Cost-Aware Collaborative Filtering for Travel Tour Recommendations

YONG GE, University of North Carolina at Charlotte
HUI XIONG, Rutgers University
ALEXANDER TUZHILIN, New York University
QI LIU, University of Science and Technology of China

Advances in tourism economics have enabled us to collect massive amounts of travel tour data. If properly analyzed, this data could be a source of rich intelligence for providing real-time decision making and for the provision of travel tour recommendations. However, tour recommendation is quite different from traditional recommendations, because the tourist's choice is affected directly by the travel costs, which includes both financial and time costs. To that end, in this article, we provide a focused study of cost-aware tour recommendation. Along this line, we first propose two ways to represent user cost preference. One way is to represent user cost preference by a two-dimensional vector. Another way is to consider the uncertainty about the cost that a user can afford and introduce a Gaussian prior to model user cost preference. With these two ways of representing user cost preference, we develop different cost-aware latent factor models by incorporating the cost information into the probabilistic matrix factorization (PMF) model, the logistic probabilistic matrix factorization (LPMF) model, and the maximum margin matrix factorization (MMMF) model, respectively. When applied to real-world travel tour data, all the cost-aware recommendation models consistently outperform existing latent factor models with a significant margin.

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information filtering*; H.2.8 [**Database Management**]: Database Applications—*Data mining*

General Terms: Algorithms, Experimentation

Additional Key Words and Phrases: Cost-aware collaborative filtering, tour recommendation

ACM Reference Format:
Ge, Y., Xiong, H., Tuzhilin, A., and Liu, Q. 2014. Cost-aware collaborative filtering for travel tour recommendations. *ACM Trans. Inf. Syst.* 32, 1, Article 4 (January 2014), 31 pages.
DOI:http://dx.doi.org/10.1145/2559169

1. INTRODUCTION

Recent years have witnessed an increased interest in data-driven travel marketing. As a result, massive amounts of travel data have been accumulated, thus providing unparalleled opportunities for people to understand user behaviors and generate useful

A preliminary version of this work has been published in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'11)* [Ge et al. 2011].
This research was partially supported by the National Science Foundation (NSF) via grant number CCF-1018151, the National Natural Science Foundation of China (NSFC) via project numbers 70890082, 71028002, and 61203034.
Authors' addresses: Y. Ge (corresponding author), University of North Carolina at Charlotte; email: yong.ge@uncc.edu, ygestrive@gmail.com; H. Xiong, MSIS Department, Rutgers University; email: hxiong@rutgers.edu; A. Tuzhilin, Stern School of Business, New York University; email: atuzhili@stern.nyu.edu; Q. Liu, University of Science and Technology of China; email: feiniaol@mail.ustc.edu.cn.

DOI:http://dx.doi.org/10.1145/2559169

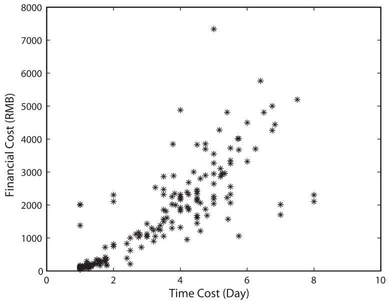


Fig. 1. The cost distribution.

knowledge, which in turn deliver intelligence for real-time decision making in various fields, including travel tour recommendation.

Recommender systems address the information-overload problem by identifying user interests and providing personalized suggestions. In general, there are three ways to develop recommender systems [Adomavicius and Tuzhilin 2005]. The first is content based: it suggests the items which are similar to those a given user has liked in the past. The second is based on collaborative filtering [Ge et al. 2011b; Liu et al. 2010a, 2010c]. In other words, recommendations are made according to the tastes of other users that are similar to the target user. Finally, a third way is to combine these two preceding approaches and lead to a hybrid solution [Burke 2007]. However, the development of recommender systems for travel tour recommendation is significantly different from developing recommender systems in traditional domains, since the tourist's choice is directly affected by the travel cost which includes the financial cost as well as various other types of costs, such as time and opportunity costs.

In addition, there are some unique characteristics of travel tour data which distinguish the travel tour recommendation from traditional recommendations, such as movie recommendations. First, the prices of travel packages can vary a lot. For example, by examining the real-world travel tour logs collected by a travel company, we find that the prices of packages can range from $20 to $10,000. Second, the time cost of packages also varies. For instance, while some travel packages take less than three days, other packages may take more than ten days. In traditional recommender systems, the cost of consuming a recommended item, such as a movie or music, is usually not a concern for customers. However, the tourists usually have financial and time constraints for selecting a travel package. In fact, Figure 1 shows the cost distributions of some tourists. In the figure, each point corresponds to one user. As can be seen, both the financial and time costs vary a lot among different tourists. Therefore, for the traditional recommendation models which do not consider the cost of travel packages, it is difficult to provide the right travel tour recommendation for the right tourists. For example, traditional recommender systems might recommend a travel package to a tourist who cannot afford it because of the price or time commitment.

To address this challenge, in this article, we study how to incorporate the cost information into traditional latent factor models for travel tour recommendation. The extended latent factor models aim to learn user cost preferences and user interests simultaneously from the large scale of travel tour logs. Specifically, we introduce two types of cost information into the traditional latent factor models. The first type of cost information refers to the observable costs of a travel package, which include both financial cost and time cost of the travel package. For example, if a person goes on a trip to Cambodia for seven days and pays $2,000 for the travel package j, then the observed costs of this travel package are denoted as a vector $C_{V_j} = (2000, 7)$. The second type of cost information refers to the unobserved financial and time cost preferences of a user. We propose two different ways to represent the unobserved user's cost preference. First, we represent the cost preference of user i with a two-dimensional cost vector C_{U_i} which denotes both financial and time costs. Second, since there is still some uncertainty about the financial and time costs that a user can afford, we further introduce a Gaussian priori $\mathcal{G}(C_{U_i})$, instead of the cost vector C_{U_i}, on the cost preference of user i to express the uncertainty.

Given this item cost information and two ways of representing user cost preference, we have introduced two cost-aware probabilistic matrix factorization (PMF) [Salakhutdinov and Mnih 2008] models [Ge et al. 2011a]. These two cost-aware probabilistic matrix factorization models are based on the Gaussian noise assumption over observed implicit ratings. However, in this article, we further argue that it may be better to assume noise term as binomial, because over 60% of implicit ratings of travel packages are 1. Therefore, we further investigate two more latent factor models, that is, logistic probabilistic matrix factorization (LPMF) [Yang et al. 2011] and maximum margin matrix factorization (MMMF) [Srebro et al. 2005] models, and propose new cost-aware models based on them in this article. Compared with the probabilistic matrix factorization model studied in Ge et al. [2011a], these two latent factor models are based on different assumptions and have different mathematical formulations. We have to develop different techniques to incorporate the cost information into these two models in this article. Furthermore, for both logistic probabilistic matrix factorization and maximum margin matrix factorization models, we need to sample negative ratings, which were not considered in Ge et al. [2011a], to learn the latent features. In sum, we develop cost-aware extended models by using two ways of representing user cost preference for PMF, LPMF, and MMMF models. In addition to the unknown latent features, such as the user's latent features, the unobserved user's cost information (e.g., C_U or $\mathcal{G}(C_U)$) is also learned by training these extended cost-aware latent factor models. Particularly, by investigating and extending the preceding three latent factor models, we expect to gain more understanding about which model works the best for travel tour recommendations in practice and how much improvement we may achieve by incorporating the cost information into the different models. Finally, we provide efficient algorithms to solve the different objective functions in these extended models.

Finally, with real-world travel data, we provide very extensive experimentation in this article, which is much more than that in Ge et al. [2011a]. Specifically, we first show that the performances of PMF, LPMF, and MMMF models for tour recommendation can be improved by taking the cost information into consideration, especially when active users have very few observed ratings. The statistical significance test shows that the improvement of cost-aware models is significant. Second, the extended MMMF and LPMF models lead to a better improvement of performance than the extended PMF models in terms of Precision@K and MAP for travel tour recommendations. Third, we demonstrate that the sampled negative ratings have interesting influence on the performance of extended LPMF and MMMF models for travel package recommendations. Finally, we demonstrate that the latent user cost information learned by extended models can help travel companies with customer segmentation.

The remainder of this article is organized as follows. Section 2 briefly describes the related work. In Section 3, we show two ways of representing user cost preference and propose both vPMF and gPMF models with the incorporated cost information. Section 4 shows the extended models of the LPMF model. In Section 5, we provide the extended models of the MMMF model. Section 6 presents the experimental results on the real-world travel tour data. Finally, in Section 7, we draw conclusions.

2. RELATED WORK

Related work can be grouped into three categories. The first includes the work on collaborative filtering models. In the second, we introduce the related work about travel recommendation. Finally, the third category includes the work on cost/profit-based recommendation.

2.1. Collaborative Filtering

Two types of collaborative filtering models have been intensively studied recently: memory-based and model-based approaches. Memory-based algorithms [Bell and Koren 2007; Deshpande and Karypis 2004; Koren 2008] essentially make rating predictions by using some other neighboring ratings. In the model-based approaches, training data are used to train a predefined model. Different approaches [Ge et al. 2011a; Hofmann 2004; Liu et al. 2010d; Marlin 2003; Xue et al. 2005] vary due to different statistical models assumed for the data. In particular, various matrix factorization [Agarwal and Chen 2009; Salakhutdinov and Mnih 2008; Srebro et al. 2005] methods have been proposed for collaborative filtering. Most MF approaches focus on fitting the user-item rating matrix using low rank approximation and use the learned latent user/item features to predict the unknown ratings. The PMF model [Salakhutdinov and Mnih 2008] was proposed by assuming Gaussian noise to observed ratings and applying Gaussian prior to latent features. Via introducing logistic function to the loss function, PMF was also extended to address binary ratings [Yang et al. 2011]. Recently, instead of constraining the dimensionality of latent factors, Srebro et al. [2005] proposed the MMMF model via constraining the norms of user and item feature matrices. Finally, more sophisticated methods are also available to consider user/item side information [Adams et al. 2010; Gu et al. 2010], social influence [Ma et al. 2009], and context information [Adomavicius et al. 2005] (e.g., temporal information [Xiong et al. 2010] and spatiotemporal context [Lu et al. 2009]). However, most of these methods were developed for recommending traditional items, such as movie, music, articles, and webpages. In these recommendation tasks, financial and time costs are usually not essential to the recommendation results and are not considered in the models.

2.2. Travel Recommendation

Travel-related recommendations have been studied before. For instance, in Hao et al. [2010], one probabilistic topic model was proposed to mine two types of topics, that is, local topics (e.g., lava, coastline) and global topics (e.g., hotel, airport), from travelogue on the website. Travel recommendation was performed by recommending a destination, which is similar to a given location or relevant to a given travel intention, to a user. Cena et al. [2006] presented UbiquiTO tourist guide for intelligent content adaptation. UbiquiTO used a rule-based approach to adapt the content of the provided recommendation. A content adaptation approach [Yu et al. 2006] was developed for presenting tourist-related information. Both content and presentation recommendations were tailored to particular mobile devices and network capabilities. They used content-based, rule-based, and Bayesian classification methods to provide tourism-related mobile recommendations. Baltrunas et al. [2011a] presented a method to recommend various places of interest for tourists by using physical, social, and modal types of contextual information. The recommendation algorithm was based on the factor model that is extended to model the impact of the selected contextual conditions on the predicted rating. A tourist guide system COMPASS [Setten et al. 2004] was presented to support many standard tourism-related functions. Finally, other examples of travel recommendations proposed in the literature are also available [Ardissono et al. 2002; Baltrunas et al. 2011b; Carolis et al. 2009; Cheverst et al. 2000; Jannach and

Hegelich 2009; Park et al. 2007; Woerndl et al. 2011], and Kenteris et al. [2011] provided an extensive categorization of mobile guides according to connectivity to Internet, being indoor versus outdoor, etc. In this article, we focus on developing cost-aware latent factor models for travel package recommendation, which is different from the preceding travel recommendation tasks.

