Learning User Dependencies for Recommendation

Yong Liu[†], Peilin Zhao[‡*], Xin Liu[§], Min Wu[†], Lixin Duan[♯], Xiao-Li Li[†]
[†]Institute for Infocomm Research, A*STAR, Singapore
[‡]Artificial Intelligence Department, Ant Financial Services Group, China
[§]Garena Online, Singapore
[♯]Big Data Research Center, University of Electronic Science and Technology of China, China
[†]{liuyo, wumin, xlli}@i2r.a-star.edu.sg, [‡]peilin.zpl@antfin.com,
[§]lucianliuxin@gmail.com, [♯]lxduan@uestc.edu.cn

Abstract

Social recommender systems exploit users' social relationships to improve recommendation accuracy. Intuitively, a user tends to trust different people regarding with different scenarios. Therefore, one main challenge of social recommendation is to exploit the most appropriate dependencies between users for a given recommendation task. Previous social recommendation methods are usually developed based on pre-defined user dependencies. Thus, they may not be optimal for a specific recommendation task. In this paper, we propose a novel recommendation method, named probabilistic relational matrix factorization (PRMF), which can automatically learn the dependencies between users to improve recommendation accuracy. In PRMF, users' latent features are assumed to follow a matrix variate normal (MVN) distribution. Both positive and negative user dependencies can be modeled by the row precision matrix of the MVN distribution. Moreover, we also propose an alternating optimization algorithm to solve the optimization problem of PRMF. Extensive experiments on four real datasets have been performed to demonstrate the effectiveness of the proposed PRMF model.

1 Introduction

Recommender systems have been widely used to help us discover useful information. For example, on e-commerce websites, *e.g.,* Amazon, recommender systems predict users' preferences on products based on their past purchasing behaviors and then recommend a user a list of interesting products she may prefer. For a recommender system, the most critical factor is the prediction accuracy of users' preferences. In practice, the most successful prediction methods are collaborative filtering based approaches [Su and Khoshgoftaar, 2009]. Especially, matrix factorization has been widely applied in different recommendation tasks [Koren *et al.*, 2009; Liu *et al.*, 2013; Hu *et al.*, 2014; Liu *et al.*, 2014; 2015].

The recent rapid developments of online social networking services (*e.g.,* Facebook and Twitter) motivate the emergences of social recommender systems that exploit users' online social friendships for recommendation [Yang *et al.*, 2014]. The social recommender systems usually assume that a user has similar interests with her social friends. Although the recommendation accuracy can usually be improved, there do not exist strong connections between the online friendship and the similarity of users' interests [Ma, 2014]. Because there are different categories of social networking friends in online social networks, *e.g.,* school friends, work-related friends, and friends sharing same interests or activities [Zhang *et al.*, 2013]. The tastes of a user's online friends usually vary significantly. In different scenarios, a user tends to trust recommendations from different subsets of her online social friends. Hence, the key to success for a social recommender system is to exploit the most appropriate dependencies between users for a specific recommendation task.

Moreover, existing social recommender systems are usually developed based on users' explicit social relationships, *e.g.,* trust relationships [Jamali and Ester, 2010; Fang *et al.*, 2014; Mei *et al.*, 2014; Guo *et al.*, 2015] and online social friendships [Ma *et al.*, 2011; Yang *et al.*, 2012; Tang *et al.*, 2013; Hu *et al.*, 2015]. However, users' explicit social relationships may be unavailable in many application scenarios. This limits the application of traditional social recommendation approaches. To remedy this problem, the underlying implicit social relationships between users that have most similar or dissimilar behaviors have also been exploited for improving recommendation accuracy [Ma, 2013]. In these studies, users' dependencies adopted for recommendation are pre-defined based on users' explicit or implicit social relationships.

Differing from previous work, this paper proposes a novel recommendation method, namely probabilistic relational matrix factorization (PRMF), which aims to automatically learn the user dependencies to improve the recommendation accuracy. The proposed method can be applied to the recommendation scenarios with or without users' explicit social relationships. In PRMF, the user latent feature matrix is assumed to follow a matrix variate normal (MVN) distribution [Gupta and Nagar, 1999], where the row precision matrix models the dependencies between different rows (*i.e.,* users). Therefore, both positive and negative dependencies between users can be learned by fitting the MVN distribution. This is motivated by the success of applying the MVN distribution to model

*Corresponding author

the task relationships for multi-task learning [Zhang and Yeung, 2010]. The optimization problem of PRMF is solved by an alternating algorithm based on the stochastic gradient descent (SGD) [Koren *et al.*, 2009] and alternating direction method of multipliers (ADMM) [Boyd *et al.*, 2011] methods. Moreover, we extensively evaluated PRMF and five baseline methods on four real datasets. Empirical experiments showed that PRMF consistently achieved the best recommendation accuracy on all datasets, in terms of root-mean-square error (RMSE) and mean absolute error (MAE).

2 Related Work and Background

This section first introduces some background about probabilistic matrix factorization (PMF) [Mnih and Salakhutdinov, 2007] and then reviews social recommendation methods.

