

Multiple Multiplicative Factor Model For Collaborative Filtering

Benjamin Marlin **Richard S. Zemel**
marlin@cs.toronto.edu **zemel@cs.toronto.edu**

University of Toronto. Department of Computer Science.
Toronto, Ontario, Canada

1. Abstract

- We describe a class of causal, discrete latent variable models called Multiple Multiplicative Factor models (MMFs).
- MMFs pair a directed (causal) model with multiplicative combination rules. The product formulation of MMFs allow factors to specialize to a subset of the items, while the causal generative semantics mean MMFs can readily accommodate missing data.
- We present a Binary/Multinomial MMF model along with variational inference and learning procedures. We apply the model to the task of rating prediction for collaborative filtering.
- We present empirical results showing that a binary/multinomial MMF model matches the performance of the best existing models while learning an interesting latent space description of the users.

2. Introduction

Collaborative Filtering Formulations

Preference Indicators

Co-occurrence Pair (u,y) : u is a user index and y is an item index.

Count Vector $(n_1^u, n_2^u, \dots, n_M^u)$: n_y^u is the number of times (u,y) is observed.

Rating Triplet (u,y,r) : u is a user index, y is an item index, r is a rating value.

Rating Vector $(r_1^u, r_2^u, \dots, r_M^u)$: r_y^u is rating assigned to item y by user u .

Content-Based Features

In a **pure** formulation no additional features are used. A **hybrid** formulation incorporates additional content-based item and user features.

Sequence-Based Features

In a **sequential** formulation the rating process is modeled as a time series. In a **non-sequential** formulation preferences are assumed to be static.

Pure Rating-Based Collaborative Filtering

Formal Description:

Items: $y=1,\dots,M$

Users: $u=1,\dots,N$

Ratings: $r=1,\dots,V$

Preference Indicators: Ordinal rating vectors

Content-Based Features: None

Sequence-Based Features: None

Rating Matrix	Items						
	1	2	3	4	M	
Users	1	1	2	□	1	□
	2	□	1	5	□	5
	3	□	2	1	4	□
	4	4	□	5	4	3
	⋮	⋮	⋮	⋮	⋮	⋮	⋮
N	□	□	1	□	2	

- **Items:** $y=1,\dots,M$
- **Users:** $u=1,\dots,N$
- **Ratings:** $r^u_y \in \{1,\dots,V\}$
- **Profiles:** $\mathbf{r}^u \in \{1,\dots,V, \emptyset\}^M$

Previous Studies:

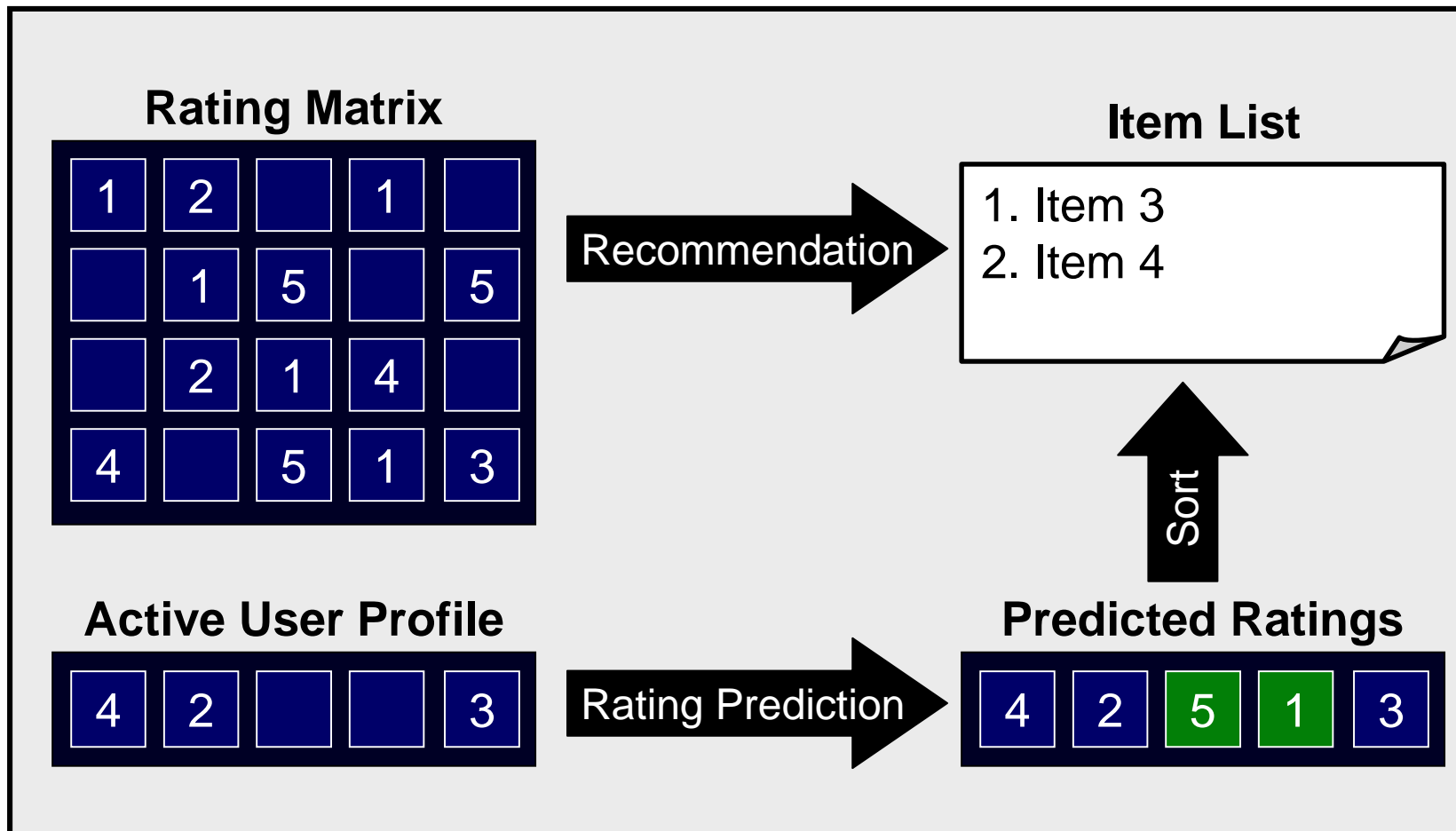
This is the formulation used by Resnick et al. (GroupLens), Breese et al (Empirical Evaluations), Hofmann (Aspect Model), Marlin (URP).

