Probabilistic Matrix Factorization for Music Recommendation

Lu Liu
luliu7@stanford.edu

Stanford
cs229 Final Project

Introduction

- Applied Probabilistic Matrix Factorization(PMF)[1] and variations of PMF to the task of music recommendation;
- Developed Constrained Kernelized Probabilistic Matrix Factorization (cKPMF) which we show to be superior to all other models discussed in this project for our task at hand.
- Motivation: learn and experiment with a simple but effective framework for making recommender systems, that can deal with sparse and imbalanced rating data.
- Input: log(listening count), user network, artist tag assignment.
- Output: missing ratings (listening count)

Data

hetrec2011-lastfm-2k [2], a set of social networking, tagging, and music artist listening information from last.fm.

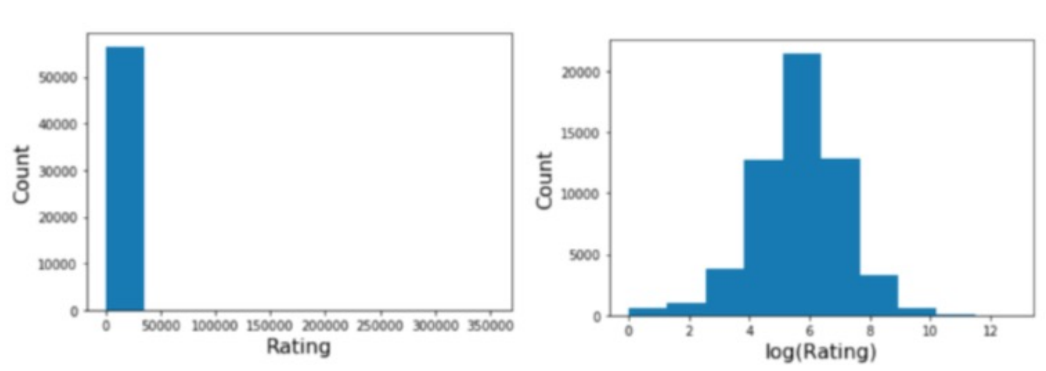
User- artist listening count

$$\begin{array}{c} \\ user_1 \\ user_2 \\ user_3 \end{array} \begin{array}{c} \begin{array}{ccc} Coldplay & Moby & Gorillaz \end{array} \\ \begin{pmatrix} 13883 & 8983 & 100 \\ 543 & 8908 & ? \\ ? & ? & ? \end{pmatrix} \end{array}$$

Artist Side information

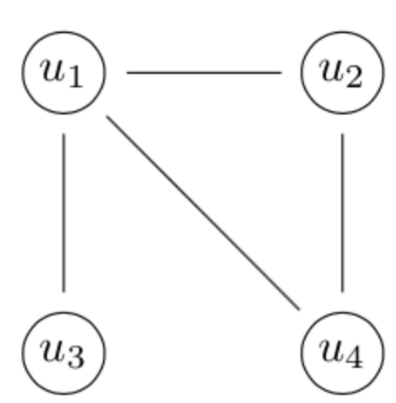
$$\begin{array}{c} \\ Coldplay \\ Moby \\ Gorillaz \end{array} \begin{array}{c} \begin{array}{ccc} ballad & pissbass & chillout \end{array} \\ \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix} \end{array}$$