2.1 Probabilistic Matrix Factorization

For the recommendation problem with m users $\{u_i\}_{i=1}^m$ and n items $\{v_j\}_{j=1}^n$, the matrix factorization models map users and items into a shared latent space, which has a low dimensionality $d \ll \min(m,n)$. For each user u_i, her latent features are represented by a latent vector $U_i \in \mathbb{R}^{1\times d}$. Similarly, the latent features of the item v_j are described by a latent vector $V_j \in \mathbb{R}^{1\times d}$. In PMF, users' ratings on items are assumed to follow a Gaussian distribution as follows:

$$p(R|U,V,\sigma^2) = \prod_{i=1}^{m}\prod_{j=1}^{n}\left[\mathcal{N}(R_{ij}|U_iV_j^\top,\sigma^2)\right]^{W_{ij}}, \quad (1)$$

where $R \in \mathbb{R}^{m\times n}$ is the matrix denoting users' ratings on items; $U \in \mathbb{R}^{m\times d}$ and $V \in \mathbb{R}^{n\times d}$denote the latent features of all users and items, respectively; σ^2 is the variance of the Gaussian distribution; W_{ij} is an indicator variable. If u_i has rated v_j, $W_{ij}=1$; otherwise, $W_{ij}=0$. In addition, we also place zero-mean spherical Gaussian priors on U and V as:

$$p(U|\sigma_u^2) = \prod_{i=1}^{m}\mathcal{N}(U_i|0,\sigma_u^2 I),\ \ p(V|\sigma_v^2) = \prod_{j=1}^{n}\mathcal{N}(V_j|0,\sigma_v^2 I). \quad (2)$$

where I is the identity matrix, σ_u^2 and σ_v^2 are the variance parameters. Through the Bayesian inference, we have

$$p(U,V|R,\sigma^2,\sigma_u^2,\sigma_v^2) \propto p(R|U,V,\sigma^2)p(U|\sigma_u^2)p(V|\sigma_v^2). \quad (3)$$

The model parameters (*i.e.,* U and V) can be learned via maximizing the log-posterior in Eq. (3), which is equivalent to solving the following problem:

$$\min_{U,V}\frac{1}{2}\|W\odot(R-UV^\top)\|_F^2 + \frac{\lambda_u}{2}\|U\|_F^2 + \frac{\lambda_v}{2}\|V\|_F^2, \quad (4)$$

where $W \in \mathbb{R}^{m\times n}$ is the indicator matrix, and W_{ij} is the (i,j) entry of W. In Eq. (4), $\odot$ denotes the Hadamard product of two matrices, $\lambda_u = \sigma^2/\sigma_u^2$, $\lambda_v = \sigma^2/\sigma_v^2$, and $\|\cdot\|_F$ denotes the Frobenius norm of a matrix.

2.2 Social Recommendation Methods

In recent years, lots of social recommendation approaches have been proposed [Yang *et al.*, 2014]. The Social Regularization (SR) [Ma *et al.*, 2011] is one of the most representative methods. The SR method was implemented by matrix factorization framework. It assumed that a user may have similar interests with her social friends, and thus the learned latent features of a user and that of her social friends should be similar. As a user tended to trust different subsets of her social friends regarding with different domains, the circle-based recommendation (CircleCon) models proposed in [Yang *et al.*, 2012] considered domain-specific trust circles for recommendation. In [Tang *et al.*, 2013], both the local and global social contexts were exploited for recommendation. Recently, the SR model was extended to exploit users' implicit social relationships for recommendation [Ma, 2013]. The implicit social relationships were defined between a user and other users that had most similar or dissimilar rating behaviors with her. In [Fang *et al.*, 2014], users' social relationships were studied from four different trust aspects for recommendation. In [Li *et al.*, 2015], SR was also extended to exploit the community structure of users' social networks to improve recommendation accuracy. Moreover, the recent work [Guo *et al.*, 2015] developed the TrustSVD model, which incorporated both user and item biases, as well as the explicit and implicit influences from rated items and trusted users for recommendation. In [Hu *et al.*, 2015], a recommendation framework named MR3 was proposed to jointly model users' rating behaviors, social relationships, and review comments.

3 The Proposed Recommendation model

This section presents the details of the proposed PRMF model that jointly learn users' preferences and the dependencies between users for recommendation.

3.1 Probabilistic Relational Matrix Factorization

In Section 2.1, the PMF model assumes the users are independent of each other (see Eq.(2)). Thus, it ignores the dependencies between users. However, in practice, users are usually connected with each other through different types of relationships, *e.g.,* trust relationships and online social relationships. In this work, to exploit users' dependencies for recommendation, we place the following priors on the user latent features:

$$p(U|\sigma_u^2) \propto \big(\prod_{i=1}^{m}\mathcal{N}(U_i|0,\sigma_u^2 I)\big)q(U), \quad (5)$$

where the first term of the priors is used to penalize the complexity of the latent features of each user, and the second term $q(U)$ is used to model the dependencies between different users. Specifically, we define

$$q(U) = \mathcal{MN}_{m\times d}\big(0,\Theta^{-1},I\big),\ \ s.t.\ \Theta \succ 0, \quad (6)$$

where $\mathcal{MN}_{a\times b}(M,A,B)$ denotes the matrix variate normal (MVN) distribution[1] with mean $M \in \mathbb{R}^{a\times b}$, row covariance $A \in \mathbb{R}^{a\times a}$, and column covariance $B \in \mathbb{R}^{b\times b}$; $\Theta \succ 0$ indicates Θ is a positive definite matrix. In Eq. (6), Θ is the row

[1]The density function for a random matrix X following the MVN distribution $\mathcal{MN}_{a\times b}(M,A,B)$ is

$$p(X) = \frac{\exp(-\frac{1}{2}\mathrm{tr}\big[B^{-1}(X-M)^\top A^{-1}(X-M)\big])}{(2\pi)^{ab/2}|B|^{a/2}|A|^{b/2}}.$$

precision matrix (*i.e.,* the inverse of row covariance matrix) that models the dependencies between different rows of U. The row U_i is independent of another row U_k, if and only if $\Theta_{ik} = 0$ [Liu *et al.*, 2009]. Note that each row of U represents the latent features of an individual user. Thus, in other words, Θ describes the dependencies between different users. In this work, Θ is called the user dependency matrix. For simplicity, we set the column covariance matrix of the MVN distribution as I, which indicates the user latent features in different dimensions are independent. Then, we rewrite Eq. (5) as:

$$p(U|\Theta, \sigma_u^2) = \mathcal{MN}_{m\times d}\big(0, (\Theta + \frac{1}{\sigma_u^2}I)^{-1}, I\big). \quad (7)$$

