Performance of Nonnegative Latent Factor Models with β-distance Functions in Recommender Systems

Ye Yuan
Chongqing Key Laboratory of Big Data and Intelligent Computing
Chongqing Institute of Green and Intelligent Technology
Chinese Academy of Sciences
Chongqing 400714, China
yuanye@cigit.ac.cn

Xin Luo
Chongqing Key Laboratory of Big Data and Intelligent Computing
Chongqing Institute of Green and Intelligent Technology
Chinese Academy of Sciences
Chongqing 400714, China
luoxin8321@sina.com

***Abstract*—Nonnegative latent factor (NLF) models are able to well represent high-dimensional and sparse (HiDS) matrices filled with nonnegative data, which are frequently encountered in industrial applications like recommender systems. Current NLF models mostly adopt the Euclidean distance or Kullback-Leibler divergence as the objective function, which actually correspond to the special case of β=2 or 1 in β-distance functions. With β not limited in such special cases, an NLF model's performance varies, making it highly attractive to investigate the resultant performance variations. We first divide the β-distance-based function into three categories, i.e., β=0, β=1, and $\beta \neq 0$ or 1, respectively. Subsequently, we deduce the nonnegative training rules corresponding for different kinds of objectives to achieve different NLF models. Experimental results on industrial matrices indicate that the frequently adopted cases of β=2 or 1 are probably not able to achieve the most accurate or efficient models. It is promising to further improve the performance of NLF models with carefully-tuned β-distance functions as the training objective.**

Keywords—High-dimensional and Sparse Matrix, Non-negative Latent Factor Analysis, Beta Distance, Objective Function

I. INTRODUCTION

The rapid expansion of world-wide-web has brought people huge convenience as well as causing a serious problem of efficiency, due to the great difficulty to filter desired information out of billions bytes. Meanwhile, online consumption becomes indispensable for people's daily life. Thousands of products are provided online by numerous online retailers. However, such massive online information leads to information overload, which has greatly reduced the utilizing ratio of information. It is highly demanded to develop robust, intelligent and efficient models for enhancing information utilization. Recommender systems, which can play the role of intelligent agents assisting people to perform efficient information filtering, have attracted people's great attention. Emerging in 1990s, related research shows that recommender systems can address the problem of information overload by connecting information to potentially related people actively, rather than passively, according to their information usage history [1 -5].

In general, a recommender system is a learning system consisting of three fundamental kinds of entities, i.e., users, items (e.g., movies, books), and user-item usage data (e.g., scores, comments). Such a system aims at figuring out useful patterns describing the hidden connections among users and items from the usage history, and then makes predictions for possible user-item links according these patterns. To deal with such tasks, researchers have proposed many models during the past 20 years [1, 6]. These models can be further divided into several categories, where a very important one is collaborative filtering (CF) [1, 5 -9]. Owing to the computational efficiency and ease of implementation, CF-based recommenders have been widely studied and adopted in industrial applications during the past decade [1 -5].

A CF-based recommender usually models the user preferences on involved items into a user-item rating matrix, whose entries are commonly proportional to the corresponding user-item preferences. With constantly increasing number of user and items in real systems, only a finite item set can be operated by each user, making this rating-matrix very sparse with a mass of missing data. Hence, the resulting user-item rating matrix is a high-dimensional and sparse (HiDS) matrix whose most entries are unknown. On the other hand, if these missing values are appropriately estimated, the system will probably able to predict a user's preferences which are not observed yet. Therefore, the main task of a CF-based recommender is to implement efficient missing-data-estimation, i.e., to estimate unknown ratings based on known ones subject to globally high accuracy and other requirements [1, 4 -6, 8, 10].

A CF-based recommender can be can be implemented through several approaches. The up-to-date progress in this area unveils that matrix factorization (MF)-based models, which are also known as latent factor (LF) models [5 -9, 11, 12] are highly accurate and scalable under many circumstances [1 -3]. The principle of an LF model is to construct a low-rank estimate to the original HiDS matrix based on its known entries. It works by mapping both items and users into the same latent feature space, training desired user/item features on known entries in the original rating matrix, and then generating predictions for unknown ratings heavily relying on the inner products of related user-item feature-vector pairs.

Inside an LF-based model, the user/item latent features are the key model factors which should be trained and stored with care. Because of the usual low-rank of the user-item rating matrix, the dimension of the latent feature space can be set low without impairing the resultant model's prediction accuracy for

Corresponding Author: *Xin Luo*. This work is supported in part by the National Natural Science Foundation of China under Grant 61772493, Grant 91646114, Grant 61602434 and Grant 61702475, and in part by the Pioneer Hundred Talents Program of Chinese Academy of Sciences
978-1-5386-5053-0/18/$31.00 ©2018 IEEE

the missing data in the target HiDS matrix. Hence, the size of feature matrices in a LF-based model is linearly related to user and item counts [1 -3], making its storage complexity low and are easy to resolve in real applications. However, most LF models do not fulfill the nonnegative constraints, i.e., the obtained features of users/items might be negative. As indicated in prior works [13 -16], Nonnegative LFs can better represent the actual meanings hidden in the target data, as well as make the obtained model more interpretable with fine representativeness of the target rating-matrix [17 -19]. Therefore, it is necessary to investigate nonnegative LF models subject to the non-negativity constraints.

