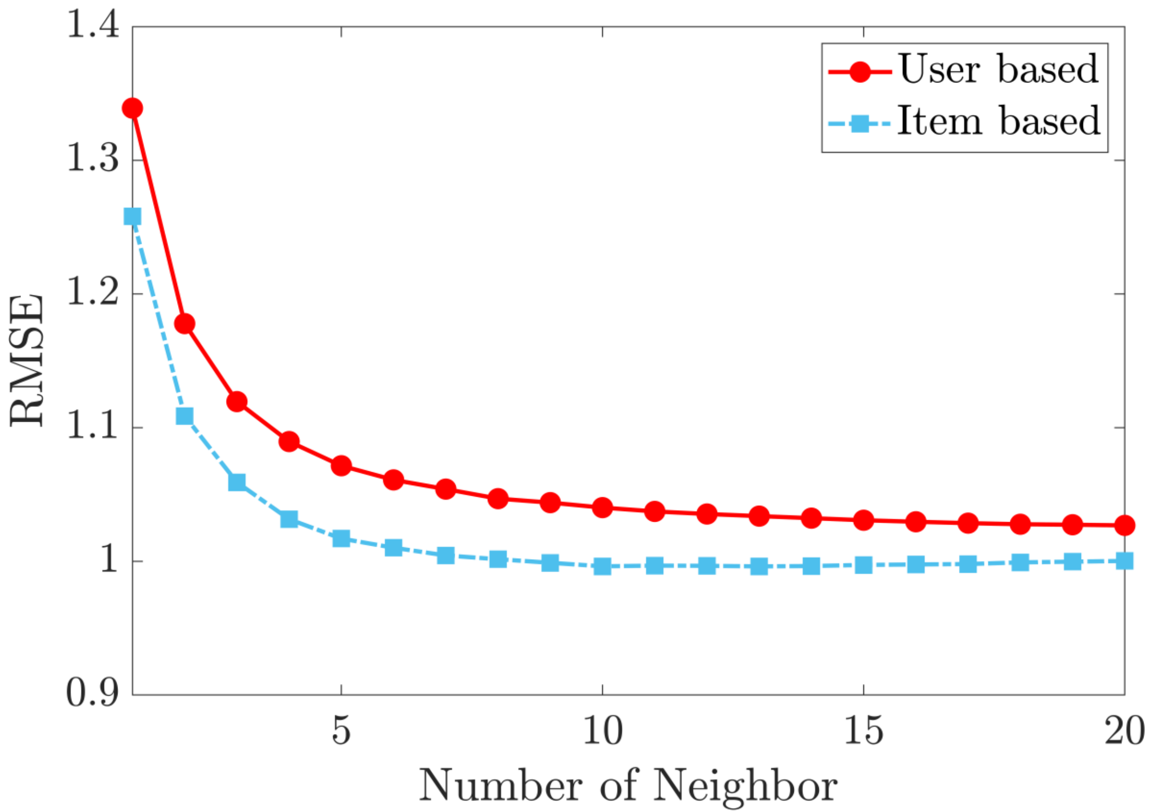


Figure 4: The value for RMSE for different number of neighbors. Red circles correspond to User-based collaborative filtering and blue squares belong to Item-based collaborative filtering.

Matrix factorization

$$X = \begin{bmatrix} & & \\ - & - & - \\ & x_u^T & \\ - & - & - \\ & & \\ & & \end{bmatrix}_{m \times f} \quad Y = \begin{bmatrix} & & \\ - & - & - \\ & y_i^T & \\ - & - & - \\ & & \\ & & \end{bmatrix}_{n \times f} \quad \rightarrow \quad \hat{R} = XY^T$$

$$J = \min_{X_*, Y_*} \|R - XY^T\|_F^2 \quad \rightarrow \quad J = \min_{X_*, Y_*} \|R - XY^T\|_F^2 + \lambda(\|X\|_F^2 + \|Y\|_F^2)$$

Alternative Least Square

$$\frac{\partial J}{\partial X} = 0 \quad \rightarrow \quad X^T = (Y^T Y + \lambda I)^{-1} Y^T R^T \quad \rightarrow \quad x_u = (Y^T Y + \lambda I)^{-1} Y^T r_u^T$$

$$\frac{\partial J}{\partial Y} = 0 \quad \rightarrow \quad Y^T = (X^T X + \lambda I)^{-1} X^T R \quad \rightarrow \quad y_i = (X^T X + \lambda I)^{-1} X^T r_i$$