In practice, a user usually only depends with a small fraction of other users. Thus, it is reasonable to assume the user dependency matrix Θ is sparse. To achieve this objective, we introduce a sparsity-inducing prior for Θ as follows:

$$p(\Theta|\gamma) \propto \exp(-\frac{\gamma}{2}\|\Theta\|_1), \quad (8)$$

where γ is a positive constant used to control the sparsity of Θ, and $\|\cdot\|_1$ is the ℓ_1-norm of a matrix. In addition, the sparsity of Θ can also help improve the computation efficiency.

Moreover, we also assume users' ratings follow the Gaussian distribution in Eq. (1) and add Gaussian priors on the item latent features as in Eq. (2). Through the Bayesian inference, we have

$$\begin{aligned} &p(U, V, \Theta|R, \sigma^2, \sigma_u^2, \sigma_v^2, \gamma) \\ \propto &p(R|U, V, \sigma^2)p(U|\Theta, \sigma_u^2)p(\Theta|\gamma)p(V|\sigma_v^2). \end{aligned} \quad (9)$$

Then, the model parameters (*i.e.,* U, V, and Θ) can be obtained by solving the following problem:

$$\begin{aligned} \min_{U,V,\Theta\succ 0} &\frac{1}{2\sigma^2}\|W \odot (R - UV^\top)\|_F^2 + \frac{1}{2\sigma_u^2}\|U\|_F^2 + \frac{1}{2\sigma_v^2}\|V\|_F^2 \\ &+\frac{1}{2}\big[\mathrm{tr}(U^\top\Theta U) - d\log|\Theta + \frac{1}{\sigma_u^2}I| + \gamma\|\Theta\|_1\big], \end{aligned} \quad (10)$$

where $|\cdot|$ denotes the determinant of a matrix. Here, the model parameters are learned without prior information about users' relationships.

Incorporating prior user relationship information

Previous studies have demonstrated that user' explicit social relationships [Yang *et al.*, 2014] and implicit social relationships derived from users' rating behaviors [Ma, 2013] can benefit recommendation accuracy. These prior relationship information contains some prior knowledge about users' dependencies in a recommendation task. Let $\Sigma \in \mathbb{R}^{m\times m}$ denote the covariance matrix calculated based on users' rating behaviors. The elements of Σ are defined as follows:

$$\Sigma_{ik} = cov(R_{i*}, R_{k*}), \quad (11)$$

where R_{i*} is the ith row in R; $cov(x_1, x_2)$ denotes the covariance between two observations. If users' explicit social relationships are available, we consider the following sparse covariance matrix Σ:

$$\Sigma_{ik} = \begin{cases} cov(R_{i*}, R_{k*}) & \text{if } u_k \in \mathcal{F}(u_i) \text{ or } u_i \in \mathcal{F}(u_k) \text{ or i=k}, \\ 0 & \text{otherwise}, \end{cases}$$

where $\mathcal{F}(u_i)$ denotes the set of u_i's social friends. Then, users' dependencies in the rating space[2] can be described using the precision matrix Σ^{-1}.

To incorporate the prior information, we assume that the learned user dependencies in latent space are close to the user dependencies derived from prior relationship information in the rating space. This is achieved by minimizing the LogDet divergence [Davis *et al.*, 2007] between the row precision matrix in Eq. (7) and the precision matrix Σ^{-1} as:

$$\begin{aligned} &\min_{\Theta} D_{ld}(\Theta + \frac{1}{\sigma_u^2}I, \Sigma^{-1}) \\ &= \mathrm{tr}[(\Theta + \frac{1}{\sigma_u^2}I)\Sigma] - \log\big|(\Theta + \frac{1}{\sigma_u^2}I)\Sigma\big| - m \\ &\propto \mathrm{tr}\big(\Theta\Sigma\big) - \log|(\Theta + \frac{1}{\sigma_u^2}I)|. \end{aligned} \quad (12)$$

Note that other distance metrics can also be used to estimate the matrix closeness. Moreover, although Σ may be singular, the proposed method can also be used in practice (*i.e.,* no computation of Σ^{-1} in Eq. (12)).

The unified model

Considering both the constraints in Eq. (10) and Eq. (12), we formulate the final objective function of the proposed PRMF model as follows:

$$\begin{aligned} \min_{U,V,\Theta\succ 0} &\frac{1}{2}\|W \odot (R - UV^\top)\|_F^2 + \frac{\lambda_u}{2}\|U\|_F^2 + \frac{\lambda_v}{2}\|V\|_F^2 \\ &+\frac{\alpha}{2}\big\{\mathrm{tr}\big[\Theta(UU^\top + \beta\Sigma)\big] - (d+\beta)\log\big|\Theta + \frac{\lambda_u}{\alpha}I\big| \\ &+\gamma\|\Theta\|_1\big\}, \end{aligned} \quad (13)$$

where $\lambda_u = \sigma^2/\sigma_u^2$, $\lambda_v = \sigma^2/\sigma_v^2$, and $\alpha = \sigma^2$; β is a parameter controlling the contribution of prior information.

3.2 Optimization Algorithm

The problem in Eq. (13) can be solved by the following alternating algorithm.