With a complete target matrix, several approaches can be adopted to conduct a nonnegative analysis process. Paatero and Tappe [20] apply alternating least squares to implementing the LF with restricting negative features at zero to maintain their non-negativity. Lee and Seung [21] derive the nonnegative multiplicative update for the desired feature matrices, which can maintain the non-negativity of initially nonnegative features. Lin [22] use projected gradient decent to implement a nonnegative matrix factorization model. These methods and their extensions [23, 24] can well factorize a complete matrix under the non-negativity constraints, but are not applicable for CF-based recommenders. This is because the problem of CF possess the sparse nature, and missing entries in a target HiDS matrix are much more than given ones. Luo et al. [9, 17, 25] propose the nonnegative latent factor (NLF) model to suit the sparsity of an HiDS matrix. NLF works based the single latent factor-dependent non-negative multiplicative update (SLF-NMU) [7, 9], which makes it able to handle the incomplete data in an HiDS matrix under the non-negativity constraints.

Although the NLF model addresses HiDS matrix non-negatively and efficiently, it is based on an objective function defined by the Euclidean distance, which is only a special case of β=2 in the β-distance functions. Note that for an LF model, the objective function is vital in deciding its various characteristics. A properly designed objective function [26 - 32] may greatly improve its convergence rate and prediction accuracy. However, existing LF models including the NLF model all adopts the Euclidean distance or Kullback-Leibler divergence as the objective function [4, 33 -35], which actually corresponds to the special case of β=2 or 1 in β-distance-based functions only. Consequently, it is highly interesting to investigate the effects of different β-distance-based functions as the training objective function in NLF models.

This paper focuses on building a series of NLF models based on different β-distance functions and validating their performance in context of recommender systems. To do so, we first review the principle of the NLF model Then we categorize β-distance-based objective functions into three cases: β=0, β=1, $\beta\neq$0 and 1. Note that we constrain β to be positive in this paper because the distance functions become meaningless with β negative. Subsequently, we deduce different updating rules of NLF models with the Tikhonov regularized version of the three cases mentioned above as the objective functions, respectively. According to the authors' best knowledge, such efforts are never seen in previous works.

The rest of this paper is organized as follows. Section II gives the preliminaries. Section III presents the methods. Section IV gives the experimental results and discusses them. Finally, Section V concludes this paper.

II. Preliminaries

In CF recommender systems, historical user behaviors are usually modeled into a use-item rating-matrix [1 -3]. Given an item set I and a user set U, the ratings-matrix R is a $|U|\times|I|$ matrix where each element $r_{u,i}$ is proportional to use u's preference on item i.

Let R_K and R_W denote the known and whole entry sets in R respectively. The CF problem is to construct an estimator $\hat{R}_{ui}$ based on R_K to prediction $\hat{r}_{ui}$ for each entry $(u,i)\in R_W$ and the error $\sum_{(u,i)\in R_W}\left|r_{u,i}-\hat{r}_{u,i}\right|$ is minimized.

Similar to the other LF-based models, NLF model [9, 17, 25] also factorizes the rating-matrix R into two rank-f matrices P and Q, where P is $|U|\times f$, Q is $f\times|I|$ and f<<min{$|U|$, $|I|$}. Note that P and Q reflect the user and item characteristics contained in the rating data. This factorization process is implemented by minimizing the objective function with Euclidean distance. Such an objective function is given by:

$$\begin{aligned}&\arg\min_{P,Q}\varepsilon(P,Q)\\&=\sum_{(u,i)\in R_K}\left(\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)^2+\lambda_P\sum_{m=1}^{f}p_{u,m}^2+\lambda_Q\sum_{m=1}^{f}q_{m,i}^2\right)\\&s.t.\quad P,Q\geq 0.\end{aligned}\tag{1}$$

where ||·|| denotes the Frobenius norm of a matrix. $p_{u,m}$ and $q_{m,i}$ denote the element of P and Q, and R_K denotes the known entry sets, respectively. The term behind λ_P and λ_Q is the Tikhonov regularizing term for avoiding over-fitting the generalized error.

Therefore, we adopt the additive gradient decent (AGD) to minimize the objective function (1). Note that in (1) only the known ratings and their corresponding estimates are considered. Therefore, an additive update rules for this training process is derived as follows:

$$\begin{aligned}&\arg\min_{P,Q}\varepsilon(P,Q)\overset{AGD}{\Rightarrow}\\&\begin{cases}p_{u,m}\leftarrow p_{u,m}-\sum_{i\in I_u}\eta_{u,m}\left(\lambda_P p_{u,m}-q_{m,i}\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)\right),\\q_{m,i}\leftarrow q_{m,i}-\sum_{u\in U_i}\eta_{m,i}\left(\lambda_Q q_{m,i}-p_{u,m}\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)\right).\end{cases}\end{aligned}\tag{2}$$

where I_u and U_i denote the item set rated by user u and user set having rated item i. Note that $\eta_{u,m}$ and $\eta_{m,i}$ in (2) are positive learning-rating-constants and the Euclidean distance objective function (1) is minimizing by moving with the opposite direction of the gradient. $\hat{r}_{u,i}=\sum_{m=1}^{f}p_{u,m}q_{m,i}$ denotes the estimate for $r_{u,i}$. Thus, (2) is reformulated to:

$$\arg\min_{P,Q}\varepsilon(P,Q)\Rightarrow$$
$$\begin{cases} p_{u,m}\leftarrow p_{u,m}+\eta_{u,m}\sum_{i\in I_u}q_{m,i}r_{u,i}-\eta_{u,m}\sum_{i\in I_u}\left(q_{m,i}\hat{r}_{u,i}+\lambda_P p_{u,m}\right), \\ q_{m,i}\leftarrow q_{m,i}+\eta_{m,i}\sum_{u\in U_i}p_{u,m}r_{u,i}-\eta_{m,i}\sum_{u\in U_i}\left(p_{u,m}\hat{r}_{u,i}+\lambda_Q q_{m,i}\right). \end{cases} \tag{3}$$