Figure 1: Pure rating-based collaborative filtering data in matrix form.

Pure Rating-Based Collaborative Filtering

Tasks:

- **Recommendation:** Selecting items the active user might like or find useful.
- **Rating Prediction:** Predicting all missing ratings in a user profile.



Recommendation by Rating Prediction:

Recommendation can be performed by predicting all missing ratings in the active user profile, and then sorting the unrated items by their predicted ratings.

The focus of research in this area is developing highly accurate rating prediction methods.

Figure 2: A break down of the recommendation problem into sub-tasks.

3. Related Work

Neighborhood Methods:

- Introduced by Resnick et al (GroupLens), Shardanand and Maes (Ringo).
- All variants can be seen as modifications of the K -Nearest Neighbor classifier.

Rating Prediction:

1. Compute similarity measure between active user and all users in database.

$$w_{au}^P = \frac{\sum_{\{i|r_i^a, r_i^u \neq \perp\}} (r_i^a - \bar{r}^a)(r_i^u - \bar{r}^u)}{\sqrt{\sum_{\{i|r_i^a, r_i^u \neq \perp\}} (r_i^a - \bar{r}^a)^2 \sum_{\{i|r_i^a, r_i^u \neq \perp\}} (r_i^u - \bar{r}^u)^2}}$$

2. Compute predicted rating for each item.

$$\hat{r}_y^a = \bar{r}^a + \frac{\sum_{k=1}^K w_{au_k}^P (r_y^{u_k} - \bar{r}^{u_k})}{\sum_{k=1}^K |w_{au_k}^P|}$$

Multinomial Mixture Model:

- A simple mixture model with fast, reliable learning by EM, and low prediction time.
- Simple but correct generative semantics. Each profile is generated by 1 of K types.

Learning:

$$\text{E-Step: } \phi_z^{u,t+1} \leftarrow \frac{\theta_z^t \prod_{y=1}^M \prod_{v=1}^V \beta_{vyz}^t \delta(r_y^a, v)}{\sum_{z'=1}^K \theta_{z'}^t \prod_{y=1}^M \prod_{v=1}^V \beta_{vyz'}^t \delta(r_y^a, v)}$$

$$\text{M-Step: } \theta_z^{t+1} \leftarrow \frac{\sum_{u=1}^N \phi_z^{u,t+1}}{\sum_{z=1}^K \sum_{u=1}^N \phi_z^{u,t+1}}$$

$$\beta_{vyz}^{t+1} \leftarrow \frac{\sum_{u=1}^N \phi_z^{u,t+1} \delta(r_y^a, v)}{\sum_{v'=1}^V \sum_{u=1}^N \phi_z^{u,t+1} \delta(r_y^a, v')}$$

Rating Prediction:

$$P(R_y = v | \mathbf{r}^a, \theta, \beta) = \sum_{z=1}^K P(R_y = v | Z = z) P(Z = z | \mathbf{r}^a)$$

User Rating Profile Model:

- Proposed by Marlin as a correct generative version of the aspect model for collaborative filtering.
- Has a rich latent space description of a user as a distribution over attitudes, but this distribution is not reflected in the generation of individual ratings.
- Has achieved the best results on EachMovie and MovieLens data sets.

Learning:

- Model learned using variational EM.

Prediction:

- Needs approximate inference. Variational methods result in an iterative algorithm.

Graphical Models:

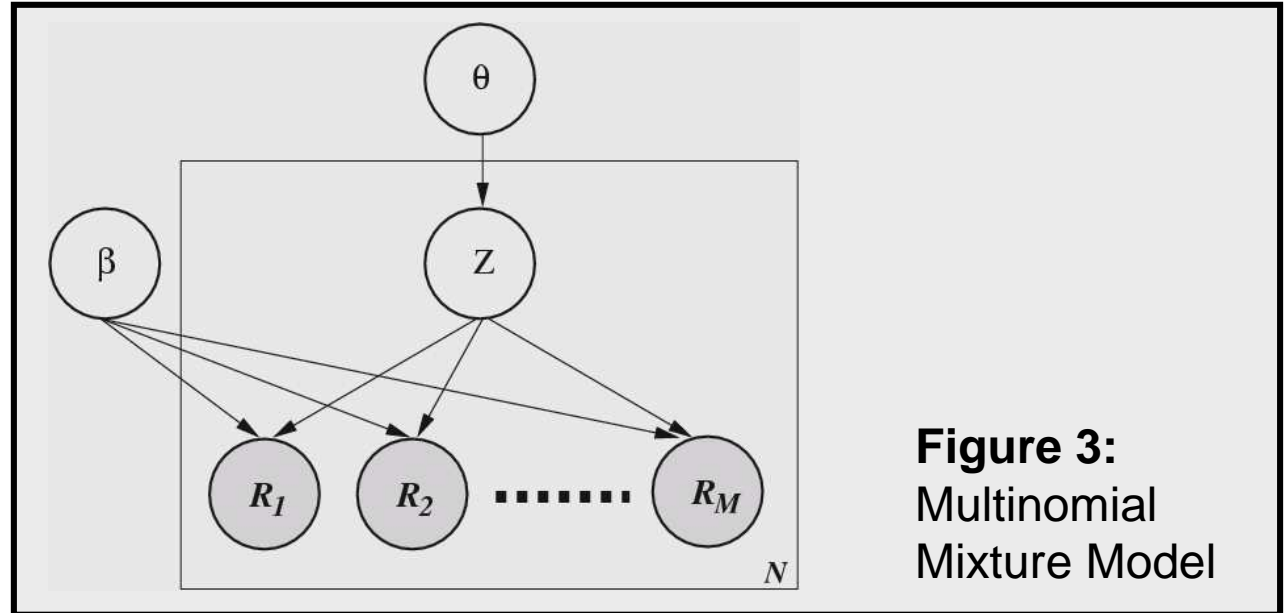


Figure 3:
Multinomial
Mixture Model

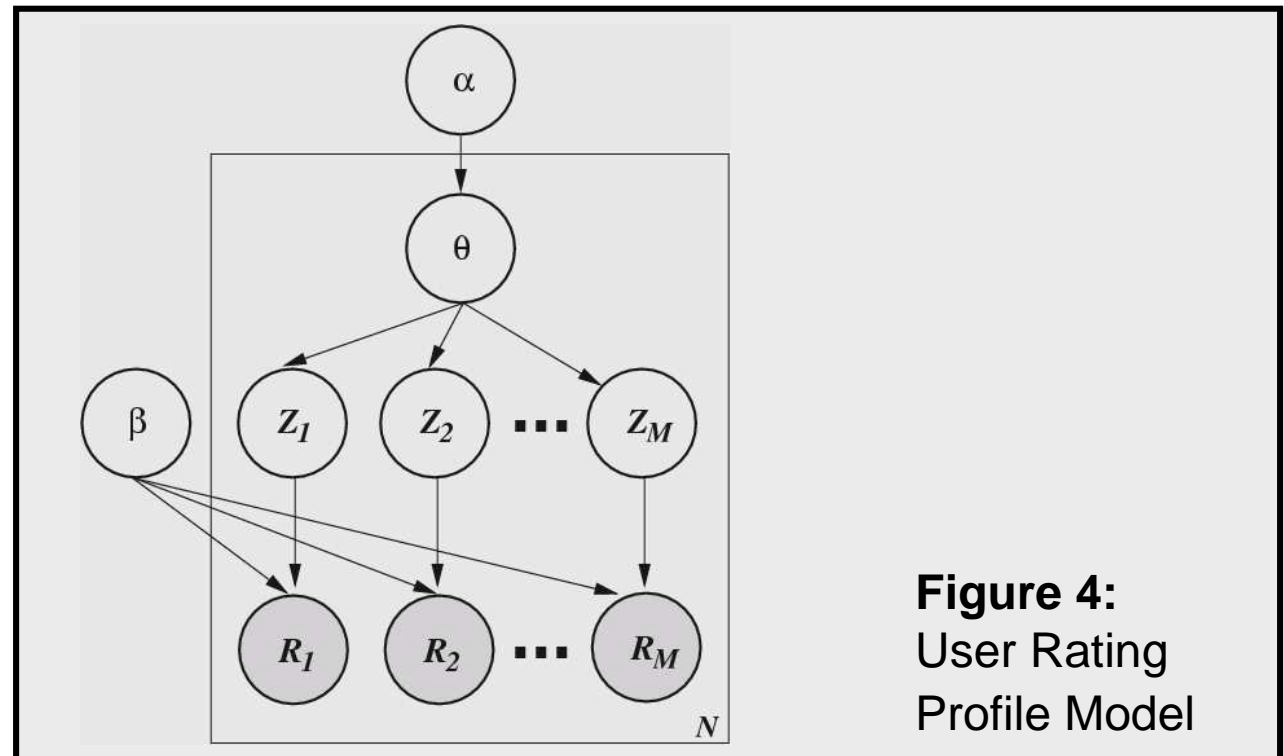


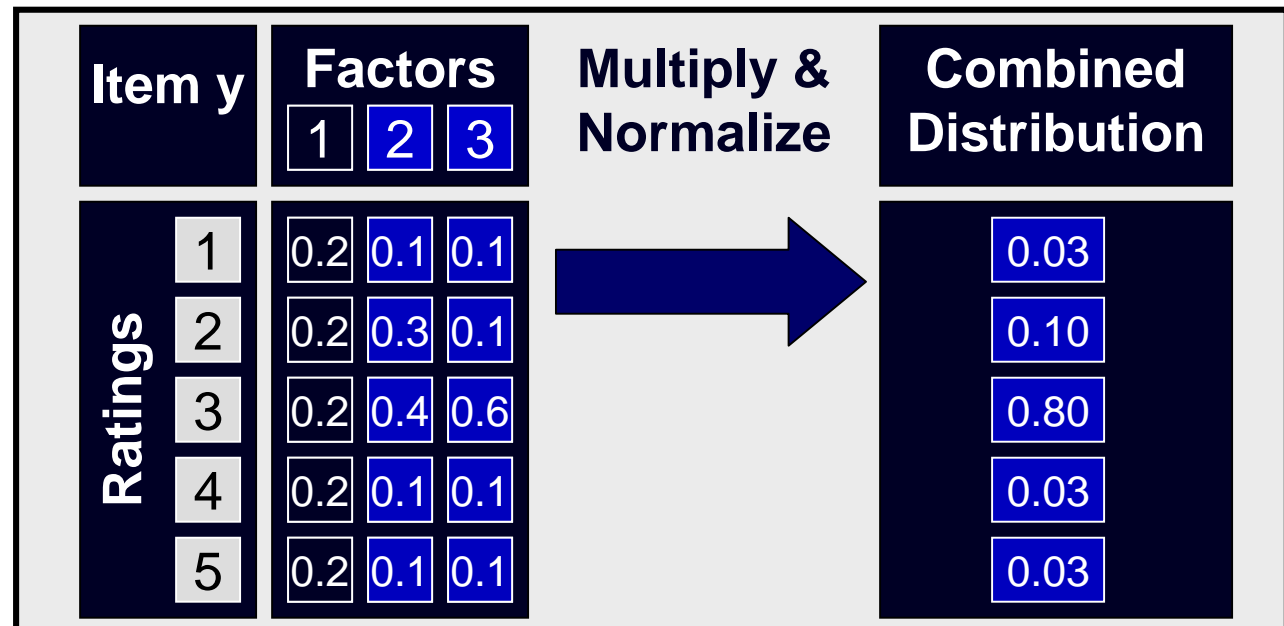