2.3. Cost/Profit-Based Recommendation

Also, there are some prior works [Chen et al. 2008; Das et al. 2010; Ge et al. 2010; Hosanagar et al. 2008] related to profit/cost-based recommender systems. For instance, Hosanagar et al. [2008] studied the impact of a firm's profit incentives on the design of recommender systems. In particular, this research identified the conditions under which a profit-maximizing recommender recommends the item with the highest margins and those under which it recommends the most relevant item. It also explored the mismatch between consumers and firm incentives and determined the social costs associated with this mismatch. Das et al. [2010] studied the question of how a vendor can directly incorporate profitability of items into the recommendation process so as to maximize the expected profit while still providing accurate recommendations. The proposed approach takes the output of a traditional recommender system and adjusts it according to item profitability. However, most of these prior travel-related and cost-based recommendation studies did not explicitly consider the expense and time cost for travel recommendation. Also, in this article, we focus on travel tour recommendation.

Finally, in our preliminary work on travel tour recommendation [Ge et al. 2011a], we developed two simple cost-aware PMF models for travel tour recommendation. In this article, we provide a comprehensive study of cost-aware collaborative filtering for travel tour recommendation. Particularly, we investigate how to incorporate the cost information into different latent factor models and evaluate the design decisions related to model choice and development.

3. COST-AWARE PMF MODELS

In this section, we propose two ways to represent user cost preferences and introduce how to incorporate the cost information into the PMF [Salakhutdinov and Mnih 2008] model by designing two cost-aware PMF models: the vPMF model and the gPMF model.

3.1. The vPMF Model

vPMF is a cost-aware probabilistic matrix factorization model which represents user/item costs with two-dimensional vectors, as shown in Figure 2(b). Suppose we have N users and M packages. Let R_{ij} be the rating of user i for package j, and U_i and V_j represent D-dimensional user-specific and package-specific latent feature vectors, respectively, (both U_i and V_j are column vectors in this article). Also, let C_{U_i} and C_{V_j} represent two-dimensional cost vectors for user i and package j, respectively. In addition, C_U and C_V simply denote the sets of cost vectors for all the users and all the packages, respectively. The conditional distribution over the observed ratings $R \in \mathcal{R}^{N \times M}$ is

$$p(R|U, V, C_U, C_V, \sigma^2) = \prod_{i=1}^{N}\prod_{j=1}^{M}[\mathcal{N}(R_{ij}|f(U_i, V_j, C_{U_i}, C_{V_j}), \sigma^2)]^{I_{ij}}, \tag{1}$$

where $\mathcal{N}(x|\mu, \sigma^2)$ is the probability density function of the Gaussian distribution with mean μ and variance σ^2, and I_{ij} is the indicator variable that is equal to 1 if user i rates item j and is equal to 0 otherwise. Also, U is a $D \times N$ matrix and V is a $D \times M$

matrix. The function $f(x)$ is to approximate the rating for item j by user i. We define $f(x)$ as

$$f(U_i, V_j, C_{U_i}, C_{V_j}) = S(C_{U_i}, C_{V_j}) \cdot U_i^T V_j \ , \tag{2}$$

where $S(C_{U_i}, C_{V_j})$ is a similarity function for measuring the similarity between user cost vector C_{U_i} and item cost vector C_{V_j}. Several existing similarity/distance functions can be used here to perform this calculation, such as Pearson coefficient, the cosine similarity, or Euclidean distance. C_V can be considered known in this article, because we can directly obtain the cost information for tour packages from the tour logs. C_U is the set of user cost vectors which is going to be estimated. Moreover, we also apply zero-mean spherical Gaussian prior [Salakhutdinov and Mnih 2008] on user and item latent feature vectors.

$$p(U|\sigma_U^2) = \prod_{i=1}^{N} \mathcal{N}(U_i|0, \sigma_U^2 \mathbf{I}),$$
$$p(V|\sigma_V^2) = \prod_{j=1}^{M} \mathcal{N}(V_j|0, \sigma_V^2 \mathbf{I}).$$

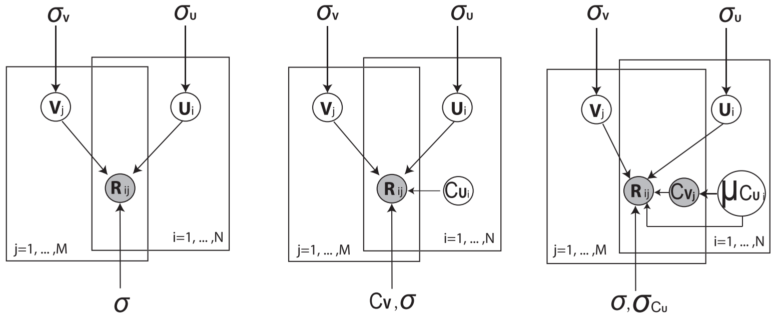
As shown in Figure 2, in addition to user and item latent feature vectors, we also need to learn user cost vectors simultaneously. By a Bayesian inference, we have

$$\begin{aligned} & p(U,V, C_U|R, C_V, \sigma^2, \sigma_U^2, \sigma_V^2) \\ & \quad \propto p(R|U, V, C_U, C_V, \sigma^2) p(U|\sigma_U^2) p(V|\sigma_V^2) \\ & \quad = \prod_{i=1}^{N} \prod_{j=1}^{M} [\mathcal{N}(R_{ij}|f(U_i, V_j, C_{U_i}, C_{V_j}), \sigma^2)]^{I_{ij}} \\ & \quad \times \prod_{i=1}^{M} \mathcal{N}(U_i|0, \sigma_U^2 \mathbf{I}) \times \prod_{j=1}^{N} \mathcal{N}(V_j|0, \sigma_V^2 \mathbf{I}). \end{aligned} \tag{3}$$

U, V, and C_U can be learned by maximizing this posterior distribution or the log of the posterior distribution over user cost vectors and user and item latent feature vectors with fixed hyperparameters, that is, the observation noise variance and prior variances. By Equation (3) or Figure 2, we can find that vPMF is actually an enhanced general model of PMF by taking the cost information into consideration. In other words, if we limit $S(C_{U_i}, C_{V_j})$ to 1 for all pairs of user and item, vPMF will be a PMF model.

The log of the posterior distribution in Equation (3) is

$$\begin{aligned} & \ln\, p(U, V, C_U|R, C_V, \sigma^2, \sigma_U^2, \sigma_V^2) = \\ & \quad - \frac{1}{2\sigma^2} \sum_{i=1}^{N} \sum_{j=1}^{M} I_{ij}(R_{ij} - f(U_i, V_j, C_{U_i}, C_{V_j}))^2 \\ & \quad - \frac{1}{2}\{(\sum_{i=1}^{N} \sum_{j=1}^{M} I_{ij}) \ln \sigma^2 + ND \ln \sigma_U^2 + MD \ln \sigma_V^2\} \\ & \quad - \frac{1}{2\sigma_U^2} \sum_{i=1}^{N} U_i^T U_i - \frac{1}{2\sigma_V^2} \sum_{j=1}^{M} V_j^T V_j + C, \end{aligned} \tag{4}$$



(a) PMF (b) vPMF (c) gPMF

Fig. 2. Graphical models.

where C is a constant that does not depend on the parameters. Maximizing the log of the posterior distribution over user cost vectors and user and item latent feature vectors is equivalent to minimizing the sum-of-squared-errors objective function with quadratic regularization terms:

$$
\begin{aligned}
E &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{M} I_{ij}(R_{ij} - S(C_{U_i}, C_{V_j}) \cdot U_i^T V_j)^2 \\
&\quad + \frac{\lambda_U}{2}\sum_{i=1}^{N} ||U_i||_F^2 + \frac{\lambda_V}{2}\sum_{j=1}^{M} ||V_j||_F^2,
\end{aligned} \tag{5}
$$

where $\lambda_U = \sigma^2/\sigma_U^2$, $\lambda_V = \sigma^2/\sigma_V^2$, and $|| \cdot ||_F^2$ denotes the Frobenius norm. From the objective function, that is, Equation (5), we can also see that the vPMF model will be reduced to the PMF model if $S(C_{U_i}, C_{V_j}) = 1$ for all pairs of user and item.

Since the dimension of cost vectors is small, we use the Euclidean distance for the similarity function as $S(C_{U_i}, C_{V_j}) = (2 - ||C_{U_i} - C_{V_j}||^2)/2$. Note that with this similarity function, we assume that a user's cost preference is around a center. A user tends to not choose travel packages which are either too expensive or too cheap for him/her. For instance, if a user always consumes travel packages which cost around \$1,000, travel packages which cost either too much (e.g., \$5,000) or too less (e.g., \$100) will be equally unattractive to this user. Actually, from the real-world data, we do observe that this assumption generally holds. Specifically speaking, we show the financial cost of travel packages which are consumed by ten different users in our training data in Figure 3. As can be seen, a user tends to consume travel packages with financial cost surrounding a center. Therefore, we use such a symmetric similarity function in this article. Furthermore, since two attributes of the cost vector have significantly different levels of scale, we utilize the min-max normalization technique to preprocess all cost vectors of items. Then the value of attribute of the cost vectors is scaled to fit in the specific range $[0, 1]$. Subsequently, the value of the preceding similarity function also locates in the range $[0, 1]$. Then, a local minimum of the objective function given by

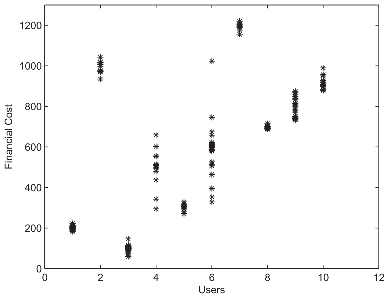


Fig. 3. An illustration of financial cost.

Equation (5) can be obtained by performing gradient descent in U_i, V_j, and C_{U_i} as

$$\frac{\partial E}{\partial U_i} = \sum_{j=1}^{M} I_{ij}(S(C_{U_i}, C_{V_j}) \cdot U_i^T V_j - R_{ij}) \cdot S(C_{U_i}, C_{V_j})V_j + \lambda_U U_i,$$

$$\frac{\partial E}{\partial V_j} = \sum_{i=1}^{N} I_{ij}(S(C_{U_i}, C_{V_j}) \cdot U_i^T V_j - R_{ij}) \cdot S(C_{U_i}, C_{V_j})U_i^T + \lambda_V V_j,$$

$$\frac{\partial E}{\partial C_{U_i}} = \sum_{j=1}^{M} I_{ij}(S(C_{U_i}, C_{V_j})U_i^T V_j - R_{ij}) \cdot U_i^T V_j S'(C_{U_i}, C_{V_j}), \tag{6}$$

where $S'(C_{U_i}, C_{V_j})$ is the derivative with respect to C_{U_i}.

3.2. The gPMF Model

In the real world, the user's expectation on the financial and time cost of travel packages may vary within a certain range. Also, as shown in Equation (5), overfitting can happen when we perform the optimization with respect to C_{U_i} ($i = 1 \cdots N$). These two observations suggest that it might be better if we could use a distribution to model the user cost preference instead of representing it as a two-dimension vector. Therefore, we propose using a two-dimensional Gaussian distribution to model user cost preference in the gPMF model as

$$p(C_{U_i}|\mu_{C_{U_i}}, \sigma^2_{C_U}) = \mathcal{N}(C_{U_i}|\mu_{C_{U_i}}, \sigma^2_{C_U}\mathbf{I}). \tag{7}$$

In Equation (7), $\mu_{C_{U_i}}$ is the mean of the two-dimensional Gaussian distribution for user U_i. $\sigma^2_{C_U}$ is assumed to be the same for all the users for simplicity.