The negative terms in (3) are $-\eta_{u,m}\sum_{i\in I_u}\left(q_{m,i}\hat{r}_{u,i}+\lambda_P p_{u,m}\right)$ for $p_{u,m}$ and $-\eta_{m,i}\sum_{u\in U_i}\left(p_{u,m}\hat{r}_{u,i}+\lambda_Q q_{m,i}\right)$ for $q_{m,i}$. For cancelling these negative terms, we set $\eta_{u,m}=p_{u,m}\Big/\sum_{i\in I_u}\left(q_{m,i}\hat{r}_{u,i}+\lambda_P p_{u,m}\right)$ and $\eta_{m,i}=q_{m,i}\Big/\sum_{u\in U_i}\left(p_{u,m}\hat{r}_{u,i}+\lambda_Q q_{m,i}\right)$. Thus, the final SLF-NMU [7, 9] for $p_{u,m}$ and $q_{m,i}$ is given by:

$$\arg\min_{P,Q}\varepsilon(P,Q)\overset{SLF-NMU}{\Rightarrow}$$
$$\begin{cases} p_{u,m}\leftarrow p_{u,m}\left(\sum_{i\in I_u}q_{m,i}r_{u,i}\right)\Big/\left(\sum_{i\in I_u}q_{m,i}\hat{r}_{u,i}+|I_u|\lambda_P p_{u,m}\right), \\ q_{m,i}\leftarrow q_{m,i}\left(\sum_{u\in U_i}p_{u,m}r_{u,i}\right)\Big/\left(\sum_{u\in U_i}p_{u,m}\hat{r}_{u,i}+|U_i|\lambda_Q q_{m,i}\right). \end{cases} \tag{4}$$

III. NLF Models with *β*-Distance Functions

A. β-distance-based Objective Functions

Through careful investigations of the present research, there are various forms of β-distance-based objective functions [29, 30] which can also be applied to LF-based model in recommender systems. So we categorize β-distance-based objective functions into three common cases for the NLF model and the formulas as follows:

$$\begin{cases} \beta=0:\varepsilon_\beta(P,Q)=\sum_{r_{u,i}\in R_K}\left(\frac{r_{u,i}}{\hat{r}_{u,i}}-\log\frac{r_{u,i}}{\hat{r}_{u,i}}-1\right), \\ \beta=1:\varepsilon_\beta(P,Q)=\sum_{r_{u,i}\in R_K}\left(r_{u,i}\log\frac{r_{u,i}}{\hat{r}_{u,i}}-r_{u,i}+\hat{r}_{u,i}\right), \\ \beta\neq 0\text{ or }1:\varepsilon_\beta(P,Q)=\sum_{r_{u,i}\in R_K}\frac{\left(r_{u,i}^{\beta}+(\beta-1)\hat{r}_{u,i}^{\beta}-\beta r_{u,i}\hat{r}_{u,i}^{\beta-1}\right)}{\beta(\beta-1)}. \end{cases} \tag{5}$$

where log denotes the base-2 logarithm.

From (5) we see that β=1 for the Kullback-Leibler divergence and β=2 for the Euclidean distance. In the following section, we will deduce the updating rules of NLF model by integrating the Tikhonov regularizing terms based on different objective functions.

B. Case 1: β=0

When β=0, the objective function with the Tikhonov regularizing terms is derived as follows:

$$\arg\min_{P,Q}\varepsilon(P,Q)$$
$$=\sum_{(u,i)\in R_K}\left(\left(\frac{r_{u,i}}{\sum_{m=1}^{f}p_{u,m}q_{m,i}}-\log\frac{r_{u,i}}{\sum_{m=1}^{f}p_{u,m}q_{m,i}}-1\right)+\lambda_P\sum_{k=1}^{f}p_{u,m}^2+\lambda_Q\sum_{k=1}^{f}q_{m,i}^2\right), \tag{6}$$
$$s.t.\ \ P,Q\geq 0.$$

Similar to (2), the additive update rules for constrained optimization problem (6) without the nonnegative constraints by employing AGD is given by:

$$\arg\min_{P,Q}\varepsilon(P,Q)\overset{AGD}{\Rightarrow}$$
$$\begin{cases} p_{u,m}\leftarrow p_{u,m}-\sum_{i\in I_u}\eta_{u,m}\left(\frac{q_{m,i}}{\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)}+\lambda_P p_{u,m}-\frac{r_{u,i}q_{m,i}}{\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)^2}\right), \\ q_{m,i}\leftarrow q_{m,i}-\sum_{u\in U_i}\eta_{m,i}\left(\frac{p_{u,m}}{\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)}+\lambda_P q_{m,i}-\frac{r_{u,i}p_{u,m}}{\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)^2}\right). \end{cases} \tag{7}$$

Similar to (3), we reformulate (7) into

$$\arg\min_{P,Q}\varepsilon(P,Q)\overset{AGD}{\Rightarrow}$$
$$\begin{cases} p_{u,m}\leftarrow p_{u,m}+\eta_{u,m}\sum_{i\in I_u}\left(\frac{r_{u,i}q_{m,i}}{\hat{r}_{u,i}^2}\right)-\eta_{u,m}\sum_{i\in I_u}\left(\frac{q_{m,i}}{\hat{r}_{u,i}}+\lambda_P p_{u,m}\right), \\ q_{m,i}\leftarrow q_{m,i}+\eta_{m,i}\sum_{u\in U_i}\left(\frac{r_{u,i}p_{u,m}}{\hat{r}_{u,i}^2}\right)-\eta_{m,i}\sum_{u\in U_i}\left(\frac{p_{u,m}}{\hat{r}_{u,i}}+\lambda_P q_{m,i}\right). \end{cases} \tag{8}$$