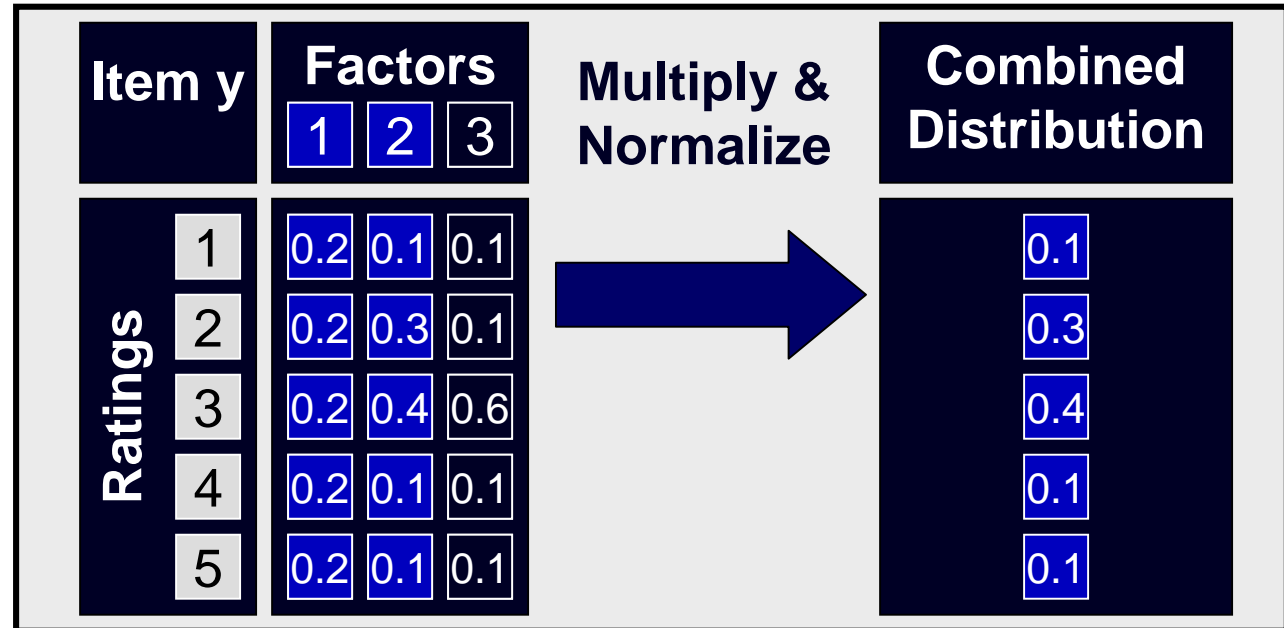
Figure 4:
User Rating
Profile Model

4. Multiple Multiplicative Factor Model

Features and Motivation:

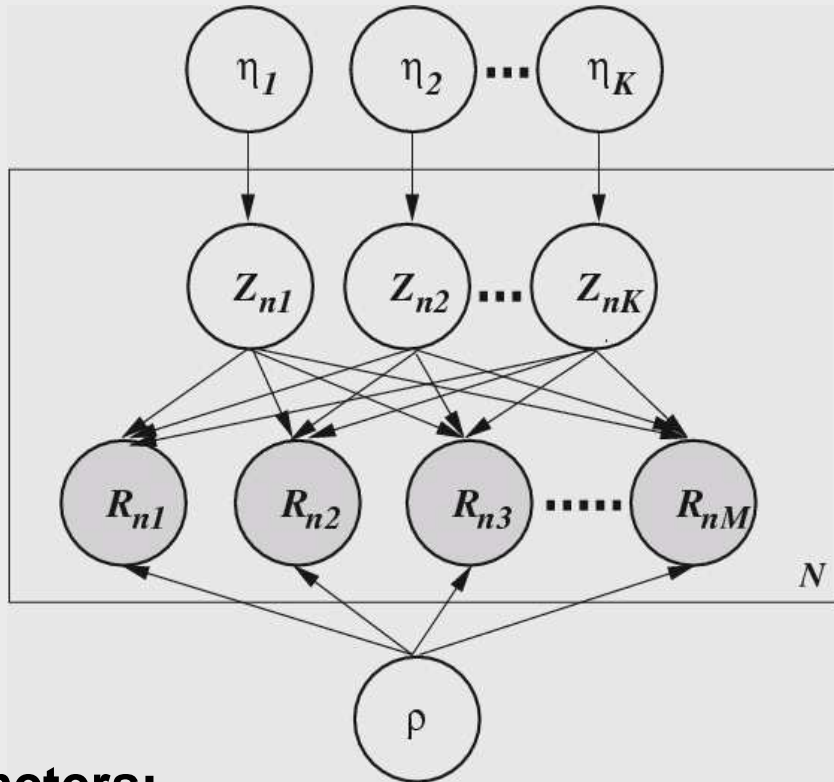
Features:

- 1. Directed graphical model:** Under the *missing at random* assumption a directed model can handle missing attribute values in the input without additional computation.
- 2. Multiple latent factors:** Multiple latent factors should underlie preferences, and they should all directly influence the rating for each item.
- 3. Multiplicative Combination :** A factor can specialize to a subset of items, and predictive distributions can become sharper when factors are combined.



Model Specification:

Graphical Model:



Parameters:

- η_k : Distribution over activity of factor k .
- ρ_{mk} : Distribution over rating values for item m according to factor k .

Variables:

- Z_{nk} : Activation level of factor k for user n .
- R_{nm} : Rating of item m for user n .

Binary/Multinomial MMF:

$$P(Z_k = z_k | \eta) = \eta_k$$

- Binary-valued, Bernoulli distributed factor activation levels Z_{nk} . η_k gives the mean of Bernoulli distribution for factor k .

$$P(R_m = v | k) = \frac{\exp(\rho_{vmk})}{\sum_{v'} \exp(\rho_{v'mk})}$$

- Ordinal-valued, multinomial distributed rating values R_{nm} . Each factor k has its own distribution over rating values v for each item m , ρ_{vmk} .

$$\begin{aligned} P(R_m = v | \mathbf{z}^n) &= \frac{\prod_{k=1}^K P(R_m = v | k)^{z_k}}{\sum_{v'=1}^V \prod_{k'=1}^K P(R_m = v' | k')^{z_{k'}}} \end{aligned}$$

- The combined distribution over each rating variable R_{nm} is obtained by multiplying together the factor distributions, taking into account the factor activation levels.

Learning

Variational Approximation

- Exact inference impractical for learning. We apply a standard mean-field approximation with a set of variational parameters μ_k for each user u .

$$P(\mathbf{z}|\mathbf{r}^u) = \frac{P(\mathbf{r}^u|\mathbf{z})P(\mathbf{z})}{\sum_{\mathbf{z}'} P(\mathbf{r}^u|\mathbf{z}')P(\mathbf{z}')} \quad Q(\mathbf{z}|\mathbf{r}^u, \mu^u) = \prod_{k=1}^K Q(z_k|\mu_k^u) = \prod_{k=1}^K (\mu_k^u)^{z_k} (1 - \mu_k^u)^{(1-z_k)}$$

Variational Inference

Input: $\mathbf{r}, \eta, \rho, I$

Output: μ

Initialize $\mu_k, \xi \leftarrow 1$

for $t = 1$ to I **do**

for $k = 1$ to K **do**

$$d_k \leftarrow \frac{\partial \tilde{F}[\mu, \eta, \rho]}{\partial \mu_k^n}$$

while $(\tilde{F}^n[\mathcal{P}(\mu^n - \xi d), \eta, \rho] > \tilde{F}^n[\mu^n, \eta, \rho])$ **do**

$$\xi \leftarrow \kappa \xi$$

$$\mu \leftarrow \mathcal{P}(\mu - \xi d)$$

Parameter Estimation

Input: $\{\mathbf{r}^u\}, K$

Output: η, ρ

Initialize $\eta, \rho, \xi^0 \leftarrow 1$

while Not Converged **do**

for $n = 1$ to N **do**

$$\mu^n \leftarrow \text{BinMMF-VarInf}(\eta, \rho, \mathbf{x}^n, H(n))$$

for $v = 1$ to $V, m = 1$ to $M, k = 1$ to K **do**

$$d_{vmk} \leftarrow \frac{\partial \tilde{F}[\mu, \eta, \rho]}{\partial \rho_{vmk}}$$

while $(\tilde{F}[\mu, \eta, \rho - \xi d] > \tilde{F}[\mu^t, \eta, \rho])$ **do**

$$\xi \leftarrow \kappa \xi$$

$$\rho \leftarrow \rho - \xi d$$

for $k = 1$ to K **do**

$$\eta_k \leftarrow \frac{1}{N} \sum_{n=1}^N \mu_k^n$$

Rating Prediction

Approximate Prediction:

1. Apply the variational approximation to the true posterior.
2. Compute predictive distribution using a small number of sampled factor activation vectors.

$$P^s(R_m = v | \mathbf{r}^a) = \sum_{s=1}^S P(R_m = v | \mathbf{z}^s) Q(\mathbf{z}^s | \mu^a)$$
$$P(R_m = v | \mathbf{r}^a) \approx \frac{P^s(R_m = v | \mathbf{r}^a)}{\sum_{v'=1}^V P^s(R_m = v' | \mathbf{r}^a)}$$

5. Experimentation

Weak Generalization Experiment:

- Available ratings for each user split into observed and unobserved sets. Trained on the observed ratings, tested on the unobserved ratings.

Strong Generalization Experiment:

- Users split into training set and testing set. Ratings for test users split into observed and unobserved sets. Trained on training users, tested on test users.