Optimize U and V: By fixing Θ, the optimization problem in Eq. (13) becomes:

$$\begin{aligned} \min_{U,V} &\frac{1}{2}\|W \odot (R - UV^\top)\|_F^2 + \frac{\lambda_u}{2}\|U\|_F^2 + \frac{\lambda_v}{2}\|V\|_F^2 \\ &+\frac{\alpha}{2}\mathrm{tr}(U^\top\Theta U). \end{aligned} \quad (14)$$

The optimization problem in Eq. (14) can be solved using the SGD algorithm.The updating rules are as follows:

$$\begin{aligned} U_i &\leftarrow U_i + \theta\big(\Delta_{ij}V_j - \lambda_u U_i - \alpha\Theta_{i*}U\big) \\ V_j &\leftarrow V_j + \theta\big(\Delta_{ij}U_i - \lambda_v V_j\big), \end{aligned} \quad (15)$$

where $\Delta_{ij} = R_{ij} - U_iV_j^\top$, Θ_{i*} is the ith row of Θ, and θ is the learning rate. Note the SGD updates are only performed on the observed rating pairs $\mathcal{D} = \{(u_i, v_j, R_{ij})|W_{ij} > 0\}$.

Optimize Θ: By fixing U and V, the optimization problem with respect to Θ is as follows:

$$\min_{\Theta\succ 0} \mathrm{tr}\big[\Theta(UU^\top + \beta\Sigma)\big] - (d+\beta)\ \log|\Theta + \frac{\lambda_u}{\alpha}I| + \gamma\|\Theta\|_1. \quad (16)$$

[2]Each user is represented by a vector of her ratings on items.

The optimization problem in Eq. (16) is convex with respect to Θ. Therefore, the optimal solution $\widehat{\Theta}$ to Eq. (16) satisfies:

$$(\widehat{\Theta} + \frac{\lambda_u}{\alpha} I)^{-1} - \frac{1}{d+\beta}(UU^\top + \beta\Sigma) = \frac{\gamma}{d+\beta}\widehat{G}, \quad (17)$$

where $\widehat{G}$ is the sub-gradient of $\|\Theta\|_1$ over Θ, and the elements of $\widehat{G}$ are in $[-1, 1]$. Thus, according to the derivation of Dantzig estimator [Candes and Tao, 2007], we drop the constraint $\Theta \succ 0$ and consider the following problem:

$$\min \|\Theta\|_1 \quad s.t.$$
$$\big\|(\Theta + \frac{\lambda_u}{\alpha} I)^{-1} - \frac{1}{d+\beta}(UU^\top + \beta\Sigma)\big\|_\infty \leq \frac{\gamma}{d+\beta}. \quad (18)$$

By multiplying $\Theta + \frac{\lambda_u}{\alpha} I$ with the constraint, we obtain the following relaxation of Eq. (18):

$$\min \|\Theta\|_1 \;\; s.t. \;\; \big\|C\Theta - E\big\|_\infty \leq \tau, \quad (19)$$

where $C = \frac{1}{d+\beta}(UU^\top + \beta\Sigma)$, $E = I - \frac{\lambda_u}{\alpha} C$, and $\tau = \frac{\gamma}{d+\beta}$. This relaxation has been used in the constrained ℓ_1-minimization for inverse matrix estimation (CLIME) [Cai *et al.*, 2011]. Indeed, the optimization in Eq. (19) is equivalent to the following optimization problem [Liu and Luo, 2014]:

$$\min_{\Theta} \mathrm{tr}\big(\Theta^\top C\Theta\big) - \mathrm{tr}(E\Theta) + \tau\|\Theta\|_1. \quad (20)$$

Let $\widetilde{\Theta}$ be the solution of Eq. (20), which is not necessarily symmetric. We use the following symmetrization step to obtain the final $\widehat{\Theta}$:

$$\widehat{\Theta}_{ik} = \widehat{\Theta}_{ki} = \widetilde{\Theta}_{ik}\mathbb{I}(|\widetilde{\Theta}_{ik}| \leq |\widetilde{\Theta}_{ki}|) + \widetilde{\Theta}_{ki}\mathbb{I}(|\widetilde{\Theta}_{ik}| > |\widetilde{\Theta}_{ki}|), \quad (21)$$

where $\mathbb{I}(\cdot)$ is an indicator function, which is equal to 1 if the condition is satisfied, and otherwise 0. In addition, although constraint $\Theta \succ 0$ is not in Eq. (20), the symmetrized $\widehat{\Theta}$ is positive definite with high probability and converge to the solution of Eq. (16) under the spectral norm, guaranteed by Remark 1 and Theorem 1 in [Liu and Luo, 2014].

The problem in Eq. (20) can be solved by the ADMM algorithm [Boyd *et al.*, 2011]. The augmented Lagrangian of Eq. (20) is as follows:

$$L_\rho(\Theta, Z, Y) = \mathrm{tr}\big(Z^\top CZ\big) - \mathrm{tr}(EZ) + \tau\|\Theta\|_1 + \langle Y, \Theta - Z\rangle + \frac{\rho}{2}\|\Theta - Z\|_F^2, \quad (22)$$

where Y is a scaled dual variable and $\rho > 0$. We can obtain the following ADMM updates:

$$\Theta^{t+1} = \arg\min \frac{\rho}{2}\|\Theta - Z^t + Y^t\|_F^2 + \tau\|\Theta\|_1, \quad (23)$$

$$Z^{t+1} = \arg\min \frac{\rho}{2}\|\Theta^{t+1} - Z + Y^t\|_F^2 + \mathrm{tr}(Z^\top CZ) - \mathrm{tr}(EZ), \quad (24)$$

$$Y^{t+1} = Y^t + \Theta^{t+1} - Z^{t+1}. \quad (25)$$

The closed solution to Eq. (23) is as follows:

$$\Theta^{t+1} = soft\big[Z - Y, \frac{\tau}{\rho}\big], \quad (26)$$

Algorithm 1: PRMF Optimization Algorithm

Input : $\mathcal{D}$, Σ, d, λ_u, λ_v, α, β, γ, θ, ρ
Output: U, V, Θ

1 **if** $\beta > 0$ **then**
2 $\quad X \leftarrow SVD(\Sigma, d)$;
3 Initialize U and V randomly, and set $\Theta = I$;
4 **for** $iter = 1, 2, \ldots, max_iter$ **do**
5 $\quad$ **for** $t = 1, 2, \ldots, T$ **do**
6 $\quad\quad$ **foreach** $(u_i, v_j, R_{ij}) \in \mathcal{D}$ **do**
7 $\quad\quad\quad U_i \leftarrow U_i + \theta\big(\Delta_{ij}V_j - \lambda_u U_i - \alpha\Theta_{i*}U\big)$;
8 $\quad\quad\quad V_j \leftarrow V_j + \theta\big(\Delta_{ij}U_i - \lambda_v V_j\big)$;
9 $\quad$ **if** $\beta = 0$ **then**
10 $\quad\quad \widehat{U} = \frac{1}{\sqrt{d}}U$; $\tau = \frac{\gamma}{d}$;
11 $\quad$ **else**
12 $\quad\quad \widehat{U} = [\frac{1}{\sqrt{d+\beta}}U, \frac{\sqrt{\beta}}{\sqrt{d+\beta}}X]$; $\tau = \frac{\gamma}{d+\beta}$;
13 $\quad Z^0 = \Theta$; $Y^0 = 0$; $E = I - \frac{\lambda_u}{\alpha}\widehat{U}\widehat{U}^\top$;
14 $\quad P = I - \frac{1}{\rho}\widehat{U}(I + \frac{1}{\rho}\widehat{U}^\top\widehat{U})^{-1}\widehat{U}^\top$;
15 $\quad$ **for** $t = 0, 1, \ldots, K-1$ **do**
16 $\quad\quad \Theta^{t+1} = soft\big[Z^t - Y^t, \frac{\tau}{\rho}\big]$;
17 $\quad\quad Z^{t+1} = P(\frac{1}{\rho}E + \Theta^{t+1} + Y^t)$;
18 $\quad\quad Y^{t+1} = Y^t + \Theta^{t+1} - Z^{t+1}$;
19 $\quad$ Compute Θ by symmetrizing Θ^K using Eq.(21);

where

$$soft\big[A, \lambda\big] = \begin{cases} A_{ij} - \lambda & \text{if } A_{ij} > \lambda, \\ A_{ij} + \lambda & \text{if } A_{ij} < -\lambda, \\ 0 & \text{otherwise.} \end{cases} \quad (27)$$

The solution to Eq. (24) is as follows:

$$Z = (\frac{1}{\rho}C + I)^{-1}(\frac{1}{\rho}E + \Theta + Y). \quad (28)$$

The time complexity of the inverse operation in Eq. (28) is $\mathcal{O}(m^3)$. As Σ is symmetric, to improve the computation efficiency, we consider the following approximation $\Sigma \approx XX^\top$, where $X \in \mathbb{R}^{m\times d}$. In this paper, we obtain X by factorizing Σ using SVD and choosing the singular vectors with respect to the top d largest singular values (see line 2 in Algorithm 1). Let $\widehat{U} = [\frac{1}{\sqrt{d+\beta}}U, \frac{\sqrt{\beta}}{\sqrt{d+\beta}}X] \in \mathbb{R}^{m\times 2d}$. When the prior covariance matrix Σ is not available, we set $\widehat{U} = \frac{1}{\sqrt{d}}U$. Then, $C \approx \widehat{U}\widehat{U}^\top$. Following Woodbury matrix identity, we have

$$(\frac{1}{\rho}C + I)^{-1} \approx I - \frac{1}{\rho}\widehat{U}(I + \frac{1}{\rho}\widehat{U}^\top\widehat{U})^{-1}\widehat{U}^\top. \quad (29)$$

Using this matrix operation, the time complexity of the update of Z in Eq. (28) becomes $\mathcal{O}(dm^2)$. The details of the optimization algorithm are summarized in Algorithm 1. At each iteration, the time complexity of the SGD updates is $\mathcal{O}(T \cdot |\mathcal{D}| \cdot \bar{m} \cdot d)$, where $\bar{m}$ denotes the average number of nonzero elements in a row of Θ. Thus, the sparsity of Θ can help improve the computation efficiency of the proposed method. The time complexity of the ADMM updates is $\mathcal{O}(K \cdot d \cdot m^2)$.

4 Experiments

We conduct empirical experiments on real datasets to demonstrate the effectiveness of the proposed method.

4.1 Experimental Setting

Datasets

The experiments are performed on four public datasets: MovieLens-100K, MovieLens-1M[3], Ciao, and Epinions[4]. MovieLens-100K contains 100,000 ratings given by 943 users to 1,682 movies. MovieLens-1M consists of 1,000,209 ratings given by 6,040 users to 3,706 movies. We denote these two datasets by ML-100K and ML-1M, respectively. For the Ciao and Epinions datasets, we remove the items that have less than 3 ratings. Finally, on Ciao dataset, we have 185,042 ratings given by 7,340 users to 21,881 items. On Epinions dataset, there are 642,104 ratings given by 22,112 users to 59,104 items. The densities of the rating matrices on these datasets are 6.30% for ML-100K, 4.47% for ML-1M, 0.12% for Ciao, and 0.05% for Epinions. Moreover, on Ciao and Epinions datasets, we have observed 111,527 and 353,419 social relationships between users. The densities of the social relation matrices are 0.21% for Ciao and 0.07% for Epinions. Table 1 summarizes the details of the experimental datasets. For each dataset, 80% of the observed ratings are randomly sampled for training, and the remaining 20% observed ratings are used for testing.