In (8), the negative terms for $p_{u,m}$ and $q_{m,i}$ are $-\eta_{u,m}\sum_{i\in I_u}\left(\frac{q_{m,i}}{\hat{r}_{u,i}}+\lambda_P p_{u,m}\right)$, $-\eta_{m,i}\sum_{u\in U_i}\left(\frac{p_{u,m}}{\hat{r}_{u,i}}+\lambda_P q_{m,i}\right)$ respectively. So we set $\eta_{u,m}=\frac{p_{u,m}\hat{r}_{u,i}}{\sum_{i\in I_u}\left(q_{m,i}+\lambda_P p_{u,m}\hat{r}_{u,i}\right)}$ and $\eta_{m,i}=\frac{q_{m,i}\hat{r}_{u,i}}{\sum_{u\in U_i}\left(p_{u,m}+\lambda_P q_{m,i}\hat{r}_{u,i}\right)}$ to cancel the negative terms. After this adjustment, the SLF-NMU rules [22, 25] of $p_{u,m}$ and $q_{m,i}$ can be reformulated into:

$$\arg\min_{P,Q}\varepsilon(P,Q)\overset{SLF-NMU}{\Rightarrow}$$
$$\begin{cases} p_{u,m}\leftarrow p_{u,m}\sum_{i\in I_u}\frac{r_{u,i}q_{m,i}}{\hat{r}_{u,i}^{\ 2}}\Big/\left(\sum_{i\in I_u}\frac{q_{m,i}}{\hat{r}_{u,i}}+\lambda_P|I_u|p_{u,m}\right), \\ q_{m,i}\leftarrow q_{m,i}\sum_{u\in U_i}\frac{r_{u,i}p_{u,m}}{\hat{r}_{u,i}^{\ 2}}\Big/\left(\sum_{u\in U_i}\frac{p_{u,m}}{\hat{r}_{u,i}}+\lambda_Q|U_I|q_{m,i}\right). \end{cases} \tag{9}$$

C. Case 2: β=1

The second case is β=1 and we call it Kullback-Leibler divergence objective function. Similar to (1), this constrained optimization problem is changed into

$$
\begin{aligned}
&\underset{P,Q}{\arg\min}\,\varepsilon(P,Q)\\
&=\sum_{(u,i)\in R_K}\left(\left(r_{u,i}\log\frac{r_{u,i}}{\sum_{m=1}^{f}p_{u,m}q_{m,i}}-r_{u,i}+\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)+\lambda_P\sum_{m=1}^{f}p_{u,m}^2+\lambda_Q\sum_{m=1}^{f}q_{m,i}^2\right),\quad(10)\\
&s.t.\ \ P,Q\geq 0.
\end{aligned}
$$

By applying AGD to (10) we obtain:

$$
\underset{P,Q}{\arg\min}\,\varepsilon(P,Q)\overset{AGD}{\Rightarrow}
\begin{cases}
p_{u,m}\leftarrow p_{u,m}-\sum_{i\in I_u}\eta_{u,m}\left(-q_{m,i}\left(\dfrac{r_{u,i}}{\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)}-1\right)+\lambda_P p_{u,m}\right),\\
q_{m,i}\leftarrow q_{m,i}-\sum_{u\in U_i}\eta_{m,i}\left(-p_{u,m}\left(\dfrac{r_{u,i}}{\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)}-1\right)+\lambda_Q q_{m,i}\right).
\end{cases}
\quad(11)
$$

Similar to (3), we reformulate (11) into

$$
\underset{P,Q}{\arg\min}\,\varepsilon(P,Q)\overset{AGD}{\Rightarrow}
\begin{cases}
p_{u,m}\leftarrow p_{u,m}+\eta_{u,m}\sum_{i\in I_u}\left(\dfrac{q_{m,i}r_{u,i}}{\hat{r}_{u,i}}\right)-\eta_{u,m}\sum_{i\in I_u}\left(q_{m,i}+\lambda_P p_{u,m}\right),\\
q_{m,i}\leftarrow q_{m,i}+\eta_{m,i}\sum_{u\in U_i}\left(\dfrac{p_{u,m}r_{u,i}}{\hat{r}_{u,i}}\right)-\eta_{m,i}\sum_{u\in U_i}\left(p_{u,m}+\lambda_Q q_{m,i}\right).
\end{cases}
\quad(12)
$$

Note that in (12), the negative terms for $p_{u,m}$ and $q_{m,i}$ are $-\eta_{u,m}\sum_{i\in I_u}\left(q_{m,i}+\lambda_P p_{u,m}\right)$, $-\eta_{m,i}\sum_{u\in U_i}\left(p_{u,m}+\lambda_Q q_{m,i}\right)$ respectively. With $\eta_{u,m}=\frac{p_{u,m}}{\sum_{i\in I_u}\left(q_{m,i}+\lambda_P p_{u,m}\right)}$ and $\eta_{m,i}=\frac{q_{m,i}}{\sum_{i\in I_u}\left(p_{u,m}+\lambda_Q q_{m,i}\right)}$, We cancel these negative terms to obtain:

$$
\underset{P,Q}{\arg\min}\,\varepsilon(P,Q)\overset{SLF-NMU}{\Rightarrow}
\begin{cases}
p_{u,m}\leftarrow p_{u,m}\sum_{i\in I_u}\dfrac{r_{u,i}q_{m,i}}{\hat{r}_{u,i}}\Bigg/\left(\sum_{i\in I_u}q_{m,i}+\lambda_P\left|I_u\right|p_{u,m}\right),\\
q_{m,i}\leftarrow q_{m,i}\sum_{u\in U_i}\dfrac{r_{u,i}p_{u,m}}{\hat{r}_{u,i}}\Bigg/\left(\sum_{u\in U_i}p_{u,m}+\lambda_Q\left|U_I\right|q_{m,i}\right).
\end{cases}
\quad(13)
$$