Data Sets:

EachMovie: Compaq Systems Research Center

- Users: 72916
- Ratings: 2,811,983
- Items: 1628
- Sparsity: 97.6%
- Rating Values: 6
- Filtering: 20 ratings

MovieLens: GroupLens Research Center

- Users: 6040
- Ratings: 1,000,209
- Items: 3900
- Sparsity: 95.7%
- Rating Values: 5
- Filtering: 20 ratings

Error Measure:

Normalized Mean Absolute Error:

- Average over all users of the absolute difference between predicted and actual ratings.
- Normalized by expectation of the difference between predicted and actual ratings under flat priors.

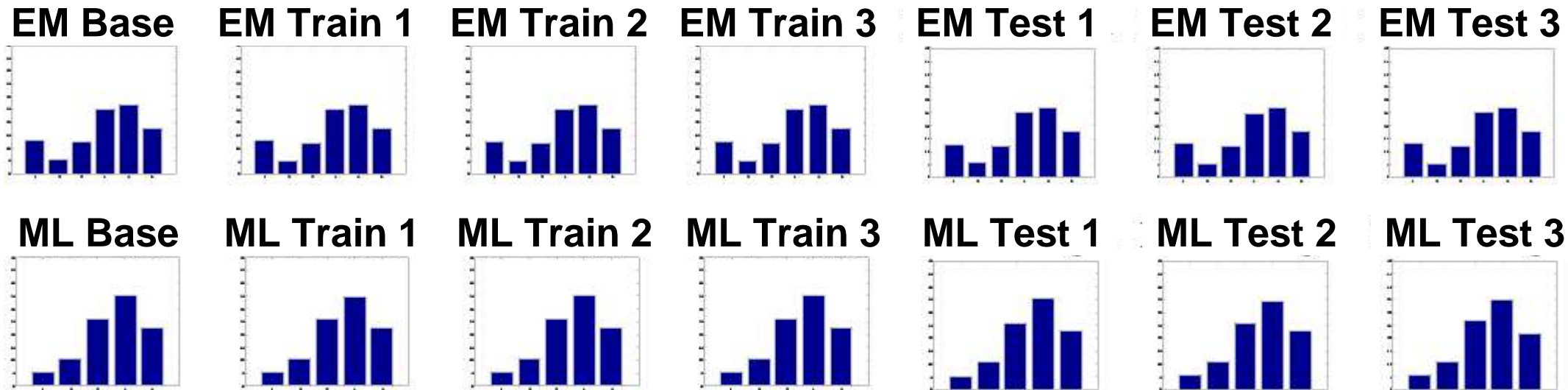


Figure 5: Distribution of ratings in filtered train and test data sets compared to base data sets.

6. Results

Prediction Performance:

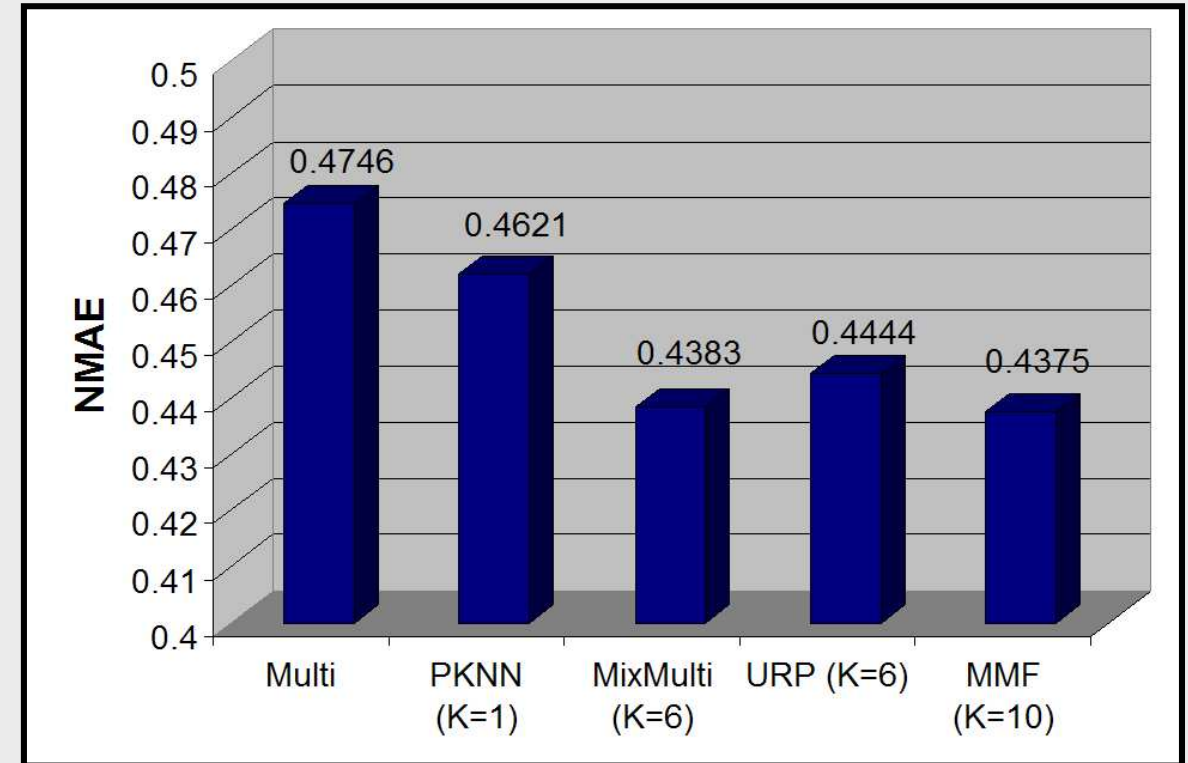
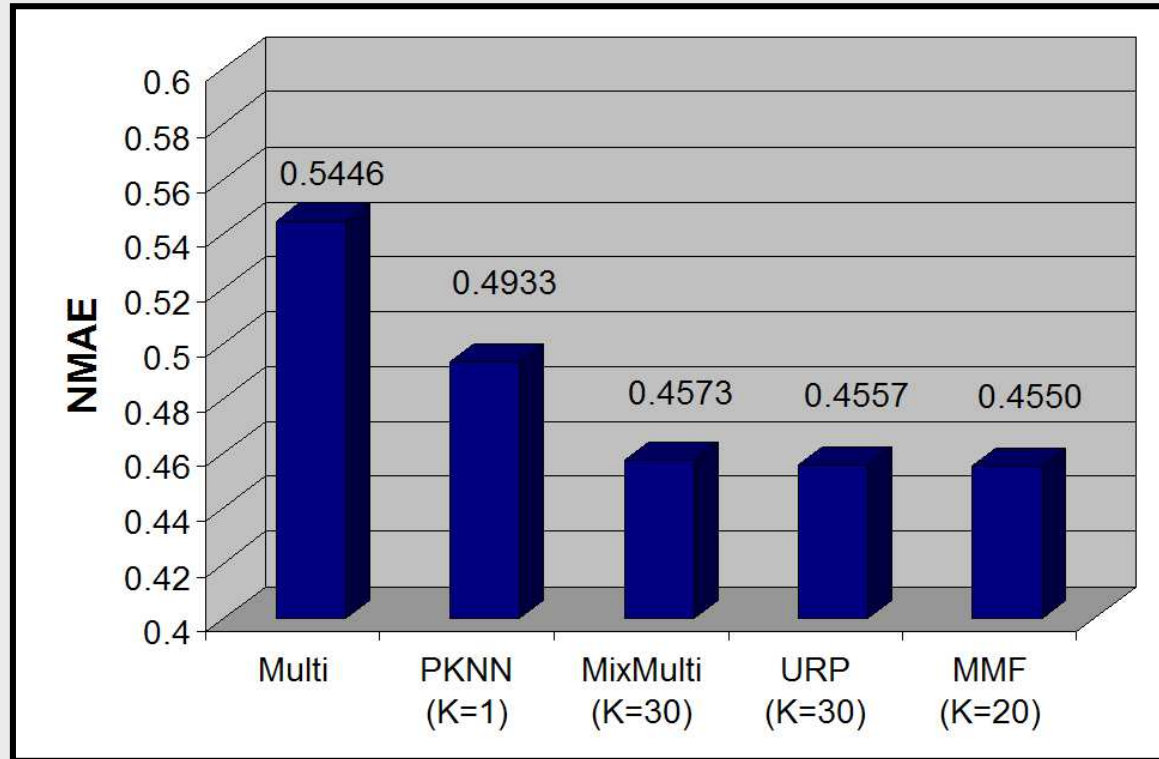


