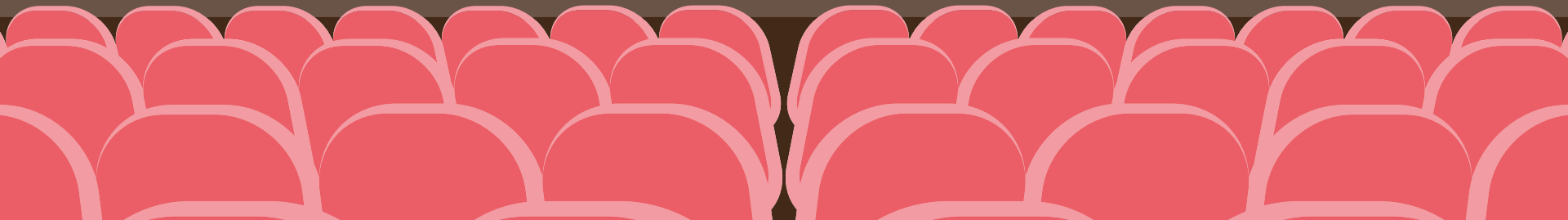
RECOMMENDER SYSTEM

Matrix Factorization - Alternating Least Squares



AGENDA

01. PROBLEM - DATASET DESCRIPTION

02. MF - ALS

03. RECOMMEND TO USERS

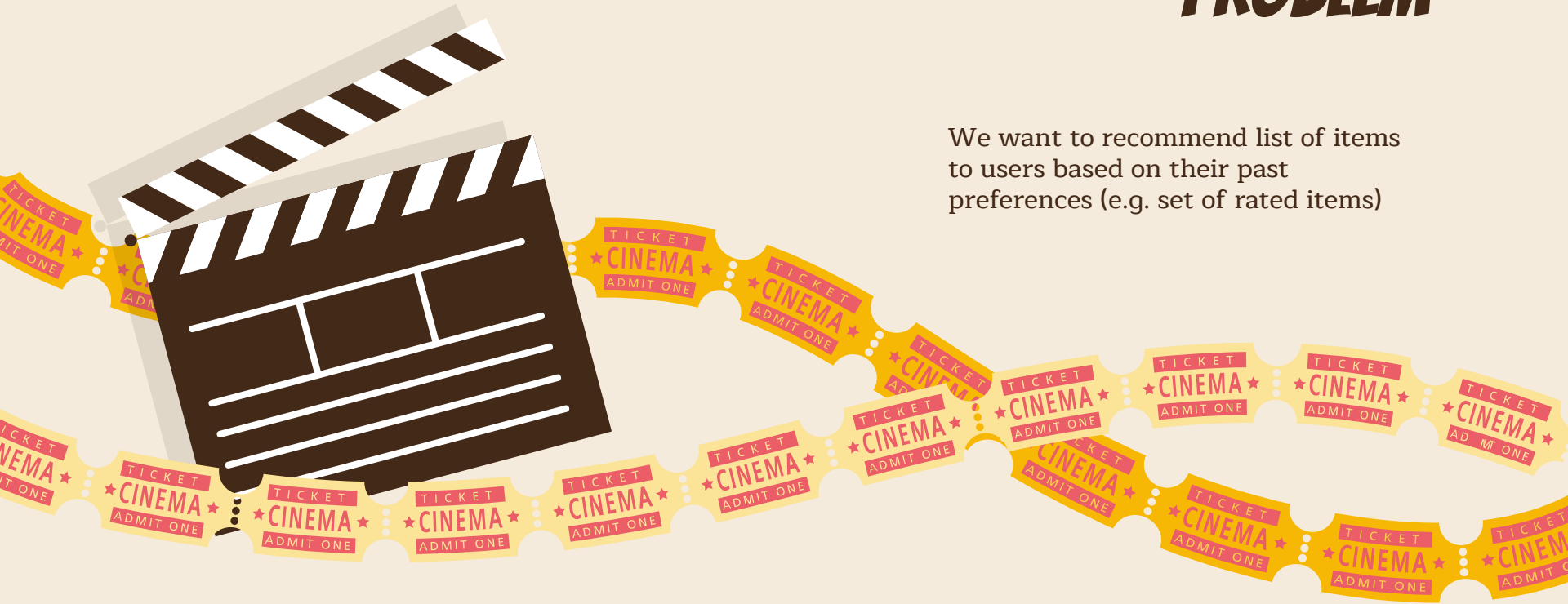
04. TUNE PARAMETERS

05. COLD START PROBLEM

06. PROS CONS

PROBLEM

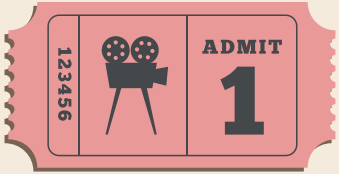
We want to recommend list of items to users based on their past preferences (e.g. set of rated items)



DATASET

MovieLens 100k

100836 ratings



Each user had rated at least 20 movies

9742 movies

Minimum rating 0.5
Maximum rating 5.0

610 users

The most rated movie had voted 329 times

DATA PREPARATION

Merge two file, ratings and movies.

MERGE

Create matrix from data, users as rows, movies as columns.

MATRIX

Fill the missing values with zero.

MISSING VALUES

MF - ALS

- Start with randomly generated U and V
- Fix U and find V

- Fix V and find U
- Repeat until number of iteration reached

ERROR FUNCTION AND COMPUTE FACTORS

Error Function that we want to minimize

$$L = \sum_{i,j \in Obs} w_{i,j}(c_{i,j} - u_{i,j}v_{i,j}^T)^2 + \lambda(\|U\|^2 + \|V\|^2)$$