D. Case 3: β ≠0 or 1

The last case is β≠0 and 1, the derivation process is the same as before. Therefore, after integrating regularizing terms, the objective of NLF model is formulated into

$$
\begin{aligned}
&\underset{P,Q}{\arg\min}\,\varepsilon(P,Q)\\
&=\sum_{(u,i)\in R_K}\left(\begin{array}{l}\dfrac{r_{u,i}{}^{\beta}}{\beta(\beta-1)}+\dfrac{\left(\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)^{\beta}}{\beta}-\dfrac{r_{u,i}\left(\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)^{\beta-1}}{\beta-1}\\+\lambda_P\sum_{m=1}^{f}p_{u,m}^2+\lambda_Q\sum_{m=1}^{f}q_{m,i}^2\end{array}\right),\quad(14)\\
&s.t.\ \ P,Q\geq 0.
\end{aligned}
$$

By applying AGD to (14) we obtain:

$$
\underset{P,Q}{\arg\min}\,\varepsilon(P,Q)\overset{AGD}{\Rightarrow}
\begin{cases}
p_{u,m}\leftarrow p_{u,m}-\sum_{i\in I_u}\eta_{u,m}\left(\lambda_P p_{u,m}-q_{m,i}\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)\left(\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)^{\beta-2}\right)\cdot\\
q_{m,i}\leftarrow q_{m,i}-\sum_{u\in U_i}\eta_{m,i}\left(\lambda_Q q_{m,i}-p_{u,m}\left(r_{u,i}-\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)\left(\sum_{m=1}^{f}p_{u,m}q_{m,i}\right)^{\beta-2}\right).
\end{cases}
\quad(15)
$$

Similar to (3), we reformulate (15) into

$$
\underset{P,Q}{\arg\min}\,\varepsilon(P,Q)\overset{AGD}{\Rightarrow}
\Rightarrow\begin{cases}
p_{u,m}\leftarrow p_{u,m}+\eta_{u,m}\sum_{i\in I_u}r_{u,i}\hat{r}_{u,i}^{\beta-2}q_{m,i}-\eta_{u,m}\sum_{i\in I_u}\left(\hat{r}_{u,i}^{\beta-1}q_{m,i}+\lambda_P p_{u,m}\right),\\
q_{m,i}\leftarrow q_{m,i}+\eta_{m,i}\sum_{u\in U_i}r_{u,i}\hat{r}_{u,i}^{\beta-2}p_{u,m}-\eta_{m,i}\sum_{u\in U_i}\left(\hat{r}_{u,i}^{\beta-1}p_{u,m}+\lambda_Q q_{m,i}\right).
\end{cases}
\quad(16)
$$

From (16), we observe that the negative components during such additive updates include $-\eta_{u,m}\sum_{i\in I_u}\left(\hat{r}_{u,i}^{\beta-1}q_{m,i}+\lambda_P p_{u,m}\right)$ for $p_{u,m}$ and $-\eta_{m,i}\sum_{u\in U_i}\left(\hat{r}_{u,i}^{\beta-1}p_{u,m}+\lambda_Q q_{m,i}\right)$ for $q_{m,i}$. Therefore, following the principle of SLF-NMU [22, 25], we cancel them to derive a nonnegative training process. In particular, we manipulate learning rates as follows:

$$
\begin{cases}
\eta_{u,m}=\dfrac{p_{u,m}}{\sum_{i\in I_u}\left(\hat{r}_{u,i}^{\beta-1}q_{m,i}+\lambda_P p_{u,m}\right)},\\
\eta_{m,i}=\dfrac{q_{m,i}}{\sum_{u\in U_i}\left(\hat{r}_{u,i}^{\beta-1}p_{u,m}+\lambda_Q q_{m,i}\right)}.
\end{cases}
\quad(17)
$$

By substituting (17) in (16), we obtain the SLF-NMU rules [22, 25] as follows:

$$
\underset{P,Q}{\arg\min}\,\varepsilon(P,Q)\overset{SLF-NMU}{\Rightarrow}
\begin{cases}
p_{u,m}\leftarrow p_{u,m}\dfrac{\sum_{i\in I_u}r_{u,i}\hat{r}_{u,i}^{\beta-2}q_{m,i}}{\sum_{i\in I_u}\hat{r}_{u,i}^{\beta-1}q_{m,i}+\left|I_u\right|\lambda_P p_{u,m}},\\
q_{m,i}\leftarrow q_{m,i}\dfrac{\sum_{u\in U_i}r_{u,i}\hat{r}_{u,i}^{\beta-2}p_{u,m}}{\sum_{u\in U_i}\hat{r}_{u,i}^{\beta-1}p_{u,m}+\left|U_i\right|\lambda_Q q_{m,i}}.
\end{cases}
\quad(18)
$$

When β=2, (18) turn into the Euclidean distance objective function. It is the same as (3) in Section II.

Based on the above analysis, we obtain a series of NLF models with different β-distance-based objective functions, as given in (9), (13) and (18).

IV. Experiments and Results

A. General Settings

1) Evaluation Metrics: Recommenders can be evaluated from several aspects, like the prediction accuracy and prediction coverage. In this work, we mainly care about the predictions accuracy to the actual entries with different β values of NLF model. Hence, we choose the root mean squared error (RMSE) and mean absolute error (MAE) [36, 37] as the accuracy metrics. RMSE is a widely used metric for evaluating the statistical accuracy of recommenders, which is given below

$$RMSE = \sqrt{\left(\sum_{(u,i)\in R_{test}} \left(r_{u,i} - \hat{r}_{u,i} \right)^2 \right) \Big/ R_{test}} . \tag{19}$$

where R_{test} denotes the validation set. For a given recommender, low RMSE stands for high prediction accuracy. Similar to RMSE, the MAE is a frequently employed metric measuring the absolute difference between the target dataset and the prediction set as follows:

$$MAE = \left(\sum_{(u,i)\in R_{test}} \left| r_{u,i} - \hat{r}_{u,i} \right|_{abs} \right) \Big/ R_{test} . \tag{20}$$

where $|\cdot|_{abs}$ denotes the absolute value of a given number. All experiments are conducted on a PC with an Intel Xeon E5-2640 CPU and 256G RAM. In terms of efficiency, we care about the convergence rate. Thus, we have recorded the necessary number of iterations to make the model converge.