Evaluation metrics

The performances of the recommendation algorithms are typically evaluated by two most popular metrics: mean absolute error (MAE) and root mean square error (RMSE), which are defined as follows:

$$MAE = \frac{1}{|\mathcal{D}_{test}|} \sum_{(u_i,v_j)\in\mathcal{D}_{test}} |R_{ij} - \hat{R}_{ij}|, \quad (30)$$

$$RMSE = \sqrt{\frac{1}{|\mathcal{D}_{test}|} \sum_{(u_i,v_j)\in\mathcal{D}_{test}} (R_{ij} - \hat{R}_{ij})^2}, \quad (31)$$

where R_{ij} denotes observed rating in testing data, $\hat{R}_{ij}$ is the predicted rating, and $\mathcal{D}_{test}$ denotes the set of tested ratings.

Evaluated recommendation methods

We compare the following methods: (1) ***PMF***: This is the probabilistic matrix factorization model [Mnih and Salakhutdinov, 2007]. (2) ***SR***^{imp}: This is the SR method that exploits users' implicit social relationships for recommendation [Ma, 2013]; (3) ***SR***^{exp}: This is the SR method that exploits users' explicit social relationships for recommendation [Ma *et al.*, 2011]; (4) ***LOCABAL***: This is the social recommendation model that exploits both local and global social contexts for recommendation [Tang *et al.*, 2013]; (5) ***eSMF***: This method extends LOCABAL to consider the graph structure of social neighbors for recommendation [Hu *et al.*, 2015]; (6) ***PRMF***: This method learns users' dependencies without prior information. The objective function is Eq. (10); (7) ***PRMF***^{imp}: This is the proposed method that exploits users' implicit social relationships as prior knowledge to learn users' dependencies; (8) ***PRMF***^{exp}: This is the proposed method exploiting users' explicit social relationships as prior knowledge to learn users' dependencies.

[3]http://grouplens.org/datasets/movielens/

[4]http://www.public.asu.edu/ jtang20/datasetcode/truststudy.htm

Table 1: The statistics of the experimental datasets.

	ML-100K	ML-1M	Ciao	Epinions
#Users	943	6,040	7,340	22,112
#Items	1682	3,706	21,881	59,104
#Ratings	100,000	1,000,209	185,042	642,104
Rat. Den.	6.30%	4.47%	0.12%	0.05%
#Soc. Rel.	N.A.	N.A.	111,527	353,419
Soc. Den.	N.A.	N.A.	0.21%	0.07%

Table 2: Performance comparisons on datasets without users' explicit social relationships.

Dataset	Method	RMSE	MAE
ML-100K	PMF	0.9251±0.0021	0.7314±0.0017
	SR^{imp}	0.9205±0.0013	0.7290±0.0008
	PRMF	0.9157±0.0006	0.7226±0.0005
	PRMF^{imp}	**0.9132**±0.0003	**0.7210**±0.0005
ML-1M	PMF	0.8728±0.0029	0.6807±0.0023
	SR^{imp}	0.8621±0.0008	0.6739±0.0006
	PRMF	0.8592±0.0009	0.6738±0.0010
	PRMF^{imp}	**0.8572**±0.0009	**0.6727**±0.0010

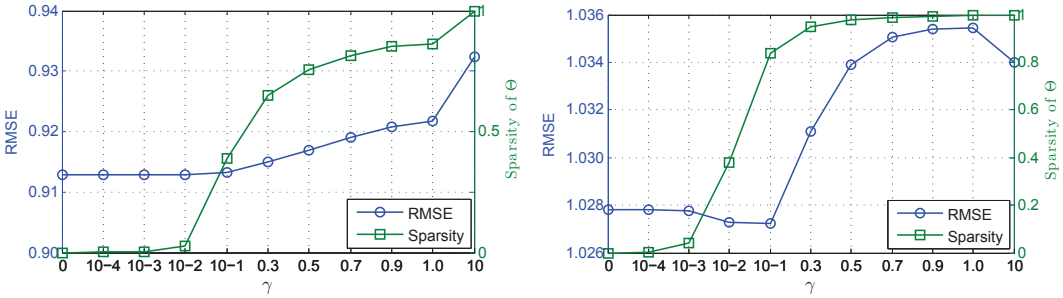
Parameter settings

Cross-validation is adopted to choose the parameters for each evaluated algorithm. The validation data is constructed by randomly chosen 10% of the ratings in the training data. For matrix factorization methods, the dimensionality of latent space d is set to 10. The latent features of users and items are randomly initialized by a Gaussian distribution with mean 0 and standard deviation $1/\sqrt{d}$. Moreover, we set the regularization parameters $\lambda_u = \lambda_v$ and choose the parameters from $\{10^{-5}, 10^{-4}, \cdots, 10^{-1}\}$. For PRMF, α is chosen from $\{2^{-5}, 2^{-4}, \cdots, 2^{-1}\}$, θ is chosen from $\{2^{-5}, 2^{-4}, \cdots, 2^{-1}\}$. For simplicity, we empirically set $\gamma = 10^{-4}$, $\beta = 10$, and $\rho = 100$. For SR methods, the regularization parameter α is chosen from $\{2^{-7}, 2^{-6}, \cdots, 2^{0}\}$. The user similarity is computed using Pearson correlation coefficient, and we set the threshold of the user similarity at 0.75 and $N = 10$, following [Ma, 2013]. The parameters of LOCABAL and eSMF are set following [Tang *et al.*, 2013] and [Hu *et al.*, 2015]. For each algorithm, we repeated the experiments five times with different random seeds. The results reported are average of the five runs.