In the gPMF model, since we use a two-dimensional Gaussian distribution to represent user cost preference, we need to change the function for measuring the similarity/match between user cost preference and package cost information. Considering that each package cost is represented by a constant vector and that user cost preference is characterized via a distribution, we measure the similarity between user cost preference and package cost as

$$S_G(C_{V_j}, \mathcal{G}(C_{U_i})) = \mathcal{N}(C_{V_j}|\mu_{C_{U_i}}, \sigma^2_{C_U}\mathbf{I}), \tag{8}$$

where we simply use $\mathcal{G}(C_{U_i})$ to represent the two-dimensional Gaussian distribution of user U_i. Note that C_{U_i} in Equations (8) and (7) represents the variable of the user cost distribution $\mathcal{G}(C_{U_i})$, instead of a user cost vector. Also note that, similar to the similarity function in vPMF model, we adopt the symmetric Equation (8) based on the same assumption and observation (as shown in Figure 3). In other words, we equally penalize both higher cost and lower cost than user cost preference. Along this line, the function for approximating the rating for item j by user i is defined as

$$\begin{aligned} f_G(U_i, V_j, \mathcal{G}(C_{U_i}), C_{V_j}) &= S_G(C_{V_j}, \mathcal{G}(C_{U_i})) \cdot U_i^T V_j \\ &= \mathcal{N}(C_{V_j} | \mu_{C_{U_i}}, \sigma_{C_U}^2 \mathbf{I}) \cdot U_i^T V_j \,. \end{aligned} \tag{9}$$

With this representation of user cost preference and the similarity function, a similar Bayesian inference as Equation (3) can be obtained:

$$\begin{aligned} & p(U, V, \mu_{C_U} | R, C_V, \sigma^2, \sigma_U^2, \sigma_V^2, \sigma_{C_U}^2) \\ & \propto p(R|U, V, \mu_{C_U}, C_V, \sigma^2, \sigma_{C_U}^2) p(C_V | \mu_{C_U}, \sigma_{C_U}^2) p(U|\sigma_U^2) p(V|\sigma_V^2) \\ & = \prod_{i=1}^{N} \prod_{j=1}^{M} \left(\mathcal{N}\left(R_{ij} | f_G \left(U_i, V_j, \mathcal{G}(C_{U_i}), C_{V_j} \right), \sigma^2 \right) \right)^{I_{ij}} \\ & \times \prod_{i=1}^{N} \prod_{j=1}^{M} \mathcal{N}(C_{V_j} | \mu_{C_{U_i}}, \sigma_{C_U}^2 \mathbf{I})^{I_{ij}} \\ & \times \prod_{i=1}^{N} \mathcal{N}(U_i | 0, \sigma_U^2 \mathbf{I}) \times \prod_{j=1}^{M} \mathcal{N}(V_j | 0, \sigma_V^2 \mathbf{I}), \end{aligned} \tag{10}$$

where $\mu_{C_U} = (\mu_{C_{U_1}}, \mu_{C_{U_2}}, \cdots, \mu_{C_{U_N}})$, which denotes the set of means of all user cost distributions. $p(C_V | \mu_{C_U}, \sigma_{C_U}^2)$ is the likelihood given the parameters of all user cost distributions. Given the known ratings of a user, the cost of packages rated by this user can be treated as observations of this user's cost distribution. This is why we represent the likelihood over C_V, that is, the set of package cost. Then we are able to derive the likelihood as $\prod_{i=1}^{N} \prod_{j=1}^{M} \mathcal{N}(C_{V_j} | \mu_{C_{U_i}}, \sigma_{C_U}^2 \mathbf{I})^{I_{ij}}$ in Equation (10).

Maximizing the log of the posterior over the means of all user cost distributions and user and item latent features is equivalent to minimizing the sum-of-squared-errors objective function with quadratic regularization terms with respect to U, V, and $\mu_{C_U} = (\mu_{C_{U_1}}, \mu_{C_{U_2}}, \cdots, \mu_{C_{U_N}})$:

$$\begin{aligned} E &= \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} I_{ij} \left(R_{ij} - \mathcal{N}(C_{V_j} | \mu_{C_{U_i}}, \sigma_{C_U}^2 \mathbf{I}) \cdot U_i^T V_j \right)^2 \\ &+ \frac{\lambda_U}{2} \sum_{i=1}^{N} ||U_i||_F^2 + \frac{\lambda_V}{2} \sum_{j=1}^{M} ||V_j||_F^2 \\ &+ \frac{\lambda_{C_U}}{2} \sum_{i=1}^{N} \sum_{j=1}^{M} I_{ij} ||C_{V_j} - \mu_{C_{U_i}}||^2, \end{aligned} \tag{11}$$

where $\lambda_{C_U} = \sigma^2/\sigma^2_{C_U}$, $\lambda_U = \sigma^2/\sigma^2_U$, and $\lambda_V = \sigma^2/\sigma^2_V$. As we can see from Equation (11), the two-dimensional Gaussian distribution for modeling user cost preference leads to one more regularization term to the objective function, thus easing overfitting. The gPMF model is also an enhanced general model of PMF, because the objective function, that is, Equation (11), is reduced to that of PMF if $\sigma^2_{C_U}$ is limited to be infinite. A local minimum of the objective function given by Equation (11) can be identified by performing gradient descent in U_i, V_j, and $\mu_{C_{U_i}}$. For the same reason, we also utilize the min-max normalization to preprocess all cost vectors of items before training the model.

In this article, instead of using Equation (2) and Equation (9), which may have predictions out of the valid rating range, we further apply the logistic function $g(x) = 1/(1 + exp(-x))$ to the results of Equation (2) and Equation (9). The applied logistic function bounds the range of predictions as $[0, 1]$. Also, we map the observed ratings from the original range $[1, K]$ (K is the maximum rating value) to the interval $[0, 1]$ using the function $t(x) = (x-1)/(K-1)$, thus the valid rating range matches the range of predictions by our models. Eventually, to get the final prediction for an unknown rating, we restore the scale of predictions from $[0, 1]$ to [1,K] by using the inverse transformation of function $t(x) = (x - 1)/(K - 1)$.

3.3. The Computational Complexity

The main computation of gradient methods is to evaluate the object function and its gradients against variables. Because of the sparseness of matrices R, the computational complexity of evaluating the object function of Eq. (5) is $\mathcal{O}(\eta f)$, where η is the number of nonzero entries in R and f is the number of latent factors. The computational complexity for gradients $\frac{\partial E}{\partial U}$, $\frac{\partial E}{\partial V}$, and $\frac{\partial E}{\partial C_U}$ in Equation (6) is also $\mathcal{O}(\eta f)$. Thus, for each iteration, the total computational complexity is $\mathcal{O}(\eta f)$. Thus, the computational cost of vPMF model is linear with respect to the number of observed ratings in the sparse matrix R. Similarly, the overall computational complexity of gPMF model is also $\mathcal{O}(\eta f)$, because the only difference between gPMF and vPMF is that we need to compute the cost similarity with the two-dimensional Gaussian distribution, instead of the Euclidean distance involved in vPMF. This complexity analysis shows that the proposed cost-aware models are efficient and can scale to very large data. In addition, instead of performing batch learning, we divide the training set into subbatches and update all latent features after subbatching in order to speed up training.

4. COST-AWARE LPMF MODELS

In this section, we first briefly introduce the LPMF model and then propose the cost-aware LPMF models to incorporate the cost information. Note that in this Section and in Section 5, all notations, such as C_{U_i} and $\mu_{C_{U_i}}$, have the same meaning as in Section 3 unless specified otherwise.

4.1. The LPMF Model

LPMF [Yang et al. 2011] generalizes the PMF model via applying the logistic function as the loss function. Given binary ratings, R_{ij} follows a Bernoulli distribution instead of a normal distribution. Then, the logistic function is used to model the rating as

$$P(R_{ij}=1|U_i,V_j)=\sigma(U_i^TV_j)=\frac{1}{1+e^{-U_i^TV_j}},$$

$$P(R_{ij}=0|U_i,V_j)=1-P(R_{ij}=1|U_i,V_j)=\frac{1}{1+e^{U_i^TV_j}}=\sigma(-U_i^TV_j)\,,$$

where $R_{ij}=1$ means R_{ij} is a positive rating and $R_{ij}=0$ indicates R_{ij} is a negative rating. Given the training set, that is, all observed binary ratings, the conditional likelihood over all available ratings can be calculated as

$$p(R|U,V)=\prod_{i=1}^{N}\prod_{j=1}^{M}\Big((P(R_{ij}=1))^{R_{ij}}(1-P(R_{ij}=1))^{1-R_{ij}}\Big)^{I_{ij}}, \tag{12}$$

where $(P(R_{ij}=1))^{R_{ij}}(1-P(R_{ij}=1))^{1-R_{ij}}$ is actually the Bernoulli probability mass function. Also, I_{ij} is the indicator variable that is equal to 1 if user i rates item j as either positive or negative and is equal to 0 otherwise.

To avoid overfitting via the maximum likelihood estimation (MLE), we also introduce Gaussian priors onto U and V and find a maximum a posteriori (MAP) estimation for U and V. The log of the posterior distribution over U and V is given by

$$\begin{aligned}\ln\, & p(U,V|R,\sigma_U^2,\sigma_V^2)\\ &=\sum_{i=1}^{N}\sum_{j=1}^{M}I_{ij}\Big(R_{ij}\ln\sigma(U_i^TV_j)+(1-R_{ij})\ln\sigma(-U_i^TV_j)\Big)\\ &-\frac{1}{2\sigma_U^2}\sum_{i=1}^{N}U_i^TU_i-\frac{1}{2\sigma_V^2}\sum_{j=1}^{M}U_j^TV_j\\ &-\frac{1}{2}(ND\ln\sigma_U^2+MD\ln\sigma_V^2)+C,\end{aligned} \tag{13}$$

where C is a constant that does not depend on the parameters. By maximizing the objective function, that is, Equation (13), U and V can be estimated.

However, in our travel tour dataset, the original ratings are not binary but ordinal. Thus, we need to binarize the original ordinal ratings before training the LPMF model. In fact, some research [Pan et al. 2008; Yang et al. 2011] has shown that the binarization can yield better recommendation performances in terms of relevance and accuracy [Herlocker et al. 2004]. We are interested in investigating this potential for our travel recommendations. Specifically, a rating R_{ij} is considered as positive if it is equal to or greater than 1. However, in our travel tour dataset, there are no negative ratings available. Actually, in many recommendation applications, such as `YouTube.com` and `Epinions.com`, negative ratings may be extremely few or completely missed because users are much less inclined to give negative ratings for items they dislike than positive ratings for items they like, as illustrated by Marlin and Zemel [2007, 2009]. To this end, we adopt the User-Oriented Sampling approach in Pan et al. [2008; Pan and Scholz 2009] to get the negative ratings. Basically, if a user has rated more items (i.e., travel packages) with positive ratings, those items that she/he has not rated positively could be rated as negative with higher probability. Overall, we control the number of sampled negative ratings by setting the ratio of the number of negative ratings to the number of positive ratings, that is, α. For example, $\alpha = 0.1$ means that the number of negative ratings we sample is 10% of the number of positive ratings.

4.2. The vLPMF Model

Similar to the vPMF model, we first represent user cost preference with a two-dimensional vector. Then we incorporate the cost information into the LPMF model as

$$P(R_{ij} = 1|U_i, V_j) = S(C_{U_i}, C_{V_j}) \cdot \sigma(U_i^T V_j) = \frac{S(C_{U_i}, C_{V_j})}{1 + e^{-U_i^T V_j}}, \tag{14}$$

$$P(R_{ij} = 0|U_i, V_j) = 1 - P(R_{ij} = 1|U_i, V_j) = \frac{1 + e^{U_i^T V_j} - S(C_{U_i}, C_{V_j})}{1 + e^{U_i^T V_j}}. \tag{15}$$

Here, the similarity $S(C_{U_i}, C_{V_j})$ needs to be set within the range $[0, 1]$ in order to maintain that the conditional probability is within the range $[0, 1]$. Thus, the similarity function defined in Section 3.1, that is, $S(C_{U_i}, C_{V_j}) = (2 - ||C_{U_i} - C_{V_j}||^2)/2$, is also applicable here.