Figure 6: EachMovie Strong Generalization Results

Figure 7: MovieLens Strong Generalization Results

- MMF and URP attain the same minimum error rate on the EachMovie data set.
- On the MovieLens data set, MMF ties with the multinomial mixture model.

Multiplicative Factor Combination:

- Some factors make clear predictions about vote values while others are relatively uniform, indicating the presence of a specialization effect (Figure 8).
- Combining factor distributions multiplicatively can result in sharper distributions than any of the individual factors (Figure 9).

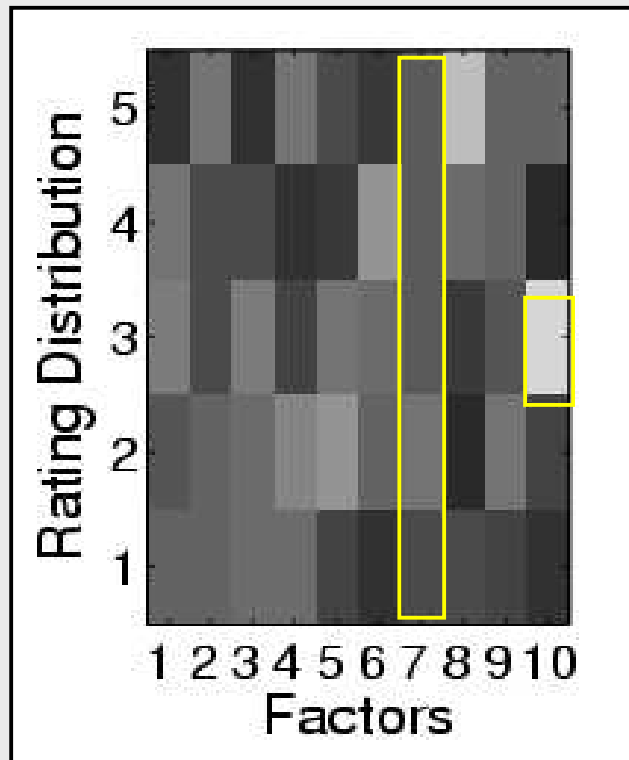


Figure 8: Learned factor distributions.