2) Data Set: The experiments were conducted on two HiDS matrices collected by industrial applications which are well-know and widely used in in the recommender systems. Their general descriptions are given below:

a) D1: Douban dataset. It is collected from Douban.com. It contains 16,830,839 ratings in the scale of [0, 5] from 129,490 users on 58,541 movies. Its rating density is 0.222%.

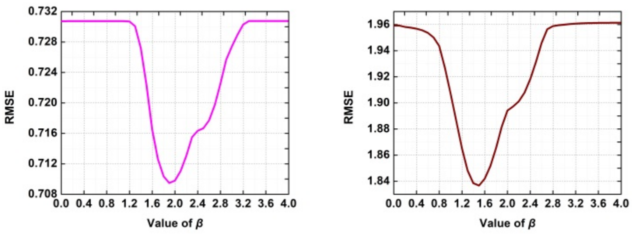
b) D2: A subset of the Dating Agency dataset. The Dating Agency dataset is collected through the online dating recommender by Charles University. D4 consists of 17,359,346 ratings in the scale of [1, 10] by 135,359 LibimSeTi users on 168,791 profiles. D4 has the rating density of 0.075% only.

3) Modeling Setting: Note that our experiments are designed for validating the performance of NLF model with different β-distance-based objective functions. Hence, the Tikhonov regularizing term $\lambda_P = \lambda_Q$ are both set as 0.05. The initial training epochs is set as 1000 for each β value. The dimension of the latent space f is set at 20 uniformly. Meanwhile, to minimize the impact caused by random initial guesses, the latent factor P and Q are also initialized with the same randomly-generated arrays (Note that with different objective functions, the initial guess of two NLF models cannot be exactly the same). Therefore, we have applied five-fold cross validation to each objective function on each data set to ensure the objectiveness of our experiments.

B. Results and Discussion

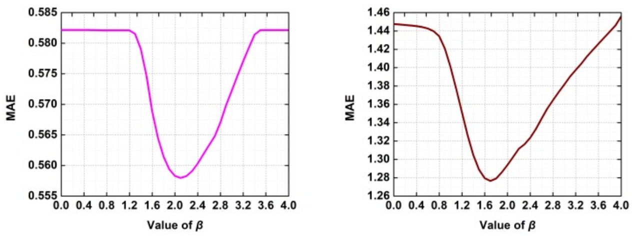
The comparison in RMSE and MAE of different β is depicted in Fig. 1 and Fig. 2. They are all tested on two datasets mentioned above. The minimum RMSE and MAE corresponding value β are recorded in Table I and Table II. The minimum training epochs are showed in Table III and Table IV. From these results, we have the following findings.

1) From Fig. 1 and Fig. 2, we see that NLF models measured by RMSE and MAE with different β are very similar on D1 and D2. The curve is the same as a parabola going upwards. The RMSE of Euclidean distance (β=2) objective function is 0.7098 on D1. From Table I, the minimum RMSE is 0.7094 when β=1.9 on D1 which is 0.06% lower than that achieved by β=2. On D2, NLF model can achieve the minimum RMSE at 1.8367 by β=1.5 and the RMSE of β=2 is 1.8941 which is 3.03% higher. The training epochs are all 1000 when NLF models achieve the optimum RMSE. NLF model always gets the poor performance when β=1.



(a) D1 (b) D2

Fig. 1. RMSE of NLF models with different β.



(a) D1 (b) D2

Fig. 2. MAE of NLF models with different β.

2) From Fig. 2, when employing MAE as the evaluation metric, the situation is similar with RMSE. As same as the situation of RMSE, β=2 is still not achieving the minimum MAE. As shown in Table II, the minimum MAE on two datasets is 0.5579 and 1.2767 respectively corresponding to value β is 2.1 and 1.7. Compared Table I with Table II, the β that corresponding to the minimum prediction accuracy all move to the right. That means β increases. For instance, on D1, getting the minimum RMSE when β=1.9 and getting the minimum MAE when β=2.1. All the results indicate that β=2 or β=1 is not the best choices for achieving the most accurate NLF model.

3) For a detailed comparison, we get the training epochs of different β in Fig. 3 and Fig. 4. By combining Figs. 3 and Fig.

4 along with Table III and Table IV, we can see that NLF models tend to consume more epochs to converge as β increases. NLF models always reach the best performance when training epochs is 1000. The epochs is always 1000 starting from β=1.2 on D1 and β=0.5 on D2. Table III and Table IV show that β=0 always lead to the minimum training epochs and poor prediction accuracy.