Given the preceding formulation, we can get the log of posterior distribution over U, V, and C_U as

$$\begin{aligned}
&\ln\, p(U, V|R, \sigma_U^2, \sigma_V^2, C_V, \sigma^2) \\
&= \sum_{i=1}^{N}\sum_{j=1}^{M} I_{ij}\{R_{ij}\ln(S(C_{U_i}, C_{V_j})\sigma(U_i^T V_j)) \\
&+ (1 - R_{ij})\ln(1 - S(C_{U_i}, C_{V_j})\sigma(U_i^T V_j))\} \\
&- \frac{1}{2\sigma_U^2}\sum_{i=1}^{N} U_i^T U_i - \frac{1}{2\sigma_V^2}\sum_{j=1}^{M} V_j^T V_j \\
&- \frac{1}{2}(ND\ln\sigma_U^2 + MD\ln\sigma_V^2) + C.
\end{aligned} \tag{16}$$

We search the local maximum of the objective function, that is, Equation (16), by performing gradient ascent in U_i $(1 \le i \le N)$, V_j $(1 \le j \le M)$ and C_{U_i} $(1 \le i \le N)$. To save space, we omit the details of partial derivatives.

4.3. The gLPMF Model

With the two-dimensional Gaussian distribution for modeling user cost preference, that is, Equation (8), we update Equations (14) and (15) as

$$P(R_{ij} = 1|U_i, V_j) = S_G(C_{V_j}, \mathcal{G}(C_{U_i})) \cdot \sigma(U_i^T V_j) = \frac{S_G(C_{V_j}, \mathcal{G}(C_{U_i}))}{1 + e^{-U_i^T V_j}},$$

$$P(R_{ij} = 0|U_i, V_j) = 1 - P(R_{ij} = 1|U_i, V_j) = \frac{1 + e^{U_i^T V_j} - S_G(C_{V_j}, \mathcal{G}(C_{U_i}))}{1 + e^{U_i^T V_j}},$$

where $S_G(C_{V_j}, \mathcal{G}(C_{U_i}))$ is defined in Equation (8). Here we also constrain the similarity $S_G(C_{V_j}, \mathcal{G}(C_{U_i}))$ to be within the range $[0, 1]$. To apply such a constraint, we limit the common variance, that is, $\sigma_{C_U}^2$ in Equation (8), to a specific range, which will be discussed in Section 6.

Then the log of the posterior distribution over U, V, and μ_{C_U} can be updated as

$$\begin{aligned}
&\ln p(U, V, \mu_{C_U} | R, \sigma_U^2, \sigma_V^2, \sigma_{C_U}^2, \sigma^2, C_V) \\
&= \sum_{i=1}^{N}\sum_{j=1}^{M} I_{ij}[R_{ij} \ln S_G(C_{V_j}, \mathcal{G}(C_{U_i}))\sigma(U_i^T V_j) \\
&+ (1 - R_{ij}) \ln(1 - S_G(C_{V_j}, \mathcal{G}(C_{U_i}))\sigma(U_i^T V_j))] \\
&- \frac{1}{2\sigma_{C_U}^2} \sum_{i=1}^{N}\sum_{j=1}^{M} I_{ij}(C_{V_j} - \mu_{C_{U_i}})^T (C_{V_j} - \mu_{C_{U_i}}) \\
&- \frac{1}{2\sigma_U^2} \sum_{i=1}^{N} U_i^T U_i - \frac{1}{2\sigma_V^2} \sum_{j=1}^{M} V_j^T V_j - \frac{1}{2}[(\sum_{i=1}^{N}\sum_{j=1}^{M} I_{ij}) \ln \sigma^2 \\
&+ (\sum_{i=1}^{N}\sum_{j=1}^{M} I_{ij}) \ln \sigma_{C_U}^2 + ND \ln \sigma_U^2 + MD \ln \sigma_V^2] + C\,. \qquad (17)
\end{aligned}$$

Finally we search the local maximum of the objective function, that is, Equation (17), by performing gradient ascent in U_i $(1 \le i \le N)$, V_j $(1 \le j \le M)$, and $\mu_{C_{U_i}}$ $(1 \le i \le N)$.

To predict an unknown rating, for example, R_{ij}, as positive or negative with an LPMF, vLPMF, or gLPMF model, we compute the conditional probability $P(R_{ij} = 1)$ with the learned U_i, V_j, C_{U_i}, or $\mu_{C_{U_i}}$. If $P(R_{ij} = 1)$ is greater than 0.5, we predict R_{ij} as positive; otherwise, we predict R_{ij} as negative. In practice, we can also rank all items based on the probability of being positive for a user and recommend the top items to the user.

The computational complexity of LPMF, vPMF, or gPMF is also linear with the number of available ratings for training. We also divide the training set into subbatches and update all latent features subbatch by subbatch.

5. COST-AWARE MMMF MODELS

In this section, we propose the cost-aware MMMF models after briefly introducing the classic MMMF model. For the MMMF model and its cost-aware extensions, we also take binary ratings as input.

5.1. The MMMF Model

MMMF [Rennie and Srebro 2005; Srebro et al. 2005] allows an unbounded dimensionality for the latent feature space via limiting the trace norm of $X = U^T V$. Specifically, given a matrix R with binary ratings, we minimize the trace norm[1] matrix X and the hinge loss as

$$||X||_{\sum} + C \sum_{ij} I_{ij} h(X_{ij} R_{ij}), \qquad (18)$$

[1] Also known as the nuclear norm and the Ky-Fan n-norm.

where C is a trade-off parameter and $h(\cdot)$ is the smooth hinge loss function [Rennie and Srebro 2005] as

$$h(z) = \begin{cases} \frac{1}{2} - z, & \text{if } x \leq 0, \\ \frac{1}{2}(1-z)^2, & \text{if } 0 < x < 1, \\ 0, & \text{if } x \geq 1 \,. \end{cases}$$

Note that for the MMMF model, we denote the positive rating as 1, and the negative rating as -1, instead of 0. By minimizing the objective function, that is, Equation (18), we can estimate U and V. In addition, we adopt the same methods as described in Section 4.1 to binarize the original ordinal ratings and obtain negative ratings.

5.2. The vMMMF Model

To incorporate both user and item cost information into MMMF model, we extend the smooth hinge loss function with the two-dimensional user cost vector as

$$h(X_{ij}, C_{U_i}, C_{V_j}, R_{ij}) = h\left(S(C_{U_i}, C_{V_j})X_{ij}R_{ij}\right). \tag{19}$$

Then we can update the objective function, that is, Equation (18), as

$$||X||_{\sum} + C\sum_{ij} I_{ij} h\left(S(C_{U_i}, C_{V_j})X_{ij}R_{ij}\right). \tag{20}$$

Here we can have different similarity measurements for $S(C_{U_i}, C_{V_j})$, but we need to constrain the similarity $S(C_{U_i}, C_{V_j})$ to be nonnegative; otherwise, the symbol of $X_{ij}R_{ij}$ may be changed by $S(C_{U_i}, C_{V_j})$. To this end, we still use the similarity function defined in Section 3.1 to compute the similarity.

To solve the minimization problem in Equation (20), we adopt the local search heuristic as suggested in Rennie and Srebro [2005], where it was shown that the minimization problem in Equation (20) is equivalent to

$$\begin{aligned} G = \; & \frac{1}{2}(||U||_F^2 + ||V||_F^2) \; + \\ & C\sum_{ij} I_{ij} h\left(S(C_{U_i}, C_{V_j})(U_i^T V_j)R_{ij}\right). \end{aligned} \tag{21}$$

In other words, instead of searching over X, we search over pairs of matrices (U, V), as well as the set of user cost vectors $C_U = \{C_{U_1}, \cdots, C_{U_N}\}$ to minimize the objective function, that is, Equation (21). Finally, we turn to the gradient descent algorithm to solve the optimization problem in Equation (21), as used in Rennie and Srebro [2005].

5.3. The gMMMF Model

Moreover, we extend the smooth hinge loss function with the two-dimensional Gaussian distribution, that is, Equation (8), as

$$h(X_{ij}, \mathcal{G}(C_{U_i}), C_{V_j}, R_{ij}) = h\left(\mathcal{N}(C_{V_j} | \mu_{C_{U_i}}, \sigma_{C_U}^2 \mathbf{I}) X_{ij}R_{ij}\right). \tag{22}$$

Here, $\mathcal{N}(C_{V_j}|\mu_{C_{U_i}}, \sigma^2_{C_U}\mathbf{I})$ is positive naturally because it is a probability density function. Then, similar to Equation (21), we can derive a new objective function:

$$\begin{aligned} G = & \frac{1}{2}(||U||_F^2 + ||V||_F^2) + \\ & C\sum_{ij} I_{ij} h\left(\mathcal{N}(C_{V_j}|\mu_{C_{U_i}}, \sigma^2_{C_U}\mathbf{I})(U_i^T V_j) R_{ij}\right). \end{aligned} \tag{23}$$

To solve this problem, we also adopt the gradient descent algorithm as used for the vMMMF model.

To predict an unknown rating, such as R_{ij}, with MMMF, we compute $U_i^T V_j$. If $U_i^T V_j$ is greater than a threshold, R_{ij} is predicted as positive; otherwise, R_{ij} is predicted as negative. With vMMMF and gMMMF, we predict an unknown rating as positive or negative by thresholding $S(C_{U_i}, C_{V_j})U_i^T V_j$ or $\mathcal{N}(C_{V_j}|\mu_{C_{U_i}}, \sigma^2_{C_U}\mathbf{I})U_i^T V_j$ in the same way. Of course, there are other methods [Rennie and Srebro 2005; Srebro et al. 2005] for deciding the final predictions, but we adopt the preceding simple way, because this is not the focus of this article.

The computational complexity of MMMF, vMMMF, or gMMMF is also linear with the number of available ratings for training. Here, we adopt the same strategy to speed up the training processing.

6. EXPERIMENTAL RESULTS

In this section, we evaluate the performance of the cost-aware collaborative filtering methods on real-world travel data for travel tour recommendation.

6.1. The Experimental Setup

Experimental Data. The travel tour dataset used in this article is provided by a travel company. In the dataset, there are more than 200,000 expense records starting from the beginning of 2000 to October 2010. In addition to the Customer ID and Travel Package ID, there are many other attributes for each record, such as the cost of the package, the travel days, the package name and some short descriptions of the package, and the start date. Also, the dataset includes some information about the customers, such as age and gender. From these records, we are able to obtain the information about users (tourists), items (packages), and user ratings. Moreover, we are able to know the financial and time cost for each package from these tour logs. Instead of using explicit ratings (e.g., scores from 1 to 5), which is actually not available in our travel tour data, we use the purchasing frequency as the implicit rating. Actually, the purchasing frequency has been widely used for measuring the utility of an item for a user [Panniello et al. 2009] in the transaction-based recommender systems [Huang et al. 2004, 2005; Pan et al. 2008; Panniello et al. 2009]. Since a user may purchase the same package multiple times for her/his family members and many local travel packages are even consumed multiple times by the same user, there are still a lot of implicit ratings larger than 1, while over 60% of implicit ratings are 1.

Tourism data is naturally much sparser than movie data. For instance, a user may watch more than 50 movies each year, while there are not many people who will travel more than 50 times every year. In fact, many tourists only have three or five travel records in the dataset. To reduce the challenge of sparseness, we simply ignore users who have traveled less than four times as well as packages which have been purchased less than four times. After this data preprocessing, we have 34,007 pairs of ratings with 1,384 packages and 5,724 users. Thus, the sparseness of this data is still higher than

Table I. Some Characteristics of Travel Data

Statistics	User	Package
Min Number of Rating	4	4
Max Number of Rating	62	1,976
Average Number of Rating	5.94	24.57

Table II. The Notations of 9 Collaborative Filtering Methods

PMF	Probabilistic Matrix Factorization
vPMF	PMF + Vector-based Cost Representation
gPMF	PMF + Gaussian-based Cost Representation
RLFM	Regression-based Latent Factor Model for Gaussian response
LPMF	Logistic Probabilistic Matrix Factorization
vLPMF	LPMF + Vector-based Cost Representation
gLPMF	LPMF + Gaussian-based Cost Representation
LRLFM	Regression-based Latent Factor Model for Binary response
MMMF	Maximum Margin Matrix Factorization
vMMMF	MMMF + Vector-based Cost Representation
gMMMF	MMMF + Gaussian-based Cost Representation

the famous Movielens dataset[2] and Eachmovie[3] datasets. Finally, some statistics of the item-user rating matrix of our travel tour data are summarized in Table I.