Set Vector V as constant and take the derivative w.r.t U

$$\frac{\partial L}{\partial U} = -2\sum w(c - uv^T)v + 2\lambda U = 0$$

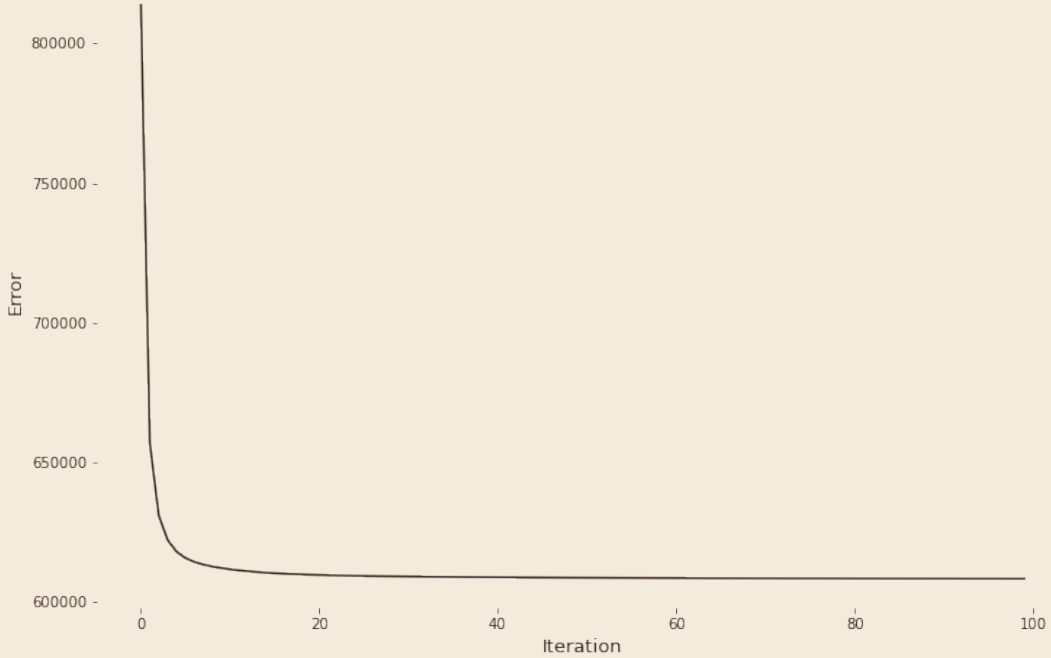
$$= U(wV^TV + \lambda I) = wCV$$

$$= U = wCV(wV^TV + \lambda I)^{-1}$$

Similarly, we can calculate the V by following same procedure

$$\frac{\partial L}{\partial V} = V = wCU(wU^TU + \lambda I)^{-1}$$