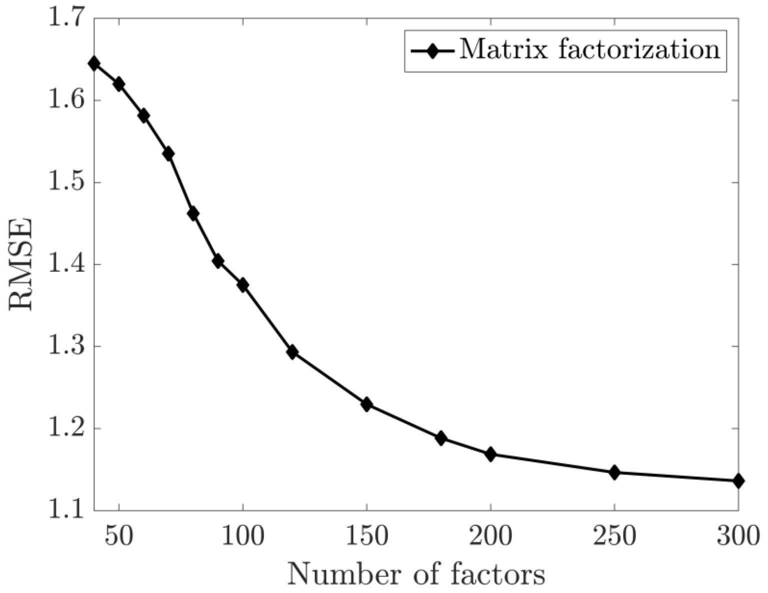


Figure 5: RMSE for the different number of factors for matrix factorization method.

Matrix factorization (Using Implicit Dataset)

r_{ui} can indicates the number of times user u bought item i or how many times user u fully watched movie i. $r_{ui} = 0.6$ means that user u watched 60 percent of movie i

$$R = \begin{bmatrix} & & \\ & r_{ui} & \\ & & \end{bmatrix}_{m\times n}$$

$$P = \begin{bmatrix} & & \\ & p_{ui} & \\ & & \end{bmatrix}_{m\times n} \qquad p_{ui} = \begin{cases} 1 & r_{ui} > 0 \\ 0 & r_{ui} = 0 \end{cases}$$

$$C = \begin{bmatrix} & & \\ & c_{ui} & \\ & & \end{bmatrix}_{m\times n} \qquad c_{ui} = 1 + \alpha r_{ui}$$

$$J = \min_{x_*, y_*} \sum_{u,i} c_{ui} (p_{ui} - x_u^T y_i)^2 + \lambda \left(\sum_u \|x_u\|^2 + \sum_i \|y_i\|^2 \right)$$

$$x_u = (Y^T C^u Y + \lambda I)^{-1} Y^T C^u p(u)$$

$$y_i = \left(X^T C^i X + \lambda I\right)^{-1} X^T C^i p(i)$$

For the evaluation of the model, the RMSE cannot be used anymore because we do not have explicit ranking by the users.

we can just use the predicted preference of users $\hat{p}_{ui}$ that its corresponding values in the test dataset be high p_{ui}^t.

$rank_{ui}$ is the percentile-ranking of program i within the ordered list of all programs prepared for user u. $rank_{ui} = 0\%$ would mean the program i is predicted to be the most desirable for user u. $rank_{ui} = 100\%$ indicates that program i is predicted to be the least preferred for user u.

