一种改进的基于二部图网络结构的推荐算法

王 茜, 段双艳
(重庆大学 计算机学院, 重庆 400044)

摘 要: 基于网络结构的推荐算法得到了研究者越来越多的关注,以往的基于二部图网络结构的推荐算法只是判断用户是否选择过项目,不区分用户对项目评分的高低。这些算法倾向于推荐流行商品,没有考虑项目度和权值的影响。针对这些问题,在区分高低分的情况下提出了改进的基于加权网络结构的推荐算法。算法在计算用户间的相似性系数时,引入项目度与项目的权值之和的比值 θ,以提高推荐多样性。实验结果表明,改进后的算法能够提高推荐准确性和多样性,并且降低了推荐项目的流行性。

关键词: 个性化推荐; 加权二部图网络; 项目度; 准确性; 多样性
中图分类号: TP311;TP301.6 **文献标志码**: A **文章编号**: 1001-3695(2013)03-0771-04
doi:10.3969/j.issn.1001-3695.2013.03.033

Improved recommendation algorithm based on bipartite networks

WANG Qian, DUAN Shuang-yan
(*College of Computer Science, Chongqing University, Chongqing* 400044, *China*)

Abstract: In recent years, the recommendation algorithm based on networks has been attracting more and more researchers' attention. However, these recommendation algorithms based on bipartite networks are only to judge whether the user has selected the objects instead of distinguishing the preferences of the user about the object. And these algorithms tend to recommend popular objects, without considering the influence of object degree and the weights of the object. To solve these problems, this paper proposed an improved recommendation algorithm based on weighted networks, which distinguished the level of rating that a user voted an object. At the same time, the ratio θ of the object degree and the sum of weights of the object were embedded into the similarity index between users to improve the recommendation diversity. Experimental results show that the improved algorithm can improve recommendation accuracy and diversity, while reducing the epidemic of the recommended objects.

Key words: recommendation system; weighted bipartite networks; object degree; accuracy; diversity

0 引言

随着互联网技术的飞速发展,网络中的信息量急剧上升。然而,这一方面带来了信息超载的问题,即过量信息同时呈现使得用户无法从中获得自己感兴趣和对自己有用的部分,这样信息使用效率反而降低;另一方面也使得大量少人问津的信息成为网络中的暗信息[1],无法被用户获取。如何帮助用户在海量的数据中快速找到对其有价值的信息,并让网络中的暗信息能够被用户获取成为亟待解决的问题。个性化的推荐系统应运而生,它是解决这些问题非常有潜力的方法。推荐系统是指利用电子商务网站向客户提供商品信息和建议,帮助用户决定应该购买什么产品,模拟销售人员帮助客户完成购买过程[2,3]。

推荐算法是个性化推荐系统的核心[1],现有的推荐算法有协同过滤推荐算法[4]、基于内容的推荐算法、组合推荐算法以及最近兴起的基于用户—产品二部图网络结构的推荐算法[5~10]等。协同过滤算法是通过计算用户之间的相似性,寻找目标用户的最近邻居,然后通过最近邻居预测目标用户对未评分项目的评分,最后产生推荐。基于内容的推荐算法是给目标用户推荐与其过去喜欢的商品类似的商品。

最近,基于用户—项目二部图网络结构的推荐算法得到了研究者的关注。基于网络结构的推荐算法就是利用二部图上的物质扩散[5,6]、热传导[7]等复杂网络动力学过程来对用户进行个性化推荐。基于网络结构的推荐算法不但在算法推荐准确性上优于经典的协同过滤算法,而且在算法复杂性上也明显低于经典的协同过滤算法。

目前很多推荐算法都倾向于向用户推荐流行商品[11],即推荐系统中的明信息。虽然推荐流行商品能提高推荐准确性,但是,这些流行商品用户可以通过其他途径了解到。所以在保证准确性的同时,给用户推荐冷门商品(推荐系统中的暗信息)比推荐流行商品更具有实际意义[12]。

对于评价推荐系统的优劣,长期以来主要采用准确性指标。但是,个性化推荐还有一个重要的特点就是给不同的用户推荐不同的商品。所以衡量推荐系统的优劣还要看系统的多样性[5,11,12],即个性化程度。

本文中,不仅判断用户是否选择过项目,还要区分用户对项目评分的高低,以提高推荐准确性。同时,为了使冷门商品也能得到推荐,在计算用户相似性系数时,综合考虑用户项目

收稿日期: 2012-07-17; **修回日期**: 2012-08-27

作者简介: 王茜(1964-),女,重庆人,教授,博士,主要研究方向为信息安全、电子商务、远程教育课件工具;段双艳(1987-),女,硕士研究生,主要研究方向为电子商务、个性化推荐(duanshuangyan@gmail.com).