CHANGE IN ERROR WITH ITERATIONS



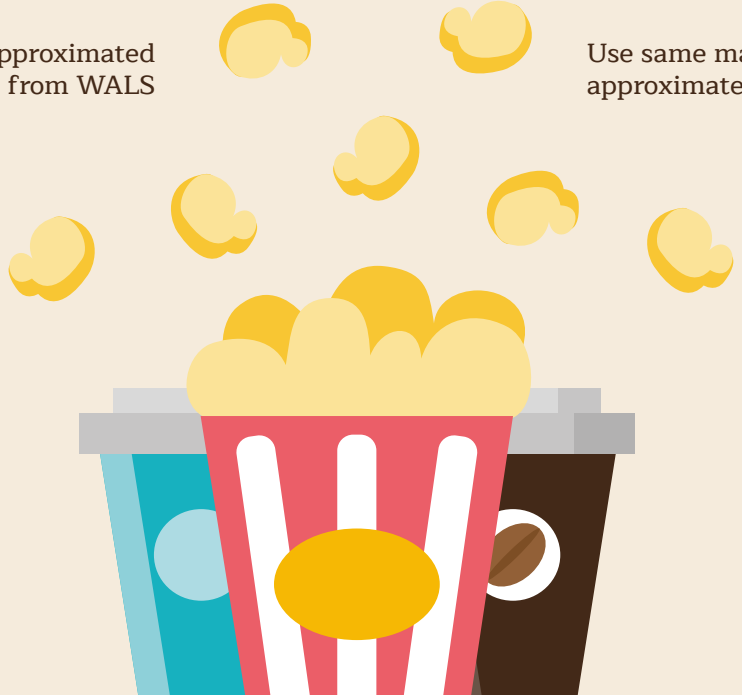
RECOMMEND TO USERS

Get the approximated matrix from WALS

Use same mask for approximated matrix

Mask the movies which is not rated, from feedback matrix

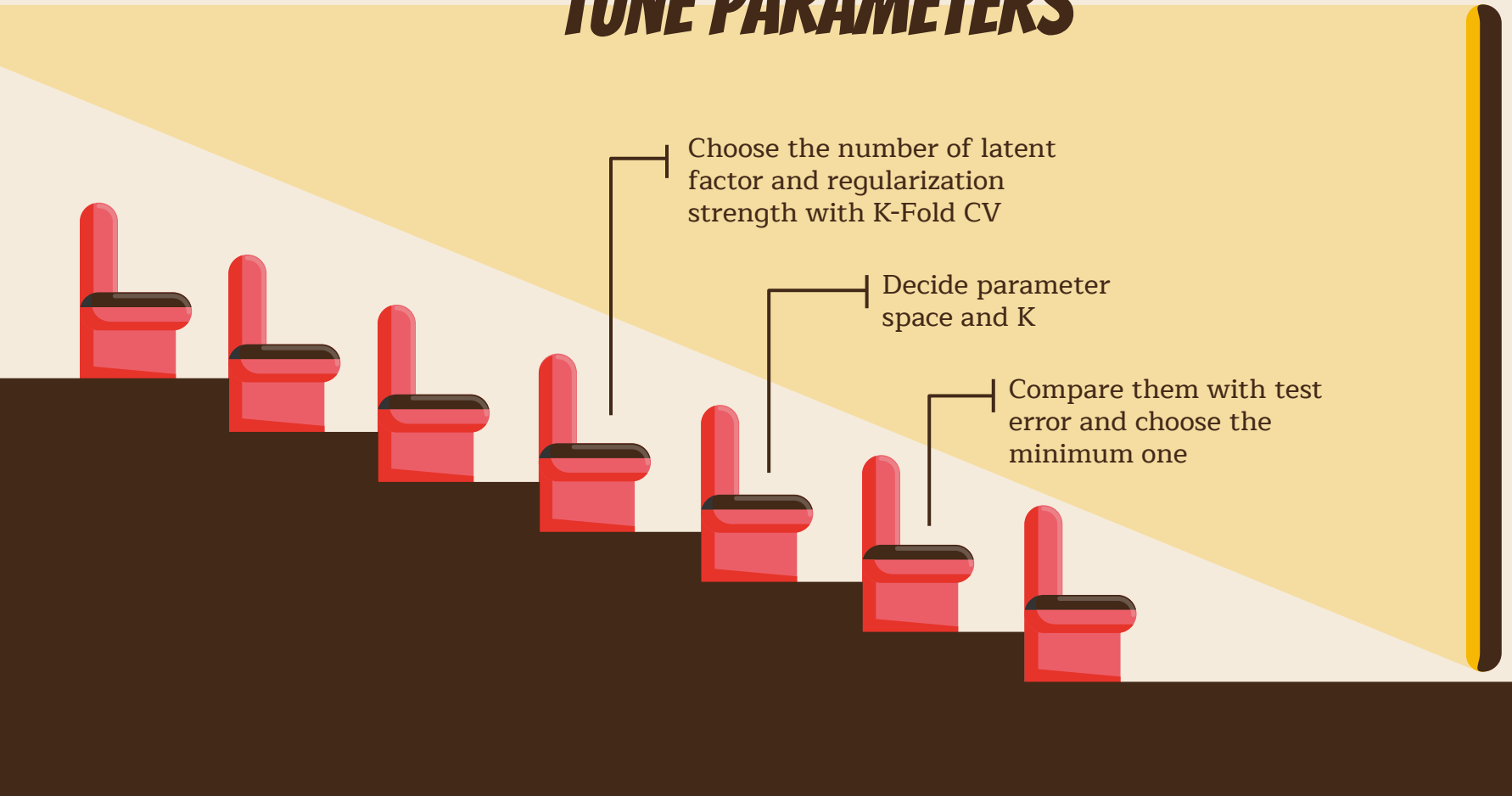
Sort the results descending order, return recommendations