Experimental Platform. All the algorithms were implemented in MatLab2008a. All the experiments were conducted on a Windows 7 with Intel Core2 Quad Q8300 and 6.00GB RAM.

6.2. Collaborative Filtering Methods

We have extended three different collaborative filtering models with two ways of representing user cost preference. Thus, we have in total nine collaborative filtering models in this experiment. Also, we compare our extended cost-aware models with regression-based latent factor models (RLFM) [Agarwal and Chen 2009], which take the cost information of packages as item features and incorporate such features into the matrix factorization framework. In Agarwal and Chen [2009], two versions of RLFM were proposed for both Gaussian and binary response. In the experiment of this article, both of them are used as additional baseline methods. To present the experimental comparisons easily, we denote these methods with acronyms in Table II.

6.3. The Details of Training

First, we train the PMF model and its extensions with the original ordinal ratings. For the PMF model, we empirically specify the parameters $\lambda_U = 0.05$ and $\lambda_V = 0.005$. For the vPMF and gPMF models, we use the same values for λ_U and λ_V, together with $\lambda_{C_U} = 0.2$ for the gPMF model. We specify $\sigma^2_{C_U} = 0.09$ for the gPMF model in the following. Also, we remove the global effect [Liu et al. 2010b] by subtracting the average rating of the training set from each rating before performing PMF-based models. Moreover, we initialize the cost vector (e.g., C_{U_i}) or the mean of the two-dimensional Gaussian distribution (e.g., $\mu_{C_{U_i}}$) for a user, with the average cost of all items rated by this user, while user/item latent feature vectors are initialized randomly.

[2] http://www.cs.umn.edu/Research/GroupLens

[3] HP retired the EachMovie dataset.

Second, we train LPMF, MMMF, and their extensions with the binarized ratings. We set different values for ratio α in order to empirically examine how the ratio affects the performances of LPMF, MMMF, and their extensions. For the LPMF-based models, the parameters are empirically specified as $\sigma_U^2 = 0.85$ and $\sigma_V^2 = 0.85$. In addition, $\sigma_{C_U}^2$ is set as 0.3 for the gLPMF model in order to constrain $S_G(C_{V_j}, \mathcal{G}(C_{U_i}))$ to be within the range [0,1], as mentioned in Section 4.3. For the MMMF-based approaches, the parameters are empirically specified as $C = 1.8$, and $\sigma_{C_U}^2 = 0.09$ for gMMMF. The cost vectors or the means of the two-dimensional Gaussian distribution of users and the user/item latent feature vectors are initialized in the same way as the PMF-based approaches.

Finally, we use cross-validation to evaluate the performance of different methods. We split all original ratings or positive ratings into two parts with a split ratio of 90/10. 90% of original or positive ratings are used for training, and 10% of them are used for testing. For each user-item pair in the testing set, the item is considered relevant to the user in this experiment. After getting the 90% of positive ratings, we sample the negative ratings with the set ratio α. We conduct the splitting five times independently and show the average results on five testing sets for all comparisons. In addition, we stop the iteration of each approach by limiting the same maximum number of iterations, which is set as 60 in this experiment.

6.4. Validation Metrics

We adopt Precision@K and mean average precision (MAP) [Herlocker et al. 2004] to evaluate the performances of all competing methods listed in Section 6.2. Moreover, we use root mean square error (RMSE) and cumulative distribution (CD) [Koren 2008] to examine performance of the PMF-based methods from different perspectives, while both RMSE and CD are less suitable for evaluating LPMF-based and MMMF-based models with the input of binary ratings.

Precision@K is calculated as

$$Precision@K = \frac{\sum_{U_i \in U} |T_K(U_i)|}{\sum_{U_i \in U} |R_K(U_i)|}, \tag{24}$$

where $R_K(U_i)$ are the top-K items recommended to user i, $T_K(U_i)$ denotes all truly relevant items among $R_K(U_i)$, and U represents the set of all users in a test set. MAP is the mean of average precision (AP) over all users in the test set. AP is calculated as

$$AP_u = \frac{\sum_{i=1}^{N} p(i) \times rel(i)}{number\ of\ relevant\ items}, \tag{25}$$

where i is the position in the rank list, N is the number of returned items in the list, $p(i)$ is the precision of a cut-off rank list from 1 to i, and $rel(i)$ is an indicator function equaling 1 if the item at position i is a relevant item; 0 otherwise. The RMSE is defined as

$$RMSE = \sqrt{\frac{\sum_{ij} \left(r_{ij} - \hat{r}_{ij}\right)^2}{N}}, \tag{26}$$

where r_{ij} denotes the rating of item j by user i, $\hat{r}_{ij}$ denotes the corresponding rating predicted by the model, and N denotes the number of tested ratings.

CD [Koren 2008] is designed to measure the qualify of top-K recommendations. CD measurement could explicitly guide people to specify K in order to contain the most interesting items in the suggested top-K set with certain probability. In the following, we briefly introduce how to compute CD with the testing set (more details about this

Table III. A Performance Comparison (10D Latent Features & $\alpha = 0.1$)

	Precision@5	Precision@10	MAP
PMF	0.0265	0.0154	0.0689
RLFM	0.0271	0.0167	0.0695
vPMF	**0.0285**	**0.0181**	**0.0718**
gPMF	**0.0301**	**0.0193**	**0.0811**
LPMF	0.0482	0.0339	0.1385
LRLFM	0.0486	0.0338	0.1394
vLPMF	**0.0497**	**0.0342**	**0.1420**
gLPMF	**0.0501**	**0.0351**	**0.1460**
MMMF	0.0545	0.0408	0.1571
vMMMF	**0.0552**	**0.0411**	**0.1606**
gMMMF	**0.0558**	**0.0413**	**0.1629**

validation method can be found in Koren [2008]). First, all the highest ratings in the testing set are selected. Assume that we have $\mathcal{M}$ ratings with the highest rating. For each item i with the highest rating by user u, we randomly select $\mathcal{C}$ additional items and predict the ratings by u for i and other $\mathcal{C}$ items. Then, we order these $\mathcal{C} + 1$ items based on their predicted ratings in decreasing order. There are $\mathcal{C} + 1$ different possible ranks for item i, ranging from the best case where none (0%) of the random $\mathcal{C}$ items appear before item i, to the worst case where all (100%) of the random $\mathcal{C}$ items appear before item i. For each of those $\mathcal{M}$ ratings, we independently draw the $\mathcal{C}$ additional items, predict the associated ratings, and derive a relative ranking (*RR*) between 0% and 100%. Finally, we analyze the distribution of overall $\mathcal{M}$ *RR* observations and estimate the cumulative distribution (CD). In our experiments, we specify $\mathcal{C} = 200$ and obtain 761 *RR* observations in total.

6.5. The Performance Comparisons

In this section, we present comprehensive experimental comparisons of all the methods with four validation measurements.

First, we examine how the incorporated cost information boosts different models in terms of different validation measurements. Table III shows the comparisons of all methods in terms of Precision@K and MAP. In Table III, the dimension of latent factors (e.g., U_i, V_j) is specified as 10 and ratio α is set as 0.1 for the sampling of negative ratings. Performances in terms of Precision@K are evaluated with different K values, that is, $K = 5$ and $K = 10$. For example, Precision@5 of vPMF and gPMF is increased by 7.54% and 13.58%, respectively. MAP of vPMF and gPMF is increased by 4.21% and 17.71%, respectively. Similarly, vLPMF (gLPMF) and vMMMF (gMMMF) outperform the LPMF, and MMMF models in terms of Precision@K and MAP. Also, vPMF (gPMF) and vLPMF (gLPMF) result in better performances than RLFM and LRLFM. In addition, we observe that MMMF, LPMF, and their extensions produce much better results than PMF and its extensions in terms of Precision@K and MAP. There are two main reasons why LPMF-based methods and MMMF-based methods perform better than PMF-based methods. First, the lost functions of LPMF and MMMF are more suitable for travel package data, because over 60% of known ratings are 1. Second, sampled negative ratings are helpful because the unknown ratings are actually not missed at random. For example, if one user has not consumed one package so far, this probably tells us that this user does not like this package. The sampled negative ratings somehow leverage this information and contribute to the better performance of LPMF-based and MMMF-based methods.

Table IV. A Performance Comparison (30D Latent Features & $\alpha = 0.1$)

	Precision@5	Precision@10	MAP
PMF	0.0271	0.0167	0.0704
RLFM	0.0280	0.0175	0.0714
vPMF	**0.0291**	**0.0184**	**0.0752**
gPMF	**0.0309**	**0.0194**	**0.0813**
LPMF	0.0485	0.034	0.1355
LRLFM	0.0489	0.0341	0.1397
vLPMF	**0.0498**	**0.0343**	**0.1423**
gLPMF	**0.0503**	**0.0354**	**0.1468**
MMMF	0.0618	0.0472	0.1723
vMMMF	**0.0629**	**0.0480**	**0.1737**
gMMMF	**0.0638**	**0.0487**	**0.1750**

Table V. A Performance Comparison in Terms of RMSE

	PMF	RLFM	vPMF	gPMF
10D Latent Features				
RMSE	0.4981	0.4963	**0.4951**	**0.4932**
30D Latent Features				
RMSE	0.4960	*0.4928*	**0.4933**	**0.4913**

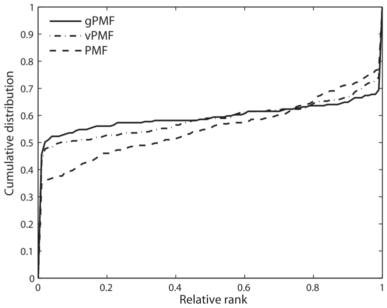


Fig. 4. A performance comparison in terms of CD (10D latent features).

Then we make the parallel comparisons in Table IV, where the dimension of latent factors is specified as 30 and $\alpha = 0.1$. By comparing Table IV with Table III, we find that increasing the dimension of latent factors could generally boost the performance of all nine methods. Furthermore, in both Table IV and Table III, the two-dimensional Gaussian distribution for modeling user cost preference leads to better results than the cost vector. All these results show that it is helpful to consider the cost information for travel recommendations and the way of representing user cost preference may influence the performance of cost-aware models.

For PMF-based methods, we also adopt RMSE and CD to evaluate their performances because they produce numerical predictions for unknown ratings. A performance comparison of PMF, vPMF, and gPMF with 10-dimensional and 30-dimensional latent features is shown in Table V. Also, we compare the performances of PMF-based models using the CD metric introduced in Section 6.4. Figure 4 shows the cumulative distribution of the computed percentile ranks for the three models over all 761 *RR* observations. Note that we use 10-dimensional latent features in Figure 4. As can be seen, both vPMF and gPMF models outperform the competing model, that is, PMF

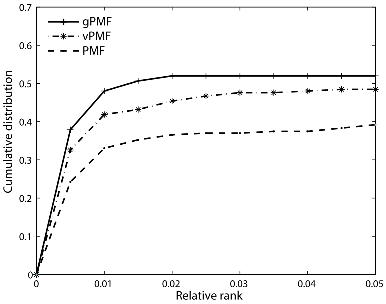


Fig. 5. A local performance comparison in terms of CD (10D latent features).

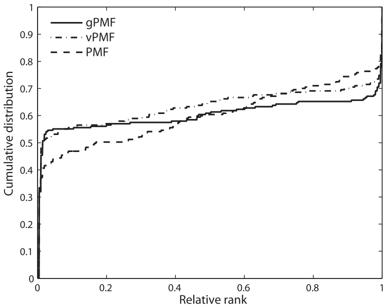


Fig. 6. A performance comparison in terms of CD (30D latent features).