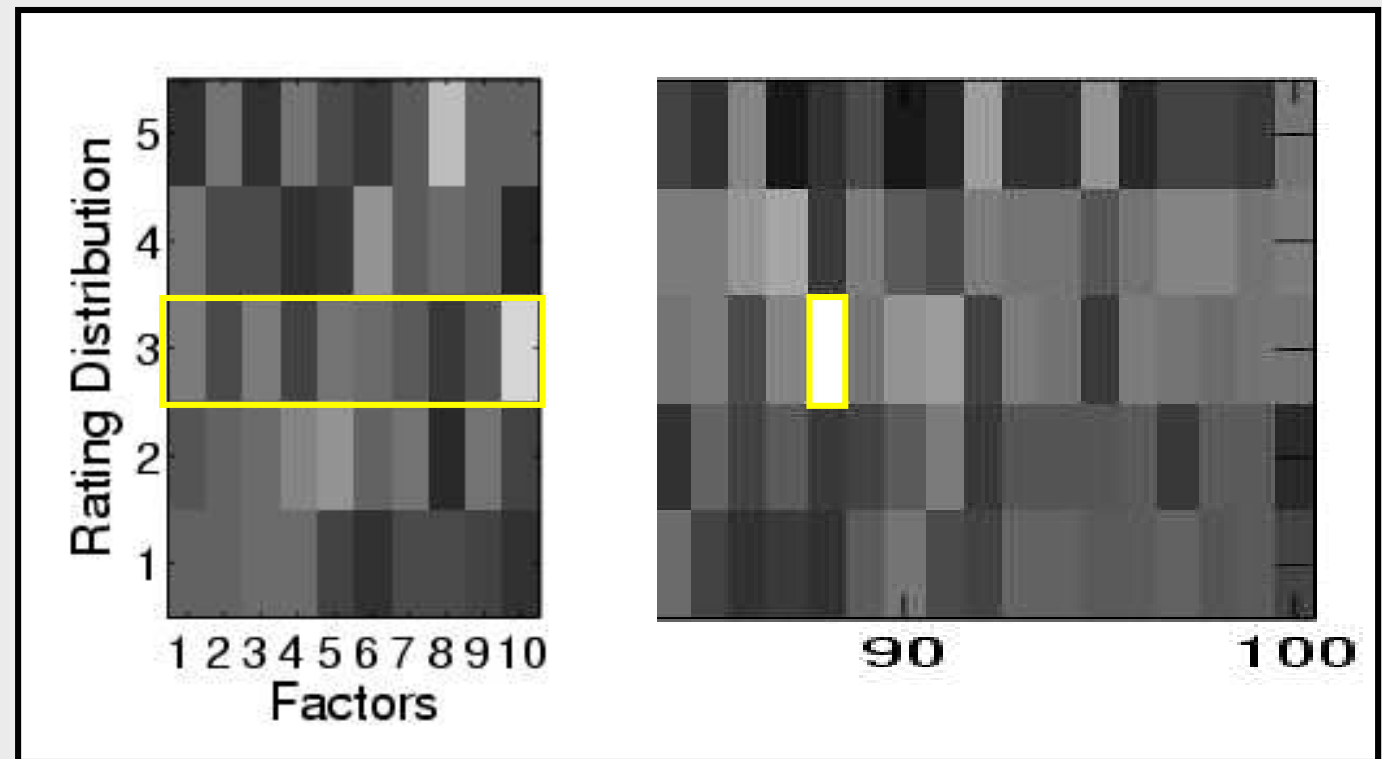


Figure 9: Learned factor distributions and predictive distribution for a particular item.

Factor Activation Levels:

- With our learning procedure, inferred factor activation levels tend to be quite sparse. This justifies using a relatively small number of samples for prediction.

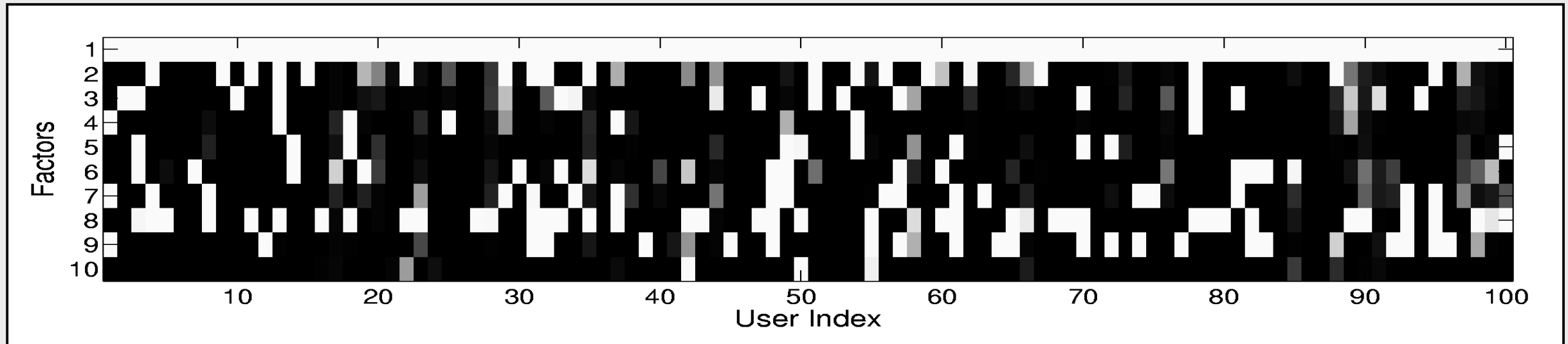


Figure 10: Factor activation levels for a random set of 100 users. Black indicates 0 probability that the factor is active, white indicates the factor is on with probability 1.

7. Conclusions and Future Work

Conclusions:

- Binary/Multinomial MMF differs from other CF models in that it combines the strength of directed models with the unique properties of multiplicative combination rules.
- Empirical results show that MMF matches the performance of the best known methods for collaborative filtering while learning an interesting, sparse representation of the data.
- Learning in MMF is computationally expensive, but can likely be improved.

Future Work:

- Extending the MMF architecture to model interdependence of latent factors.
- Studying other instances of the model like Integer/Multinomial and Binary/Gaussian.
- Studying additional applications like document modeling and analysis of micro arrays.