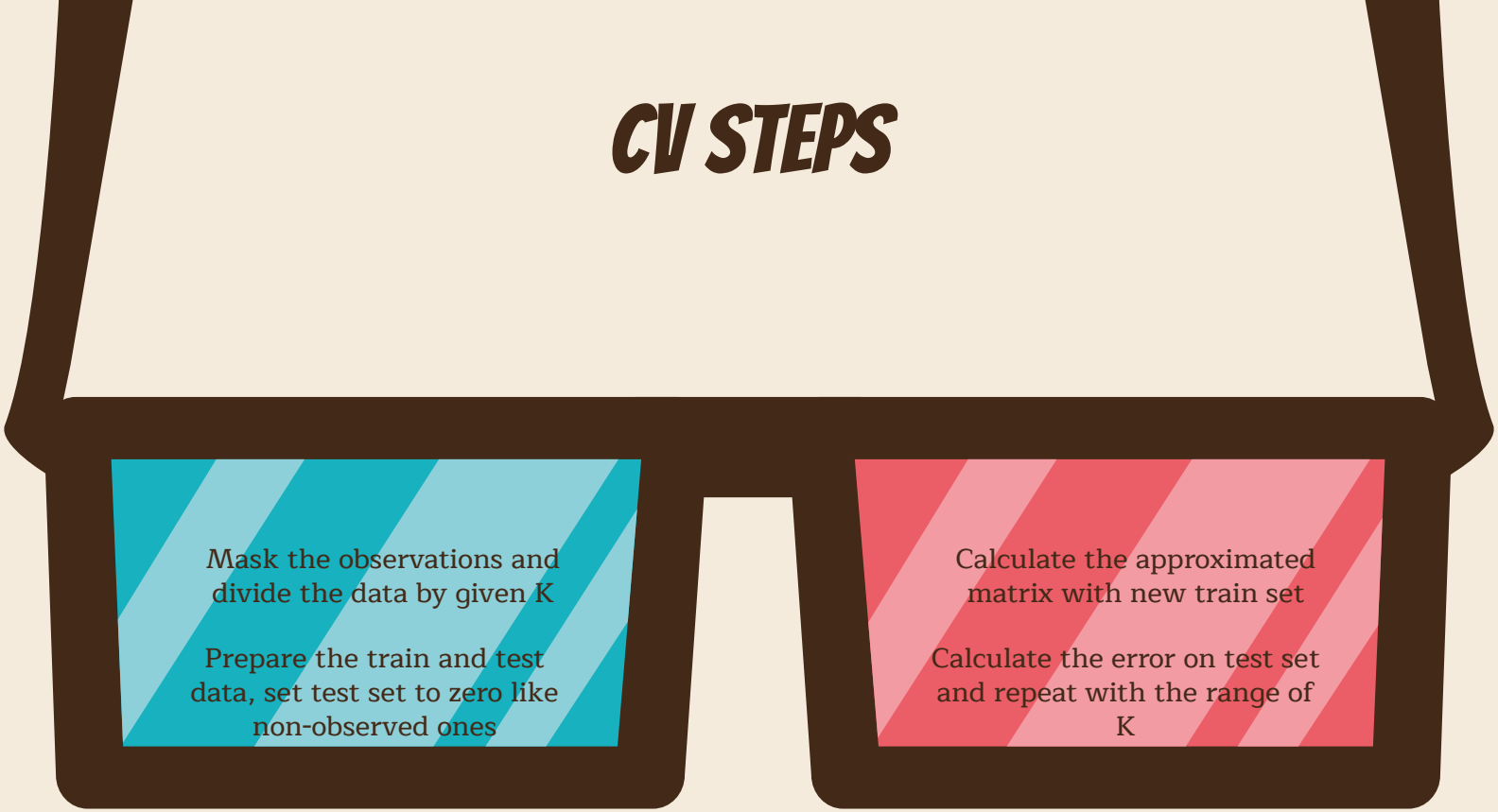
RECOMMEND TO USER (455) EXAMPLE

MOVIE ID	SCORE	TITLE	GENRES
296	3.56	Pulp Fiction (1994)	Comedy|Crime|Drama|Thriller
153	2.31	Batman Forever (1995)	Action|Adventure|Comedy|Crime
185	2.16	Net, The (1995)	Action|Crime|Thriller
47	2.03	Seven (a.k.a. Se7en) (1995)	Mystery|Thriller
208	1.87	Waterworld (1995)	Action|Adventure|Sci-Fi

TUNE PARAMETERS



CV STEPS



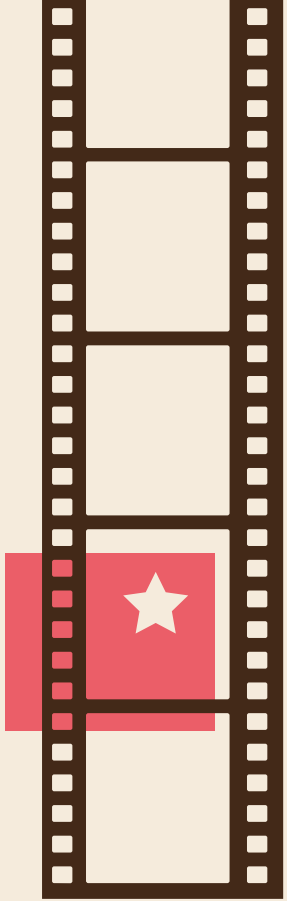
Mask the observations and divide the data by given K

Prepare the train and test data, set test set to zero like non-observed ones

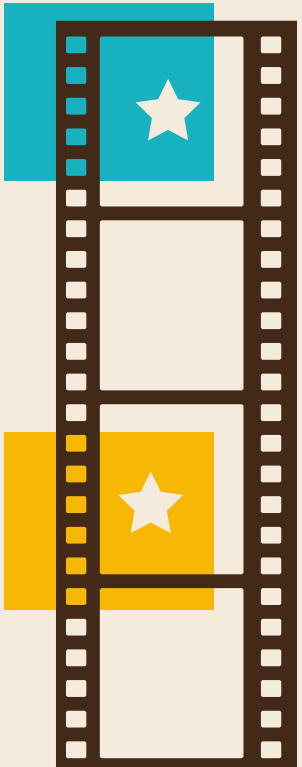
Calculate the approximated matrix with new train set

Calculate the error on test set and repeat with the range of K

COLD START PROBLEM FOR USERS



What should we recommend when a new user enter the system



User has no ratings, so we can recommend most popular movies

Most popular movies are the ones which have the highest average of ratings

RECOMMEND TO NEW USER EXAMPLE

MOVIE ID	SCORE	TITLE	GENRES
318	2.30	Shawshank Redemption, The (1994)	Crime|Drama
356	2.24	Forrest Gump (1994)	Comedy|Drama|Romance|War
296	2.11	Pulp Fiction (1994)	Comedy|Crime|Drama|Thriller
2571	1.91	Matrix, The (1999)	Action|Sci-Fi|Thriller
593	1.90	Silence of the Lambs, The (1991)	Crime|Horror|Thriller