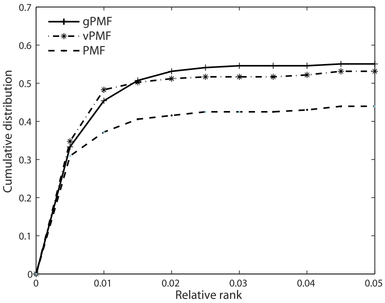


Fig. 7. A local performance comparison in terms of CD (30D latent features).

model. For example, considering point 0.1 on the x-axis, the CD value for gPMF at this point suggests that if we recommend the top-20 ones from randomly-selected 201 packages, approximately at least one package matches user interest and cost expectation with a probability of 53%. Since people are usually more interested in the top-5 or even top-3 packages, out of 201 packages, we zoom in on the head of the x-axis, which represents the top-K recommendations in a more detailed way. As shown in Figure 5, a more clear difference can be observed. For example, the gPMF model has a probability of 0.5 to suggest a highest-rated package before the other 198 packages. In other words, if we use gPMF to recommend the top-2 packages out of 201 packages, we can match user needs with a probability of 0.5. This outperforms PMF by over 60%. Also, vPMF leads to better performance than PMF. In addition, we show more comparisons in Figures 6 and 7 with 30-dimensional latent features, where a similar trend can be observed.

Table VI. A Performance Comparison (10D Latent Features & $\alpha = 0.3$)

	Precision@5	Precision@10	MAP
LPMF-based Methods			
LPMF	0.0466	0.0329	0.1325
vLPMF	**0.0472**	**0.033**	**0.1336**
gLPMF	**0.0475**	**0.034**	**0.1339**
MMMF-based Methods			
MMMF	0.053	0.0369	0.1507
vMMMF	**0.0537**	**0.0369**	**0.1525**
gMMMF	**0.0541**	**0.0372**	**0.1534**

Table VII. A Performance Comparison (30D Latent Features & $\alpha = 0.3$)

	Precision@5	Precision@10	MAP
LPMF-based Methods			
LPMF	0.0496	0.0340	0.1418
vLPMF	**0.0497**	**0.0341**	**0.1422**
gLPMF	**0.0502**	**0.0355**	**0.1430**
MMMF-based Methods			
MMMF	0.0557	0.0376	0.1555
vMMMF	**0.0563**	**0.0378**	**0.1585**
gMMMF	**0.0565**	**0.0379**	**0.1588**

Furthermore, we conduct a statistical significance test to show whether the performance improvement of cost-aware latent factor models is statistically significant. We do the statistical significance test based on the results in Tables III, IV, VI, and VII. Specifically, we first get the difference between the performance measurement of one cost-aware model (e.g., vPMF or gPMF) and the performance measurement of the corresponding original model (i.e., PMF, LPMF, or MMMF). For example, from Table III, we get the difference between Precision@5 of vPMF and Precision@5 of PMF, which is $0.0285 - 0.0265 = 0.002$, and the different between Precision@5 of gPMF and Precision@5 of PMF, which is $0.0301 - 0.0265 = 0.0036$. Along this line, from Table III, we get 18 samples of difference between the performance measurements of cost-aware models and those of original models (i.e., PMF, LPMF, and MMMF). And from Tables III, IV, VI, and VII, we get a total of 60 samples of difference between the performance measurements of cost-aware models and those of original models. While half of these samples are for cost-aware models with vector-based cost representation, half of them are for cost-aware models with Gaussian-based cost representation. The statistical significance test is conducted for each half of these 60 samples separately in order to examine the different statistical significance of improvement by different cost representations in cost-aware latent factor models. More specifically, the null hypothesis of each test is that there is no significant difference between the mean of the samples of difference and zero. For the 30 samples of difference for vector-based cost representation, the sample mean is around 0.0015; the sample standard deviation is around 0.0016. Then, we can derive that the one-tailed p-value is less than 0.0001. Thus, we can conclude that we should reject the null hypothesis, and the mean of the samples of difference is significantly larger than zero at the significance level of 0.01. For the another half of the 60 samples for Gaussian-based cost representation, we gain the same conclusion.

In addition, we further conduct a similar statistical significance test by using the relative difference between performance measurements of cost-aware models and those

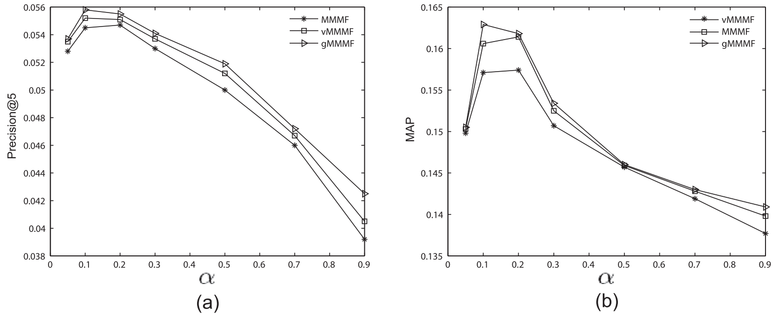


Fig. 8. Performances with different α (10D latent features).

of original models. For example, from Table III, we get the relative difference between Precision@5 of vPMF and Precision@5 of PMF as $(0.0285 - 0.0265)/0.0265 = 0.07547$. After obtaining all 60 samples of such relative difference, we conduct a similar statistical test on each half of these samples. The null hypothesis of each test is that there is no significant difference between the mean of samples of relative difference and μ_0. μ_0 is the assumed population mean of relative difference of performance measurements. For the vector-based cost representation, the conclusion is that the mean of relative difference is significantly larger than 0.018 at the significance level of 0.05. For the Gaussian-based cost representation, the conclusion is that the mean of relative difference is significantly larger than 0.037 at the significance level of 0.05.

6.6. The Performances with Different Values of α and D

As we mentioned in Section 6.3, ratio α may influence the results of LPMF- and MMMF-based methods. To examine this point, we set the ratio α as $\alpha = 0.3$ and produce another set of results by LPMF- and MMMF-based methods, as shown in Table VI, where the dimension of latent factors is set as 10. By comparing with Table III, we can observe that increasing α from 0.1 to 0.3 actually causes the performances of LPMF- and MMMF-based methods to generally decrease. A similar trend can be observed in Table VII, where the dimension of latent factors is 30. This is probably caused by the increased negative ratings by sampling being noisy or not accurate. Though more accurate training ratings should generally yield better results, more noisy or inaccurate negative ratings may lead to biased parameter estimations and worse predictions. On the contrary, fewer but accurate sampled negative ratings may result in better performance. To further examine this point, we show the performance of MMMF-based models with a series of α values in Figure 8, where the dimension of latent factors is also 10. As can be seen in Figure 8, the performances in terms of Precision@5 and MAP first increase and then decrease as ratio α is increased from 0 to 1.

By comparing Table III and Table IV, we can observe that increasing the dimension of latent factors tends to lead to better performance. To further investigate this observation, in Figure 9, we show the Precision@10 of latent factor models versus the dimension of latent features. As can be seen, Precision@K of all methods gradually increases when the dimension of latent features becomes larger.

6.7. The Performances on Different Users

For most collaborative filtering models, the prediction performance for users with different numbers of observed ratings usually varies a lot. Particularly, performances on users with very few ratings may be quite bad for traditional collaborative filtering

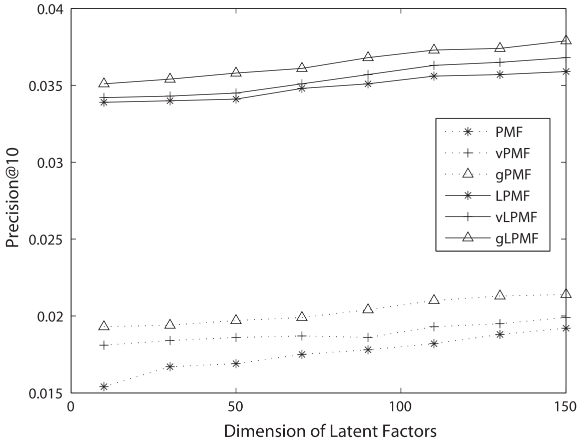


Fig. 9. Performances with different D ($\alpha = 0.1$).

Table VIII. Performances on Different Users (10D Latent Features & $\alpha = 0.1$)

Groups	"1–5"	"6–10"	"11–20"	"21–30"	">30"
			PMF		
Precision@5	0.0211	0.0295	0.0482	0.072	
MAP	0.0586	0.0784	0.0902	0.0958	0.0054
			vPMF		
Precision@5	0.0223	0.0306	0.0498	0.096	
MAP	0.0573	0.0865	0.0959	0.1228	0.005
			gPMF		
Precision@5	0.0259	0.0308	0.053	0.096	
MAP	0.0752	0.086	0.0937	0.1154	0.0045
			LPMF		
Precision@5	0.036	0.0488	0.0738	0.1419	0.0857
MAP	0.1109	0.1466	0.1836	0.2118	0.1722
			vLPMF		
Precision@5	0.0386	0.0496	0.0744	0.1419	0.0857
MAP	0.1186	0.1471	0.1863	0.2120	0.2613
			gLPMF		
Precision@5	0.0391	0.0500	0.0748	0.1426	
MAP	0.1191	0.1479	0.1869	0.2128	0.2621
			MMMF		
Precision@5	0.0483	0.0521	0.0719	0.0786	0.0889
MAP	0.143	0.1626	0.1766	0.1436	0.1266
			vMMMF		
Precision@5	0.0499	0.0525	0.0694	0.0857	0.1111
MAP	0.1487	0.1631	0.1775	0.1751	0.2276
			gMMMF		
Precision@5	0.0502	0.0527	0.0695	0.0860	0.1117
MAP	0.1488	0.1636	0.1782	0.1754	0.2279

models. However, the user and item cost information play as an effective constraint for tuning prediction via the similarity weight. Thus, our extended models with cost information are expected to perform better on users with fewer ratings than the traditional

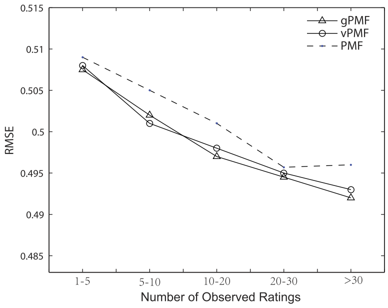


Fig. 10. The performances on different users (10D latent features).

Table IX. Performances with Tail Users/Packages (30D Latent Features & $\alpha = 0.1$)

	Precision@5	Precision@10	MAP
PMF	0.0253	0.0148	0.0644
vPMF	0.0254	0.0157	0.0658
gPMF	0.0265	0.0164	0.0663
LPMF	0.043	0.0305	0.1286
vLPMF	0.0441	0.0324	0.1292
gLPMF	0.0462	0.0339	0.1315
MMMF	0.0553	0.0416	0.1651
vMMMF	0.0561	0.0431	0.1668
gMMMF	0.0578	0.0454	0.1683

models. In order to examine this potential, we first group all users based on the number of observed ratings in the training set and then compare the performances of different methods over different user groups. Specifically, users are grouped into five classes: 1–5, 6–10, 11–20, 21–30, and > 30. For example, the 1–5 group denotes that the number of observed ratings per user in the training set is between 1 and 5.

Table VIII shows the performance of different methods in terms of Precision@K and MAP. In Table VIII, the dimension of latent factors is 10, and ratio α is 0.1. As can be seen in Table VIII, our extended models with the incorporated cost consistently outperform traditional methods. For example, for the 1–5 group, MAP of gPMF, gLPMF, and gMMMF is increased by 13.26% on average. In addition, the comparisons of RMSE among PMF-based methods are shown in Figure 10, where the dimension of latent factors is also 10 and the RMSE is the value of final iteration for each method.