TABLE I. MINIMUM RMSE AND TRAINING EPOCHS

Dataset	Minimum RMSE/training epochs	Value β
D1	0.7094/1000	1.9
D2	1.8367/1000	1.5

TABLE II. MINIMUM MAE AND TRAINING EPOCHS

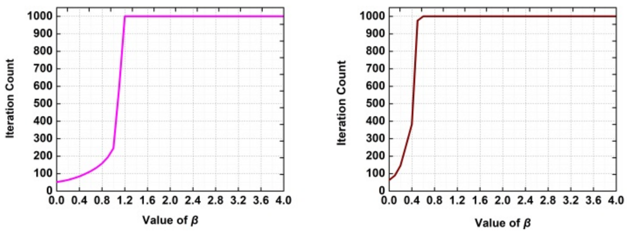
Dataset	Minimum MAE/training epochs	Value β
D1	0.5579/1000	2.1
D2	1.2767/1000	1.7

TABLE III. MINIMUM TRAINING EPOCHS AND RMSE

Dataset	Minimum training epochs/RMSE	Value β
D1	53/0.7305	0
D2	72/1.9601	0

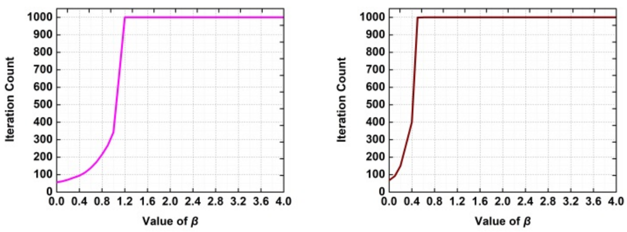
TABLE IV. MINIMUM TRAINING EPOCHS AND MAE

Dataset	Minimum training epochs/MAE	Value β
D1	61/0.5821	0
D2	80/1.4489	0



(a) D1 (b) D2

Fig. 3. Training epoch count of NLF models with different β and RMSE.



(a) D1 (b) D2

Fig. 4. Training epoch count of NLF models with different β and MAE.

Based on the experimental results, we conclude that β=1 or β=2 is not the best choices for achieving the most accurate nor efficient NLF models. Values β in MAE that corresponding to the minimum prediction accuracy all move to the right compared with RMSE. NLF models always reach the best performance when training epochs is 1000. β=0 always lead to the minimum training epochs.

V. CONCLUSIONS

This paper focuses on developing and validating NLF models with different β-distance-based objective functions. To do so, we first classify β-distance functions into three cases, i.e., β=0, β=1, β≠0 and 1. For each case, we define the objective function of an NLF model accordingly, and then deduce the updating rules following the principle of single latent factor-dependent multiplicative update. For β=0 and β=1, the update rule is more specific. For β≠0 and 1, we have achieved a general mathematic form of the update rules. Experimental results on two large, real datasets generated by industrial applications well demonstrate that as β varies, the resultant NLF model can achieve significantly higher prediction accuracy than the models based on the special cases of β=2 or β=1 do, which are commonly adopted by prior research. Therefore, the Euclidean distance (β=2) or Kullback-Leibler divergence (β=1) are probably not the best choice for an NLF model as the loss function.

From the experimental results, we see that the optimal value of β is data dependent. Currently, we suggest to perform cross-validation with respect to the value of β on probe dataset in real application. However, to design selection strategies for β and testing the influence on other efficient model [38] are highly demanded and included in our future plan.

REFERENCES

[1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recom-mender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, Jun. 2005.

[2] G. Adomavicius and Y. Kwon, "Improving aggregate recommendation diversity using ranking-based techniques," *IEEE Trans. on Knowledge and Data Engineering*, vol. 24, no. 5, pp. 896–911, May. 2012.

[3] Y. Koren and R. Bell, "Advances in collaborative-filtering," *in Recommender Systems Handbook*, NY, USA, pp. 145–186, 2015.

[4] X. Luo, M. C. Zhou, Y. N. Xia, et al, "An Efficient Non-Negative Matrix-Factorization-Based Approach to Collaborative Filtering for Recommender Systems[J]," *IEEE Trans. on Industrial Informatics*, vol. 10, no. 2, pp. 1273-1284, Feb. 2014.

[5] J. de la Rosa, N. Hormazabal, S. Aciar, G. Lopardo, A. Trias, and M. Montaner, "A negotiation-style recommender based on computational ecology in open negotiation environments," *IEEE Trans. on Industrial Electronics*, vol. 58, no. 6, pp. 2073–2085, Jun. 2011.

[6] P. Resnick and H. R. Varian, "Recommender systems," *Communications of the ACM*, vol. 40, no. 3, pp. 56–58, 1997.

[7] X. Luo, M. C. Zhou, S. Li, et al, "A Nonnegative Latent Factor Model for Large-Scale Sparse Matrices in Recommender Systems via Alternating Direction Method[J],". *IEEE Trans. on Neural Networks and Learning Systems*, vol. 27, no. 3, pp. 579-592, May. 2015.

[8] F. Dinuzzo, G. Pillonetto, and G. De Nicolao, "Client–server multitask learning from distributed datasets," *IEEE Trans. on Neural Networks*, vol. 22, no. 2, pp. 290–303, Feb. 2011.

[9] A. Friedman, S. Berkovsky, and M. A. Kaafar, "A differential privacy framework for matrix factorization recommender systems[J]," *User Modeling and User-Adapted Interaction*, vol. 26, no. 5, pp. 1-34, Dec. 2016.

[10] X. Luo, M. C. Zhou, Y. N. Xia, and Q. S. Zhu, "An Incremental-and-Static-Combined Scheme for Matrix-Factorization-Based Collaborative Filtering[J]," *IEEE Trans. on Automation Science and Engineering*, vol. 13, no. 1, pp. 333-343, Aug. 2016.

[11] A. Hernando, J. Bobadilla, and F. Ortega, "A non negative matrix factorization for collaborative filtering recommender systems based on a

Bayesian probabilistic model[J]," *Knowledge-Based Systems*, vol. 97, no. C, pp. 188-202, Apr. 2016.

[12] X. Luo, M. C. Zhou, Y. N. Xia, Q. S. Zhu, A. C. Ammari, and A. Alabdulwahab, "Generating Highly Accurate Predictions for Missing QoS-data via Aggregating Non-negative Latent Factor Models," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 27, no. 3, pp. 524-537, Apr. 2015.