PROS CONS

Converges fast with few iteration

Train and recommendations are efficient

Provide a solution for cold start problem for users

Reliant on Loss Squares only (ALS)

Sparsity may spawn problems

Parameter tuning takes time

REFERENCES

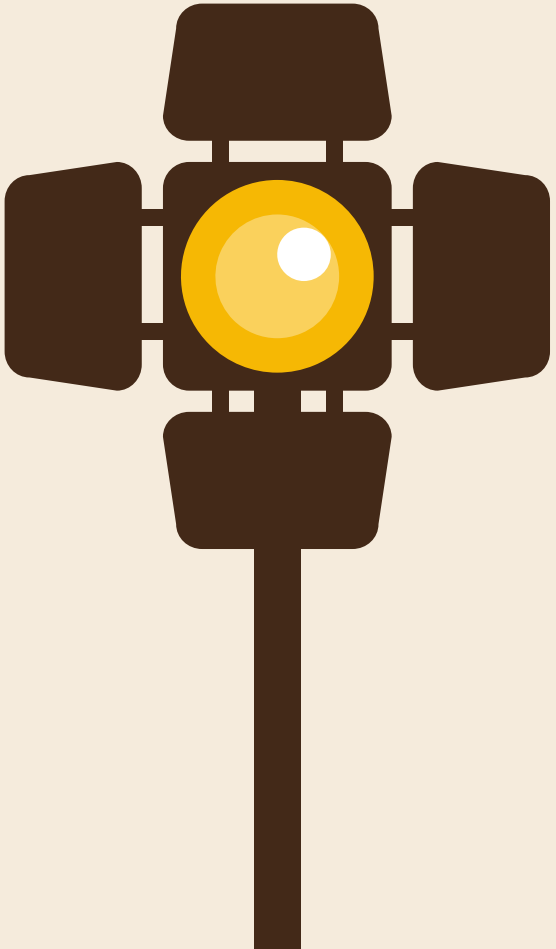
MovieLens

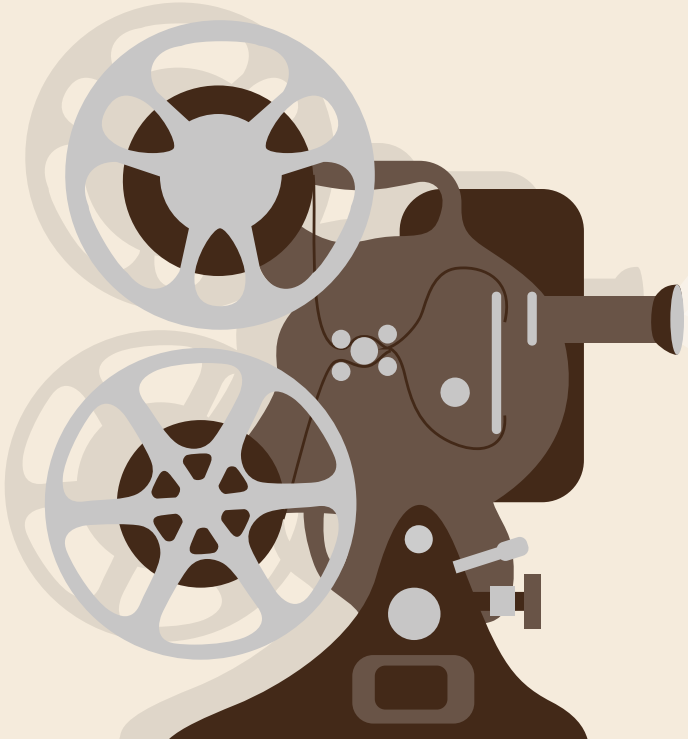
MATRIX FACTORIZATION TECHNIQUES FOR RECOMMENDER SYSTEMS

Matrix Factorization | Recommendation Systems

1_ALSWR

Explicit Matrix Factorization: ALS, SGD, and All That Jazz





THANKS

Do you have any questions?