Performance with Tail Packages and Users. In Table IX, we demonstrate the performance of different methods with all tail users and packages. Tail users are those who have consumed less than four different travel packages. Tail packages are those which have been purchased by less than four different users. These tail users or packages usually contribute a lot to the high sparseness of recommendation data [Park and Tuzhilin 2008], and eventually cause the average performance of collaborative filtering methods to decrease [Park and Tuzhilin 2008]. As shown in Table IX, Precision@K or MAP are generally lower than those in Table IV. While the long tail is a general and important topic in the recommender systems field, it is not the focus of this article.

6.8. The Learned User Cost Information

By training cost-aware latent factor models, we can not only produce better recommendation result, as shown in Sections 6.7 and 6.5, but also learn latent user cost

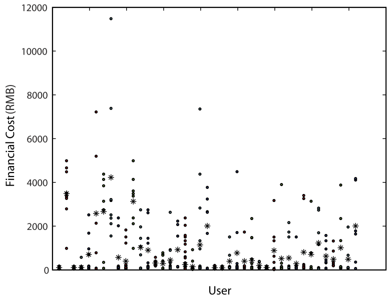


Fig. 11. Financial cost observations and learned cost features of 40 users.

information. In the following, we illustrate user cost information learned by our models and demonstrate how learned user cost information helps with travel package recommendation and customer clustering or segmentation.

Since we normalize the package cost vectors into [0, 1] before feeding into our models, the learned user cost features (C_U and μ_{C_U}) via our models have a similar scale as normalized package cost vectors. To visualize the learned C_U, we first restored the scale of user cost features (C_U and μ_{C_U}) by using the inverse transformation of Min-Max normalization. Figure 11 shows the financial cost feature of C_U by the vPMF model for randomly-selected 40 users, where each user corresponds to a column of vertically-distributed points. For example, for the rightmost vertical points, the *star* represents the learned user financial cost feature, and the *dots* represent the financial cost of packages, which are rated by this specific user in the training set. As we can see, the learned user financial cost feature is relatively representative. However, there is still obvious variance among the cost features of packages by some users. That is why we apply the Gaussian distribution to model user cost preference. In Figure 12, we visualize the learned μ_{C_U} by gPMF for randomly-selected 12 users. For each subfigure of Figure 12, we directly plot the learned two-dimensional $\mu_{C_{U_i}}$ (without inverse transformation) for individual users and all normalized two-dimensional cost vectors of packages, which are rated by the user in the training set. Again, $\mu_{C_{U_i}}$ is represented as the *star* and the *dots* represent the package cost vector.

The learned user cost information, together with the latent features of user and packages, can help recommend packages, which are more similar to user cost preference, and can match user general interest at the same time. To demonstrate this point, we randomly selected ten users (denoted as $u_1, \cdots, u_{10}$) from the test set. For each user, we recommend the top-5 travel packages (denoted as 1st package, 2nd package, 3rd package, 4th package, and 5th package) based on the predicted ratings with two methods (i.e., LPMF and vLPMF). For each user u_i and package p recommended to this user by individual method, we compute the similarity (i.e., specifically, cosine similarity) between the cost of this package p and the learned cost information of this user, and the similarity (i.e., specifically, cosine similarity) between the latent feature of this user and the latent feature of this package. The learned cost information of a user here is the two-dimensional user cost vector learned via the vLPMF model, and the latent feature of a user or a package is the D-dimensional vector learned by LPMF or vLPMF. Since we have ten users, for a single method (LPMF or vLPMF), we can get ten groups of these two types of similarity for all 1st packages recommended to different users. Note that the 1st package may be different for different users and different methods. For the 2nd, 3rd, 4th, or 5th packages, we similarly can get ten groups of these two types of similarity. Then, for the individual method, we average the two types of

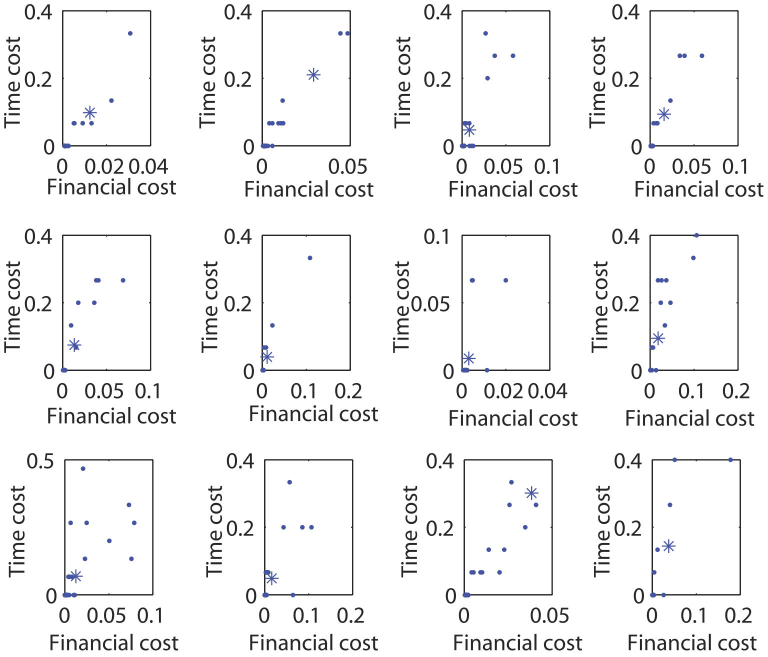


Fig. 12. Cost observations and learned Gaussian parameters of 12 users.

Table X. Comparisons of Similarity of Cost and Latent Features

	1st Pkg	2nd Pkg	3rd Pkg	4th Pkg	5th Pkg
LPMF&Ave. Simi. of Cost	0.421	0.398	0.407	0.376	0.299
vLPMF&Ave. Simi. of Cost	0.734	0.745	0.691	0.693	0.625
LPMF&Ave. Simi. of Latent Feature	0.876	0.813	0.799	0.787	0.776
vLPMF&Ave. Simi. of Latent Feature	0.869	0.802	0.784	0.773	0.765

similarity over all 1st packages recommended to different users. For the 2nd, 3rd, 4th, and 5th packages, we get a similar average similarity of two types. Finally, we show the results in Table X, where 1st Pkg means the first package. As can be see, the average similarity of cost of vLPMF is generally larger than that of LPMF. For example, for the 1st package, the average similarity of cost for vLPMF is 0.734, which is much larger than the corresponding one of LPMF, which is 0.421. For the average similarity of the latent feature, vLPMF is quite similar as LPMF for packages ranked at different positions. To be more specific, we show the information of the 1st packages recommended to three users in Table XI, where we also show the information of partial travel packages consumed by these three users in the historical data. For instance, for user1, most of her/his consumed travel packages in the past are priced between $20 and $100, and the duration is between 1 and 2 days. Both 1st packages recommended by LPMF andvLPMF should be very interesting to user1, because she/he showed her/his interest in theme parks. However, the price and duration of the 1st package recommended via LPMF model is $1,070 and 5 days, which are probably beyond user1's financial and time affordability. On the contrary, the price and duration of the 1st package recommended via vLPMF clearly fall into the ranges identified from her/his travel history (i.e., [20, 100] for price and $[1Day, 2Days]$ for duration). Thus, there should be a better chance that user1 will consume the 1st package recommended via vLPMF rather than that via LPMF. For other users in Table XI, we can observe similar comparisons.

Table XI. Recommended Packages and Consumed Packages of Three Users

	User1	User2	User3
1st Pkg (LPMF)	Disney Park: ($1070,5 days)	Leizhou Island: ($200, 2 days)	Japan Gourmet Trip: ($1900, 5 days)
1st Pkg (vLPMF)	Xinhui Water: Park: ($50, 1 day)	Fiji Island: ($1500, 5 days)	Yunan Village Gourmet Trip: ($500, 2 days)
Consumed Pkgs	Changlong Water Park: ($40,1 day) Pearl Land: ($80,2 day) Amusement Land: ($99,2 day) Orient Culture Park: ($80,2 day) Shunde Eco Park: ($20,1 day)	Hainai Island: ($800, 4 days) Bali Island: ($1699, 6 days) Hawaii Beach: ($1999, 7 days) Maldives Island: ($2000, 7 days) Rizhao: ($700, 4 days)	Macau Gourmet Trip: ($200, 2 days) Taibei Gourmet Trip: ($458, 3 days) Fuzhou Gourmet Trip: ($349, 2 days) Guangzhou Gourmet Trip:($399, 2 days) HongKong Gourmet Trip: ($416, 2 days)

Table XII. Comparison of Variance

Results on Clu						
	C1	C2	C3	C4	C5	Average
Financial Variance	0.00091	0.00102	0.00079	0.00086	0.00114	0.000944
Time Variance	0.0292	0.0012	0.0321	0.0093	0.0125	0.0169
Results on Clu^+						
	C1	C2	C3	C4	C5	Average
Financial Variance	0.00073	0.00105	0.00047	0.00090	0.00035	**0.00070**
Time Variance	0.0193	0.0009	0.0214	0.0098	0.0133	**0.0129**

The learned user latent features, for example U_i, with PMF, LPMF, or MMMF models, can be used to group users or customers. We argue that the learned user cost information, in addition to user latent features, can improve customer clustering or segmentation. In order to show this effect, we first cluster users with latent features learned by PMF by representing each user with her/his latent feature vector. We use a K-means algorithm to perform clustering and denote the clustering result as Clu. Then, with the same clustering method, we cluster users with both user latent features and user cost information, that is, C_U or μ_{C_U}, learned by vPMF and gPMF. Now each user is represented by a vector containing her/his latent features and cost vector C_{U_i} or $\mu_{C_{U_i}}$. We denote this clustering result as Clu^+. However, there is no available benchmark to evaluate these two clustering results with traditional external clustering validation measurements [Wu et al. 2009]. To this end, we leverage the explicit cost information of items to make comparisons between these two clustering results. Specifically, for each user within a cluster, we can get the average financial/time cost of all travel packages, which are consumed by this user. After obtaining the average financial/time cost of each user of one cluster, we can get the variances of such average financial/time costs of all users for this cluster. Table XII shows the comparisons of these two clustering results in terms of such variance. Here, the number of clusters is specified as 5 for the K-means algorithm, and C1 indicates cluster 1. Also in Table XII, Clu^+ is obtained by using μ_{C_U} learned by gPMF in addition to user latent features. As can be seen from Table XII, the average variance over five clusters of Clu^+ is much less than that of Clu. From this perspective, we can see that learned user cost information improves results of customer clustering or segmentation.

Table XIII. Comparison of Model Efficiency (10D Latent Features)

	PMF	vPMF	gPMF
Training Time (Sec)	3.411	4.894	10.878
	LPMF	vLPMF	gLPMF
Training Time (Sec)	63.452	81.411	201.329
	MMMF	vMMMF	gMMMF
Training Time (Sec)	82.306	98.187	187.250

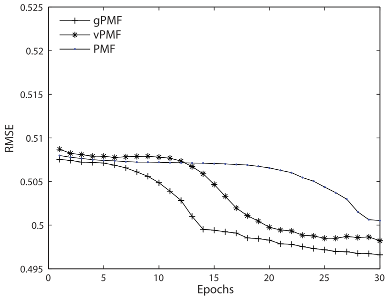


Fig. 13. An illustration of the convergence of RMSEs (10D latent features).

6.9. An Efficiency Analysis

As stated in Section 3.3, the computational complexity of the proposed approaches is linear with respect to the number of ratings. This indicates that the extended models are theoretically scalable for very large data. Here, we would like to show the efficiency of all the methods in this experiment. Table XIII shows the training time of all nine models. Here, we used the 10-dimensional latent features. Since there is some additional cost for computing similarity functions and updating cost vectors or parameters of Gaussian distribution for the six cost-aware models, more time is required for these six models, for example, vMMMF and gMMMF. In addition, the Gaussian distribution causes more time than the two-dimensional vector, because there is one more regularization item caused by the Gaussian prior in the objective functions. But, the computing time of cost-aware models is still linearly increasing as the number of observed ratings increases, as discussed in Section 3.3. In addition, we show the convergence of RMSEs on the test set for PMF-based methods in Figure 13. As can be seen, vPMF and gPMF can quickly converge to relatively low RMSEs after the first 25 rounds of iterations.