log(listening count) as Pseudo-Ratings



(a) Rating distribution (b) Log-rating distribution

User Side information

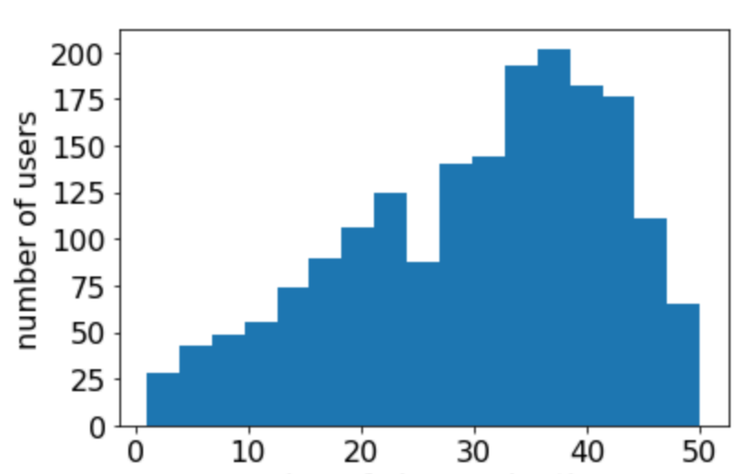


Challenges

sparsity

Imbalance

ITEM	STATS
# USERS	1871
# ITEMS	1000
# RATINGS	56620
RATING DENSITY	3.03%
# RELATIONS	25424
# TAGS	87366



Models

Notations

N - Number of users M - Number of items
R - Rating matrix D - latent dimension
U - user latent matrix V - items latent matrix
K_u - Covariance matrix for rows of R
K_v - Covariance matrix for columns of R
I - indicator matrix for whether $R_{i,j}$ is observed
W - latent similarity constraint matrix

Ratings

User side

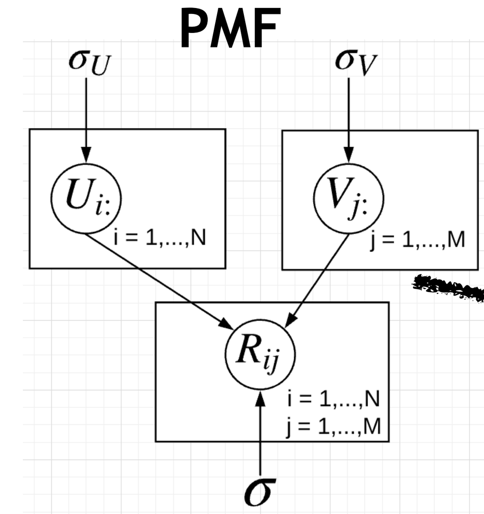
Commute Time kernel
$= \text{Laplacian Matrix}^{\dagger}$