[13] G. Chen, F. Wang, and C. Zhang, "Collaborative filtering using orthogonal nonnegative matrix tri-factorization," *Information Processing and Management*, vol. 45, no. 3, pp. 368–379, May. 2009.

[14] X. Luo, M. C. Zhou, Z. D. Wang, Y. N. Xia, and Q. S. Zhu, "An Effective QoS Estimating Scheme via Alternating Direction Method-based Matrix Factorization," *IEEE Trans. on Services Computing*, vol. pp, no. 99, pp. 1-1, Aug. 2016.

[15] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," *in Proc. of 10th Int. Conf. on World Wide Web*, Hong Kong, pp. 285–295, May 2001.

[16] S. Zhang, W. Wang, J. Ford, and F.Makedon, "Learning from incomplete ratings using non-negative matrix factorization," *in Proc.of 6th SIAM Int. Conf. on Data Mining*, Bethesda, MD, USA, pp. 549–553, Apr. 2006.

[17] X. Luo and M. S. Shang, "Symmetric Non-negative Latent Factor Models for Undirected Large Networks,". In *Proc. of the 27th Int. Joint Conf. on Artificial Intelligence*, pp. 2435-2442, 2017.

[18] X. Luo, J. P. Sun, Z. D. Wang, S. Li, and M. S. Shang, "Symmetric and Non-negative Latent Factor Models for Undirected, High Dimensional and Sparse Networks in Industrial Applications," *IEEE Trans. on Industrial Informatics*, vol. 13, no. 6, pp. 3098-3107, Jul. 2017.

[19] P. Paatero and U. Tapper, "Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values," *Environmetrics*, vol. 5, no. 2, pp. 111–126, Jun. 1994.

[20] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, Oct. 1999.

[21] X. Luo, M. C. Zhou, M. S. Shang, S. Li, and Y. N. Xia, "A Novel Approach to Extracting Non-negative Latent Factors from Big Sparse Matrices," *IEEE Access*, vol. 4, pp. 2649-2655, 2016.

[22] C. J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural Computation*, vol. 19, no. 10, pp. 2756–2779, Oct. 2007.

[23] C. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 45–55, Jan. 2010.

[24] N. Guan, D. Tao, Z. Luo, and B. Yuan, "Online nonnegative matrix factorization with robust stochastic approximation," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 23, no. 7, pp. 1087–1099, Jul. 2012.

[25] X. Luo, M. C. Zhou, S. Li, and M. S. Shang, "An Inherently Non-negative Latent Factor Model for High-dimensional and Sparse Matrices from Industrial Applications," *IEEE Trans. on Industrial Informatics*, vol. pp. no. 99. pp. 1-1, Oct. 2017.

[26] P. Greistorfer, A. Kketangen, Stefan, et al, "Experiments concerning sequential versus simultaneous maximization of objective function and distance[J]," *Journal of Heuristics*, vol. 14, no. 6, pp. 613-625, Dec. 2008.

[27] A. Løkketangen and D. L. Woodruff, "A distance function to support optimized selection decisions[J]," *Decision Support Systems*, vol. 39, no. 3, pp. 345-354, May. 2005.

[28] S. Li, L. Liu, X. Yin, et al, "Time Domain Objective Function Based on Euclidean Distance Matrix and its Application in Optimization of Short Pulse Power Divider[J]," *IEEE Microwave and Wireless Components Letters*, vol. 269, no. 1, pp. 4-8, Dec. 2015.

[29] T. Liu, M. Gong, and D. Tao, "Large-Cone Nonnegative Matrix Factorization[J]," *IEEE Trans. on Neural Networks and Learning Systems*, vol. pp, no. 99, pp. 1-14, Jun. 2016.

[30] R. Sandler and M. Lindenbaum, "Nonnegative Matrix Factorization with Earth Mover's Distance Metric for Image Analysis[J]," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1590-1602, 2011, Jan. 2011.

[31] P. Zhang, Y. Tian, Z. Zhang, et al, "Select Objective Functions for Multiple Criteria Programming Classification[C]," *in proc. of IEEE/WIC/ACM Int. Conf. on Web Intelligence and Intelligent Agent Technology*, vol. 3, pp. 420-423, Dec. 2008.

[32] B. Zhu, J. F. Gao, M. Nakagawa, "Objective Function Design for MCE-Based Combination of On-line and Off-line Character Recognizers for On-line Handwritten Japanese Text Recognition[C]," In *Proc. of the Int. Conf. on Document Analysis and Recognition*, pp. 594-598, Sep. 2011.

[33] G. Gorrell, "Generalized Hebbian algorithm for incremental singular value decomposition in natural language processing," *in Proc. of 11th Conf. Eur. Chapter Assoc. Comput. Linguist*, Italy, pp. 97–104, Jan. 2006.

[34] F. Y. Shih and C. C. Pu, "A maxima-tracking method for skeletonization from Euclidean distance function[C]," *in proc. of Int. Conf. on TOOLS for Artificial Intelligence*, pp. 246-253, Nov. 1991.

[35] L. Vincent, "Exact Euclidean distance function by chain propagations[C]," *in proc. of IEEE Computer Society Conf. on Computer Vision and Pattern*, pp. 520-525, Jun. 1991.

[36] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Trans. on Information Systems*, vol. 22, no. 1, pp. 5–53, Jan. 2004.

[37] G. Shani and A. Gunawardana, "Evaluating recommendation systems," *in Recommender Systems Handbook*, New York, NY, USA: Springer-Verlag, pp. 257-297, Oct. 2011.

[38] X. Luo, M. C. Zhou, S. Li, et al, "An Efficient Second-order Approach to Factorizing Sparse Matrices in Recommender Systems," *IEEE Trans. on Industrial Informatics*, vol. 11, no. 4, pp. 946-956, Aug. 2015..