To further speed up the computation of cost-aware models for big datasets, we may leverage MapReduce to distribute the computing onto clusters. By using MapReduce clusters, we can partition the data and arrange the computation to maximize data locality and parallelism. For instance, as shown in Liu et al. [2010a], using MapReduce can make the matrix factorization scalable to million-by-million matrices with billions of nonzero values. Due to the focus of this article, we will not discuss this in detail.

7. CONCLUSION AND DISCUSSION

In this article, we studied the problem of travel tour recommendation by analyzing a large amount of travel logs collected from a travel agent company. One unique characteristic of tour recommendation is that there are different financial and time costs associated with each travel package. Different tourists usually have different affordability for these two aspects of cost. Thus, we explicitly incorporated observable and unobservable cost factors into the recommendation models. Specifically, we first proposed two ways to model user cost preference. With these two ways of representing user cost preference, we incorporated the cost information into three classic latent factor models for collaborative filtering, including the probabilistic matrix factorization (PMF) model, the logistic probabilistic matrix factorization (LPMF) model, and the maximum margin matrix factorization (MMMF) model. When applied to real-world travel tour data, the extended PMF, LPMF, and MMMF models showed consistently better performances for travel tour recommendation than classic PMF, LPMF, and MMMF models which do not consider the cost information. Furthermore, the extended MMMF and LPMF models lead to better performance improvement than the extended PMF models. Finally, we have demonstrated that latent user cost information learned by these models can help to perform customer segmentation for travel companies.

Discussion. People may argue that some dimensions of learned latent factors of users/packages might somehow capture cost factors implicitly. However, it is hard to identify which dimensions correspond to these cost factors. At the same time, in our application (and in many others), the cost information is given explicitly, and it is every natural to incorporate it into the model(s)—that is what we do in this article. Furthermore, through extensive experimentation, we showed that this additional information, indeed, boosts the performance of collaborating filtering methods that do not take this cost information into account.

As shown in Table IX, tail users/packages result in lower performances for different collaborative filtering methods. Since the long tail is a major challenge in the recommendation field and is not the focus of this article, we would like to study this topic for travel package recommendations in the future.

Like cost information, time sensitivity is another important factor for travel package recommendations. For example, Orlando trips may be more attractive to people in the Northeast of the U.S. during winter. However, since the focus of the article is on incorporating economic indicators, such as costs, into recommendation models, we would like to work on time sensitivity as a topic of future research.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their helpful and constructive comments that greatly contributed to improving this article. The authors would also like to thank the Editor-in-Chief and the editor for their generous comments and support during the review process.

REFERENCES

Adams, R. P., Dahl, G. E., and Murray, I. 2010. Incorporating side information in probabilistic matrix factorization with gaussian processes. arXiv:1003.4944.

Adomavicius, G. and Tuzhilin, A. 2005. Towards the next generation of recommender systems: A survey of the state-of-the art and possible extensions. *IEEE Trans. Knowl. Data Eng. 17,* 6, 734–749.

Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A. 2005. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Trans. Inf. Syst. 23,* 1, 103–145.

Agarwal, D. and Chen, B. C. 2009. Regression-based latent factor models. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 19–28.

Ardissono, L., Goy, A., Petrone, G., Segnan, M., and Torasso, P. 2002. Ubiquitous user assistance in a tourist information server. In *Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*. 14–23.

Baltrunas, L., Ludwig, B., Peer, S., and Ricci, F. 2011a. Context-aware places of interest recommendations for mobile users. In *Proceedings of the International Conference on Human-Computer Interaction*. 531–540.

Baltrunas, L., Ricci, F., and Ludwig, B. 2011b. Context relevance assessment for recommender systems. In *Proceedings of the International Conference on Intelligent User Interfaces*.

Bell, R. M. and Koren, Y. 2007. Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In *Proceedings of the IEEE International Conference on Data Mining*. 43–52.

Burke, R. D. 2007. Hybrid Web recommender systems. In *The Adaptive Web*. Lecture Notes in Computer Science, vol. 4321, 377–408.

Carolis, B. D., Mazzotta, I., Novielli, N., and Silvestri, V. 2009. Using common sense in providing personalized recommendations in the tourism domain. In *Proceedings of the Workshop on Context-Aware Recommender Systems*.

Cena, F., Console, L., Gena, C., Goy, A., Levi, G., Modeo, S., and Torre, I. 2006. Integrating heterogeneous adaptation techniques to build a flexible and usable mobile tourist guide. *AI Commun. 19,* 4, 369–384.

Chen, L.-S., Hsu, F.-H., Chen, M.-C., and Hsu, Y.-C. 2008. Developing recommender systems with the consideration of product profitability for sellers. *Inf. Sci. 178*, 4, 1032–1048.

Cheverst, K., Davies, N., Mitchell, K., Friday, A., and Efstratiou, C. 2000. Developing a context-aware electronic tourist guide: Some issues and experiences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 17–24.

Das, A., Mathieu, C., and Ricketts, D. 2010. Maximizing profit using recommender systems. In *Proceedings of the International Conference on World Wide Web*.

Deshpande, M. and Karypis, G. 2004. Item-based top-N recommendation algorithms. *ACM Trans. Inf. Syst. 22*, 1, 143–177.

Ge, Y., Xiong, H., Tuzhilin, A., Xiao, K., Gruteser, M., and Pazzani, M. J. 2010. An energy-efficient mobile recommender system. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 899–908.

Ge, Y., Liu, Q., Xiong, H., Tuzhilin, A., and Chen, J. 2011a. Cost-aware travel tour recommendation. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 983–991.

Ge, Y., Xiong, H., Tuzhilin, A., and Liu, Q. 2011b. Collaborative filtering with collective training. In *Proceedings of the ACM Conference on Recommender Systems*. 281–284.

Gu, Q., Zhou, J., and Ding, C. H. Q. 2010. Collaborative filtering weighted nonnegative matrix factorization incorporating user and item graphs. In *Proceedings of the SIAM International Conference on Data Mining*. 199–210.

Hao, Q., Cai, R., Wang, C., Xiao, R., Yang, J.-M., Pang, Y., and Zhang, L. 2010. Equip tourists with knowledge mined from travelogues. In *Proceedings of the International Conference on World Wide Web*.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., John, and Riedl, T. 2004. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst. 22*, 1, 5–53.

Hofmann, T. 2004. Latent semantic models for collaborative filtering. *ACM Trans. Inf. Syst. 22*, 1, 89–115.

Hosanagar, K., Krishnan, R., and Ma, L. 2008. Recommended for you: The impact of profit incentives on the relevance of online recommendations. In *Proceedings of the International Conference on Information Systems*.

Huang, Z., Chung, W., and Chen, H. 2004. A graph model for e-commerce recommender systems. *J. Amer. Soc. Inf. Sci. Technol. 55*, 3, 259–274.

Huang, Z., Li, X., and Chen, H. 2005. Link prediction approach to collaborative filtering. In *Proceedings of the Joint Conference on Digital Libraries*. 141–142.

Jannach, D. and Hegelich, K. 2009. A case study on the effectiveness of recommendations in the mobile internet. In *Proceedings of the ACM Conference on Recommender Systems*. 205–208.

Kenteris, M., Gavalas, D., and Economou, D. 2011. Electronic mobile guides: A survey. *Pers. Ubiq. Comput. 15,* 1.

Koren, Y. 2008. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 426–434.

Liu, C., chih Yang, H., Fan, J., He, L.-W., and Wang, Y.-M. 2010a. Distributed nonnegative matrix factorization for web-scale dyadic data analysis on MapReduce. In *Proceedings of the 19th International Conference on World Wide Web*. 681–690.

Liu, N. N., Xiang, E., Zhao, M., and Yang, Q. 2010b. Unifying explicit and implicit feedback for collaborative filtering. In *Proceedings of the 19th ACM Conference on Information and Knowledge Management*.

Liu, N. N., Zhao, M., Xiang, E. W., and Yang, Q. 2010c. Online evolutionary collaborative filtering. In *Proceedings of the ACM Conference on Recommender Systems*. 95–102.

Liu, Q., Chen, E., Xiong, H., and Ding, C. H. Q. 2010d. Exploiting user interests for collaborative filtering: Interests expansion via personalized ranking. In *Proceedings of the ACM Conference on Information and Knowledge Management*. 1697–1700.

Lu, Z., Agarwal, D., and Dhillon, I. S. 2009. A spatio-temporal approach to collaborative filtering. In *Proceedings of the ACM Conference on Recommender Systems*. 13–20.

Ma, H., King, I., and Lyu, M. R. 2009. Learning to recommend with social trust ensemble. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*. 203–210.

Marlin, B. 2003. Modeling user rating profiles for collaborative filtering. In *Proceedings of the 17th Annual Conference on Neural Information Processing Systems*.

Marlin, B. M. and Zemel, R. S. 2007. Collaborative filtering and the missing at random assumption. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*. 267–275.

Marlin, B. M. and Zemel, R. S. 2009. Collaborative prediction and ranking with non-random missing data. In *Proceedings of the ACM Conference on Recommender Systems*. 5–12.

Pan, R. and Scholz, M. 2009. Mind the gaps: Weighting the unknown in large-scale one-class collaborative filtering. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 667–676.

Pan, R., Zhou, Y., Cao, B., Liu, N. N., Lukose, R., Scholz, M., and Yang, Q. 2008. One-class collaborative filtering. In *Proceedings of the IEEE International Conference on Data Mining*. 502–511.

Panniello, U., Tuzhilin, A., Gorgoglione, M., Palmisano, C., and Pedone, A. 2009. Experimental comparison of pre- vs. post-filtering approaches in context-aware recommender systems. In *Proceedings of the ACM Conference on Recommender Systems*. 265–268.

Park, M.-H., Hong, J.-H., and Cho, S.-B. 2007. Location-based recommendation system using bayesian user's preference model in mobile devices. In *Proceedings of the International Conference on Ubiquitous Intelligence and Computing*.

Park, Y.-J. and Tuzhilin, A. 2008. The long tail of recommender systems and how to leverage it. In *Proceedings of the ACM Conference on Recommender Systems*.

Rennie, J. D. M. and Srebro, N. 2005. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the International Conference on Machine Learning*. 713–719.

Salakhutdinov, R. and Mnih, A. 2008. Probabilistic matrix factorization. In *Proceedings of the 22nd Annual Conference on Neural Information Processing Systems*.

Setten, M. V., Pokraev, S., Koolwaaij, J., and Instituut, T. 2004. Context-aware recommendations in the mobile tourist application compass. In *Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*. 235–244.

Srebro, N., Rennie, J., and Jaakkola, T. 2005. Maximum margin matrix factorizations. In *Proceedings of the 19th Annual Conference on Neural Information Processing Systems*.

Woerndl, W., Huebner, J., Bader, R., and Vico, D. G. 2011. A model for proactivity in mobile, context-aware recommender systems. In *Proceedings of the ACM Conference on Recommender Systems*.

Wu, J., Xiong, H., and Chen, J. 2009. Adapting the right measures for k-means clustering. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 877–886.

Xiong, L., Chen, X., Huang, T.-K., Schneider, J. G., and Carbonell, J. G. 2010. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *Proceedings of the SIAM International Conference on Data Mining*. 211–222.

Xue, G., Lin, C., Yang, Q., Xi, W., Zeng, H., Yu, Y., and Chen, Z. 2005. Scalable collaborative filtering using cluster-based smoothing. In *Proceeding of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. 114–121.

Yang, S.-H., Long, B., Smola, A. J., Sadagopan, N., Zheng, Z., and Zha, H. 2011. Like like alike: Joint friendship and interest propagation in social networks. In *Proceedings of the International Conference on World Wide Web*. 537–546.

Yu, Z., Zhou, X., Zhang, D., Chin, C.-Y., Wang, X., and Men, J. 2006. Supporting context-aware media recommendations for smart phones. *IEEE Perv. Comput. 5,* 3, 68–75.

Received March 2012; revised November 2012, May 2013, August 2013; accepted October 2013