4.2 Summary of Experimental Results

Table 2 summarizes the experimental results on the datasets without users' explicit social relationships, and Table 3 summarizes the results on the datasets with users' explicit social relationships. We make the following observations:

- On all datasets, PRMF outperforms PMF by 0.94% on ML-100K, 1.36% on ML-1M, 7.05% on Ciao, and 5.16% on Epinions, in terms of RMSE. This indicates the recommendation accuracy can be improved by jointly learning users' preferences and users' dependencies.



(a) ML-100K (b) Ciao

Figure 1: Performance trend of PRMF^{imp} on the ML-100K and Ciao datasets measured by RMSE with different settings of γ.

Table 3: Performance comparisons on datasets with users' explicit social relationships.

Dataset	Method	RMSE	MAE
Ciao	PMF	1.1031±0.0045	0.8439±0.0034
	SR^{imp}	1.0597±0.0042	0.8234±0.0036
	SR^{exp}	1.0286±0.0029	**0.7951**±0.0020
	LOCABAL	1.0777±0.0039	0.8330±0.0022
	eSMF	1.0592±0.0043	0.8122±0.0010
	PRMF	1.0326±0.0030	0.8006±0.0023
	PRMF^{imp}	1.0305±0.0030	0.7993±0.0023
	PRMF^{exp}	**1.0258**±0.0035	0.7980±0.0025
Epinions	PMF	1.1613±0.0022	0.8932±0.0019
	SR^{imp}	1.1321±0.0054	0.8874±0.0016
	SR^{exp}	1.1263±0.0027	0.8795±0.0023
	LOCABAL	1.1289±0.0008	0.8734±0.0010
	eSMF	1.1306±0.0025	0.8749±0.0025
	PRMF	1.1097±0.0024	0.8703±0.0022
	PRMF^{imp}	**1.1081**±0.0023	**0.8691**±0.0022
	PRMF^{exp}	1.1085±0.0024	0.8695±0.0022

- Compared with SR^{imp}, PRMF^{imp} achieves better results on all datasets. For example, in terms of RMSE, PRMF^{imp} outperforms SR^{imp} by 0.73%, 0.49%, 2.92%, and 2.40%, respectively. This shows that users' dependencies learned from the rating data are more effective than the user dependencies pre-defined based on users' implicit social relationships, for improving the recommendation accuracy.
- The proposed PRMF^{exp} method outperforms several existing social recommendation methods that exploit users' explicit social relationships for recommendation. On the Ciao and Epinions datasets, the average improvements of PRMF^{exp} over SR^{exp}, LOCABAL, and eSMF, in terms of RMSE, are 1.03%, 3.60%, 2.77%, respectively. This again demonstrates the effectiveness of the proposed method.
- PRMF^{imp} and PRMF^{exp} outperforms PRMF on all datasets. This observation consistently indicates the prior information about user dependencies (*i.e.,* users' explicit and implicit social relationships) are beneficial in improving recommendation accuracy. However, the improvements are not very significant. One potential reason is the SVD factorization used in Algorithm 1 (line 2) may not accurately approximate the prior covariance matrix.

In addition, we also study the impact of the sparsity of the learned user dependency matrix Θ on the recommendation accuracy. We choose the regularization parameter γ from $\{0, 10^{-4}, 10^{-3}, 10^{-2}, 0.1, 0.3, 0.5, 0.7, 0.9, 1.0, 10\}$. Figure 1 shows the performance trend of PRMF^{imp} on the ML-100K and Ciao datasets, in terms of RMSE. Observed that the sparsity of Θ increases with the increase of γ. As shown in Figure 1(a), denser user dependency matrix generally achieves better recommendation accuracy. However, denser user dependency matrix causes more computation time used to learn the user latent features. Indeed, there exists some balance between the computation efficiency and the recommendation accuracy. For example, on the ML-100K dataset, by setting γ to 0.3, the sparsity of the learned Θ is 65.21%, and the RMSE value is 0.9149, which is 0.56% better than the best competitor SR^{imp}. Moreover, Figure 1(b) also indicates that better recommendation accuracy may be achieved by learning a sparse Θ. For example, on the Ciao dataset, the best recommendation accuracy is achieved by setting γ to 0.1, and the sparsity of the learned Θ is 83.67%. On the ML-1M and Epinions datasets, we have similar observations with that on the ML-100K dataset. Due to space limitation, we do not report those results here.

5 Conclusion and Future Work

In this paper, we propose a novel recommendation method, namely probabilistic relational matrix factorization (PRMF). For a specific recommendation task, the proposed approach jointly learns users' preferences and the dependencies between users, to improve the recommendation accuracy. Empirical results on real datasets demonstrate the effectiveness of PRMF, in comparison with strong baseline algorithms.

The future work will focus on the following directions. First, we would like to develop more efficient optimization algorithms for PRMF using the parallel optimization framework [Wang *et al.*, 2013]. Second, we are also interested in extending PRMF to solve top-N item recommendation problems with users' implicit feedback. The potential directions are adopting logistic matrix factorization [Johnson, 2014; Liu *et al.*, 2016] or ranking metrics optimization methods [Zhao *et al.*, 2011] to model users' implicit feedback.

References

[Boyd *et al.*, 2011] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[Cai *et al.*, 2011] Tony Cai, Weidong Liu, and Xi Luo. A constrained ℓ_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106(494):594–607, 2011.

[Candes and Tao, 2007] Emmanuel Candes and Terence Tao. The dantzig selector: statistical estimation when p is much larger than n. *The Annals of Statistics*, pages 2313–2351, 2007.