Artist side

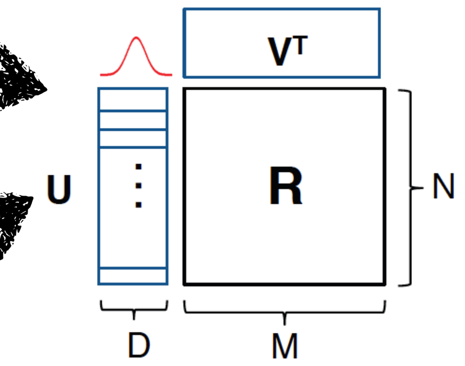
RBF kernel

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{2\sigma^2}\right)$$

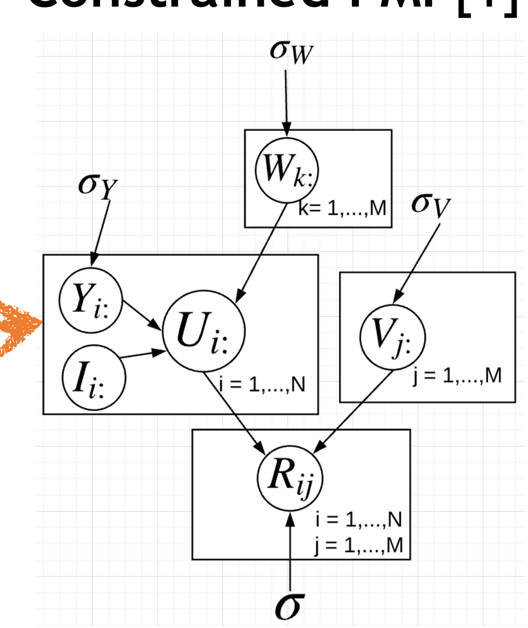
PMF



U,V both Sampled Row-wise
Assume generated from zero mean Normal Distrbution



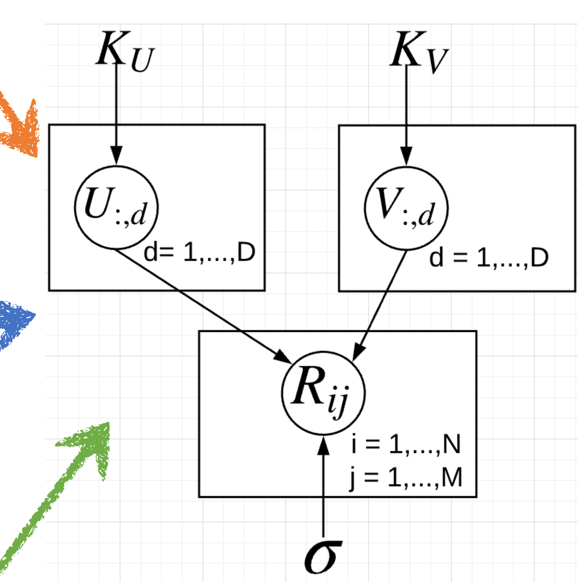
Constrained PMF[1]



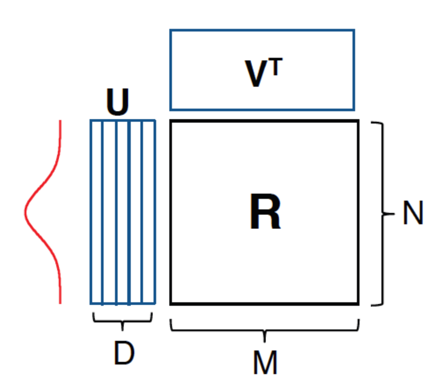
Constrain U with W :Users who rated the same artists should have similar latent feature vector

$$U_{i,:} = Y_{i,:} + \frac{\sum_{k=1}^{M} I_{i,k} W_{k,:}}{\sum_{k=1}^{M} I_{i,k}}$$

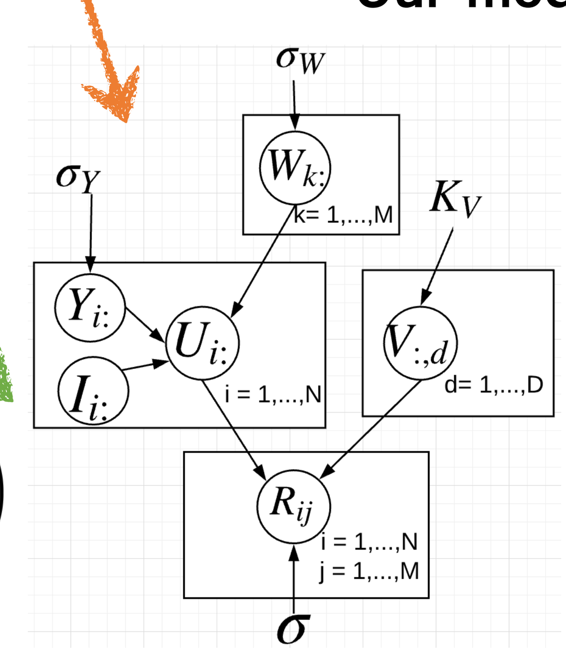
Kernelized PMF[3]



U, V both Sampled Column-wise.
Assume generated from zero-mean GP
Capture covariance between rows