$$\overline{rank} = \frac{\sum_{u,i} r_{ui}^t rank_{ui}}{\sum_{u,i} r_{ui}^t}$$

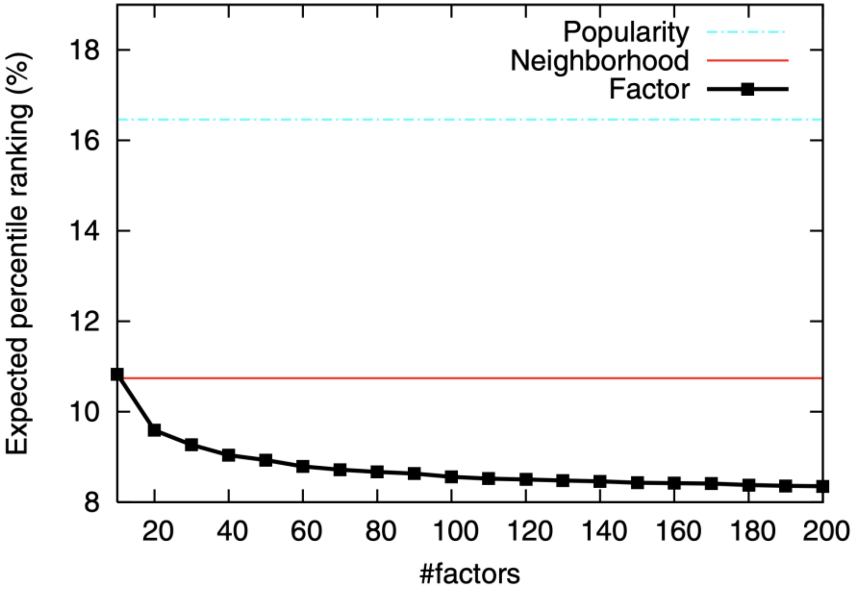


Figure 6: Comparing factor model with popularity ranking and neighborhood model [3].

Conclusion

- The item-based model has better performance than the user-based model.
- The performance of both matrix factorization models is getting better as the number of factor increases

Future works

Use modified projections methods and preconditioners in order to solve user-factor X and item-factor Y matrices.

References

[1] Ramlatchan, Andy, et al. "A survey of matrix completion methods for recommendation systems." *Big Data Mining and Analytics* 1.4 (2018): 308-323.

[2] Sarwar, Badrul, et al. *Application of dimensionality reduction in recommender system-a case study*. Minnesota Univ Minneapolis Dept of Computer Science, 2000.

[3] Hu, Yifan, Yehuda Koren, and Chris Volinsky. "Collaborative filtering for implicit feedback datasets." *2008 Eighth IEEE international conference on data mining*. Ieee, 2008.

[4] Saad, Yousef. Numerical methods for large eigenvalue problems: revised edition. Society for Industrial and Applied Mathematics, 2011.

University of Minnesota

Driven to Discover®

Crookston Duluth Morris Rochester Twin Cities

Backup Slides

$$J = \|R - XY^T\|_F^2 + \lambda(\|X\|_F^2 + \|Y\|_F^2) \rightarrow \|R\|_F^2 + \|XY^T\|_F^2 - 2tr(R^T XY^T) + \lambda(\|X\|_F^2 + \|Y\|_F^2)$$

$$J = tr(R^T R) + tr\left((XY^T)^T XY^T\right) - 2tr(R^T XY^T) + \lambda(tr(X^T X) + tr(Y^T Y))$$

$$\frac{\partial J}{\partial X} = dtr\left((XY^T)^T XY^T\right) - 2dtr(R^T XY^T) + \ \lambda(dtr(X^T X))$$

$$\frac{\partial J}{\partial X} = tr\left((dXY^T)^T XY^T\right) + tr\left((XY^T)^T dXY^T\right) - 2tr(R^T dXY^T) + \ \lambda(tr(dX^T X) + tr(X^T dX))$$

Backup Slides

$$\frac{\partial J}{\partial X} = tr(YdX^TXY^T) + tr(YX^TdXY^T) - 2tr(R^TdXY^T) + \lambda(tr(dX^TX) + tr(X^TdX)) \xrightarrow{tr\left(AA^T\right)=tr(A^TA)}$$

$$\frac{\partial J}{\partial X} = tr(YX^TdXY^T) + tr(YX^TdXY^T) - 2tr(R^TdXY^T) + \lambda(tr(X^TdX) + tr(X^TdX)) \xrightarrow{tr(ABC)=tr(CAB)}$$

$$\frac{\partial J}{\partial X} = 2tr(Y^TYX^TdX) - 2tr(Y^TR^TdX) + \lambda\left(2tr(X^TdX)\right) = 0$$

$$Y^TYX^T - Y^TR^T + \lambda X^T = 0$$

$$X^T = (Y^TY + \lambda I)^{-1}Y^TR^T$$