[Davis *et al.*, 2007] Jason V Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S Dhillon. Information-theoretic metric learning. In *ICML'07*, pages 209–216, 2007.

[Fang *et al.*, 2014] Hui Fang, Yang Bao, and Jie Zhang. Leveraging decomposed trust in probabilistic matrix factorization for effective recommendation. In *AAAI'14*, pages 30–36, 2014.

[Guo *et al.*, 2015] Guibing Guo, Jie Zhang, and Neil Yorke-Smith. Trustsvd: Collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In *AAAI'15*, pages 123–125, 2015.

[Gupta and Nagar, 1999] Arjun K Gupta and Daya K Nagar. *Matrix variate distributions*, volume 104. 1999.

[Hu *et al.*, 2014] Longke Hu, Aixin Sun, and Yong Liu. Your neighbors affect your ratings: on geographical neighborhood influence to rating prediction. In *SIGIR'14*, pages 345–354. ACM, 2014.

[Hu *et al.*, 2015] Guang-Neng Hu, Xin-Yu Dai, Yunya Song, Shu-Jian Huang, and Jia-Jun Chen. A synthetic approach for recommendation: combining ratings, social relations, and reviews. In *IJCAI'15*, pages 1756–1762, 2015.

[Jamali and Ester, 2010] Mohsen Jamali and Martin Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *RecSyS'10*, pages 135–142. ACM, 2010.

[Johnson, 2014] Christopher C Johnson. Logistic matrix factorization for implicit feedback data. *NIPS'14*, 27, 2014.

[Koren *et al.*, 2009] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8):30–37, 2009.

[Li *et al.*, 2015] Hui Li, Dingming Wu, Wenbin Tang, and Nikos Mamoulis. Overlapping community regularization for rating prediction in social recommender systems. In *RecSyS'15*, pages 27–34, 2015.

[Liu and Luo, 2014] Weidong Liu and Xi Luo. High-dimensional sparse precision matrix estimation via sparse column inverse operator. *Journal of Multivariate Analysis*, 2014.

[Liu *et al.*, 2009] Han Liu, John Lafferty, and Larry Wasserman. The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. *Journal of Machine Learning Research*, 10(Oct):2295–2328, 2009.

[Liu *et al.*, 2013] Xin Liu, Yong Liu, Karl Aberer, and Chunyan Miao. Personalized point-of-interest recommendation by mining users' preference transition. In *CIKM'13*, pages 733–738, 2013.

[Liu *et al.*, 2014] Yong Liu, Wei Wei, Aixin Sun, and Chunyan Miao. Exploiting geographical neighborhood characteristics for location recommendation. In *CIKM'14*, pages 739–748, 2014.

[Liu *et al.*, 2015] Yong Liu, Peilin Zhao, Aixin Sun, and Chunyan Miao. A boosting algorithm for item recommendation with implicit feedback. In *IJCAI'15*, volume 15, pages 1792–1798, 2015.

[Liu *et al.*, 2016] Yong Liu, Min Wu, Chunyan Miao, Peilin Zhao, and Xiao-Li Li. Neighborhood regularized logistic matrix factorization for drug-target interaction prediction. *PLoS Computational Biology*, 12(2):e1004760, 2016.

[Ma *et al.*, 2011] Hao Ma, Dengyong Zhou, Chao Liu, Michael R Lyu, and Irwin King. Recommender systems with social regularization. In *WSDM'11*, pages 287–296, 2011.

[Ma, 2013] Hao Ma. An experimental study on implicit social recommendation. In *SIGIR'13*, pages 73–82, 2013.

[Ma, 2014] Hao Ma. On measuring social friend interest similarities in recommender systems. In *SIGIR'14*, pages 465–474, 2014.

[Mei *et al.*, 2014] Jian-Ping Mei, Han Yu, Yong Liu, Zhiqi Shen, and Chunyan Miao. A social trust model considering trustees' influence. In *PRIMA'14*, pages 357–364, 2014.

[Mnih and Salakhutdinov, 2007] Andriy Mnih and Ruslan Salakhutdinov. Probabilistic matrix factorization. In *NIPS'07*, pages 1257–1264, 2007.

[Su and Khoshgoftaar, 2009] Xiaoyuan Su and Taghi M Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, 2009:4, 2009.

[Tang *et al.*, 2013] Jiliang Tang, Xia Hu, Huiji Gao, and Huan Liu. Exploiting local and global social context for recommendation. In *IJCAI'13*, pages 2712–2718, 2013.

[Wang *et al.*, 2013] Huahua Wang, Arindam Banerjee, Cho-Jui Hsieh, Pradeep K Ravikumar, and Inderjit S Dhillon. Large scale distributed sparse precision estimation. In *NIPS'13*, pages 584–592, 2013.

[Yang *et al.*, 2012] Xiwang Yang, Harald Steck, and Yong Liu. Circle-based recommendation in online social networks. In *KDD'12*, pages 1267–1275, 2012.

[Yang *et al.*, 2014] Xiwang Yang, Yang Guo, Yong Liu, and Harald Steck. A survey of collaborative filtering based social recommender systems. *Computer Communications*, 41:1–10, 2014.

[Zhang and Yeung, 2010] Yu Zhang and Dit-Yan Yeung. A convex formulation for learning task relationships in multi-task learning. In *UAI'10*, pages 733–742, 2010.

[Zhang *et al.*, 2013] Xingang Zhang, Qijie Gao, Christopher S.G. Khoo, and Amos Wu. Categories of friends on social networking sites: An exploratory study. In *ALIEP'13*, 2013.

[Zhao *et al.*, 2011] Peilin Zhao, Rong Jin, Tianbao Yang, and Steven C Hoi. Online auc maximization. In *ICML'11*, pages 233–240, 2011.