Our model: cKPMF



U is sampled row-wise
V is sampled column-wise

MAP: Maximize log-posterior over Y,W,V

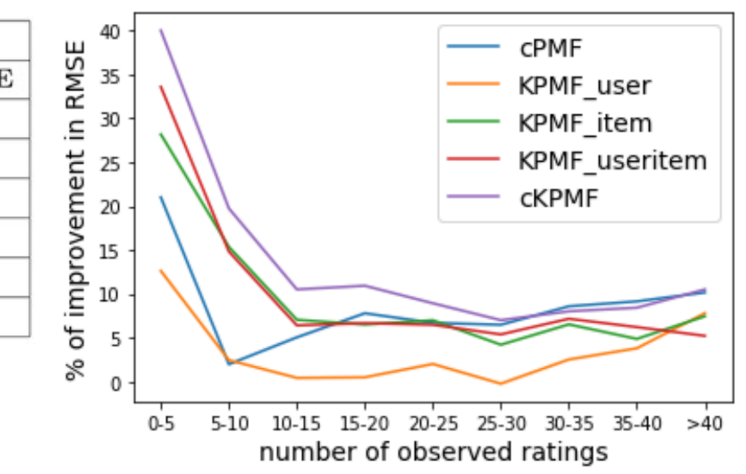
$$\begin{aligned} & p(Y, W, V | R, \sigma^2, \sigma_Y^2, \sigma_W^2, K_V) \\ =& p(R|Y, W, V, \sigma^2) p(Y|\sigma_Y^2) p(W|\sigma_W^2) p(V|K_V) \\ =& \prod_{i=1}^{N} \prod_{j=1}^{M} [\mathcal{N}(R_{i,j} | (Y_{i,:} + \frac{\sum_{k=1}^{M} I_{i,k} W_{k,:}}{\sum_{k=1}^{M} I_{i,k}}) V_{j,:}^T), \sigma^2)]^{I_{i,j}} \\ & \prod_{i=1}^{N} \mathcal{N}(Y_{i,:}|0, \sigma_Y^2 I) \prod_{j=1}^{M} \mathcal{N}(W_{j,:}|0, \sigma_W^2 I) \prod_{d=1}^{D} GP(V_{:,d}|0, K_V) \end{aligned}$$

Experiments/Results

- 10% as validation ; 10% as test; (5662 ratings)
- Training set 1: 80% of ratings (45296 ratings)
- Training set 2: 20% of ratings (11324 ratings)
- Training using Stochastic Gradient Descent
- Grid search for hyperparameters (latent dimension, learning rate, regulation coefficient) using validation set.
- Evaluation: $RMSE = \sqrt{\frac{\sum_{i=1}^{n}(r_i - \hat{r}_i)^2}{n}}$

Model	80% ratings		20% ratings	
	train RMSE	test RMSE	train RMSE	test RMSE
PMF	0.827	1.139	0.600	2.808
cPMF	0.895	1.056	0.800	1.662
KPMF_user	0.815	1.119	0.598	2.444
KPMF_item	0.794	1.070	0.598	1.361
KPMF_useritem	0.787	1.064	0.598	1.273
cKPMF	0.778	**1.039**	0.621	**1.224**

% improvement in test RMSE over PMF by user group



- Having more training data is more important than picking the best model. The best model performance with 20% ratings (1.224) is still worse than the worst model performance with 80% ratings (1.139).
- Kernelized models able to use side information to make predictions when ratings are sparse. Though both useful, user side information is not as effective as item side information.
- Difference in model performance is much more obvious when data is sparse. (with 20% ratings)
- All PMF variations made more improvement over the baseline PMF model for users with 0-5 ratings.
- Our cKPMF model exceed the performance of all others. Advantage is more obvious for infrequent users.

Future work

- Explore the effect of adding user and artist bias into all these models.
- Investigate the computational efficiency and convergence behaviour of these models in different settings.
- Explore Bayesian Probabilistic Matrix Factorization

References:
[1] A. Mnih and R. R. Salakhutdinov. Probabilistic matrix factorization. In Advances in neural information processing systems, pages 1257–1264, 2008.
[2] I. Cantador, P. Brusilovsky, and T. Kuflik. 2nd workshop on information heterogeneity and fusion in recommender systems (hetrec 2011). In Proceedings of the 5th ACM conference on Recommender systems, RecSys 2011,New York, NY, USA, 2011. ACM.
[3] T. Zhou, H. Shan, A. Banerjee, and G. Sapiro. Kernelized probabilistic matrix factorization: Exploiting graphs and side information. In Proceedings of the 2012 SIAM international Conference on Data mining, pages 403–414. SIAM, 2012.

https://drive.google.com/open?id=1rt7CFxppZGFxF2pttmysw3FMSOm0dazJ