度和评分的影响,以提高算法的个性化程度。分析实验结果得出,提出的改进算法能够提高算法的推荐质量。

1 基于网络结构的推荐算法

一个由 n 个用户和 m 个项目组成的推荐系统,如果用户 u_i 购买、选择过项目 o_j,则用户 u_i 和项目 o_j 就被一条边连接。这就构成了用户—项目二部图 G[5,6]。定义用户集合 $U=\{u_1,u_2,u_3,\cdots,u_n\}$,项目集合 $O=\{o_1,o_2,o_3,\cdots,o_m\}$,因此可以用 $(n+m)$ 个节点表示整个系统。推荐系统能够描述为一个邻接矩阵 $\boldsymbol{A}=\{a_{ij}\}\in \mathrm{R}^{n,m}$。$a_{ij}=1$ 表示用户 u_i 购买或选择过项目 o_j,否则,$a_{ij}=0$。

受复杂网络物质扩散的启发,Zhou 等人[5] 和 Liu 等人[6] 提出用物质扩散的方法计算用户之间的相似性。假设给每个用户分配一个单位的资源(推荐能量),然后将这一单位的资源等分地分给其选择的项目,每个项目再把其收到的资源分给选择它的用户。假设任意两个用户 u_α 和 u_β 之间存在某种数量的物质,权重 $S_{\alpha\beta}$ 表示用户 u_β 可能贡献给用户 u_α 的物质的比例,即两个用户的相似性系数,如式(1)所示。

$$S_{\alpha\beta}=\frac{1}{k(u_\beta)}\sum_{i=1}^{m}\frac{a_{\alpha i}a_{\beta i}}{k(o_i)} \tag{1}$$

其中:$k(u_\beta)=\sum_{i=1}^{m}a_{\beta i}$ 表示用户 u_β 的度(该用户选择过多少项目);$k(o_i)=\sum_{l=1}^{n}a_{li}$ 表示项目 o_i 的度(该项目被多少用户选择过)。

对于目标用户 u_α 未评分的项目 o_i,可以通过式(2)预测用户对该项目的评分 $v_{\alpha i}$,这个分数表示用户对项目的喜欢程度,分数越高表示用户越喜欢该项目。

$$v_{\alpha i}=\frac{\sum_{\beta=1,\beta\neq\alpha}^{n}S_{\alpha\beta}\times a_{\beta_i}}{\sum_{\beta=1,\beta\neq\alpha}^{n}S_{\alpha\beta}} \tag{2}$$

对于给定的目标用户 u_α,推荐算法可以根据用户的历史信息给用户一个其未选择过的项目的排序列表,然后将前 N 项推荐给目标用户。基于网络结构的推荐算法得到推荐列表的步骤如下:

a)计算用户间的相似性 S。

b)对目标用户 u_α 根据式(2)预测其没有选择过的项目 o_i 的评分值 $v_{\alpha i}$。

c)将目标用户 u_α 没有选择过的项目的评分值进行降序排列,推荐 top-N(前 N 项)给目标用户。

2 改进的基于网络结构的推荐算法

2.1 区分高低分

许多推荐系统都允许用户对购买、浏览过的项目进行评分,所以推荐系统也可以看成是一个评分系统。比如 Yahoo 的音乐推荐系统,允许用户对音乐评 5 个星级。评分越高说明用户越喜欢该音乐。在协同过滤算法中[4,6],利用 Pearson 相关系数计算用户间的相似性,这种方法认为低分在推荐系统中起负面作用。前面的基于网络结构的推荐算法只是判断用户是否选择过项目,不区分高低评分对推荐结果的影响。然而用户的评分一定程度上表示了用户对项目的喜好程度[13],不区分高低分可能会造成信息的损失。有些推荐算法直接忽略低分,在 5 分的推荐系统中,当且仅当用户对项目的评分大于等于 3 才认为用户选择过项目。但是用户对项目评分时有可能受当时环境、心情等因素的影响,而且低分同样包含着一些信息,如用户关注过该项目,所以不应该直接去掉低分。

在本文的改进算法中要区分高分和低分,即用户对项目的喜好。对每条用户—项目的边赋一个权值 ω,用户 u_i 选择过项目 o_j,且评分大于等于 3(用户喜欢该项目),则 $\omega_{ij}=1$;用户 u_i 选择过项目 o_j 但评分小于 3(用户不喜欢该项目),则 $\omega_{ij}=\lambda$,λ 为可调参数,取值在 0 和 1 之间,用于调节低分对推荐结果的影响;用户 u_i 未选择过项目 o_j 则 $\omega_{ij}=0$。当 $\lambda=1$ 时,退化到只判断用户是否购买过项目的情况。

区分高低分时,计算用户间相似性的公式如下:

$$S_{\alpha\beta}=\frac{1}{d(u_\beta)}\sum_{i=1}^{m}\frac{\omega_{\alpha i}\omega_{\beta i}}{d(o_i)} \tag{3}$$

其中:$\omega=1,\lambda$ 或 0;$d(u_\beta)=\sum_{i=1}^{m}\omega_{\beta i}$ 表示用户 u_β 的所有连边的权值之和;$d(o_i)=\sum_{l=1}^{n}\omega_{li}$ 表示项目 o_i 的所有连边的权值之和。

计算目标用户 u_α 对未评分项目 o_i 的评分 $v_{\alpha i}$ 的公式如下:

$$v_{\alpha i}=\frac{\sum_{\beta=1,\beta\neq\alpha}^{n}S_{\alpha\beta}\times\omega_{\beta i}}{\sum_{\beta=1,\beta\neq\alpha}^{n}S_{\alpha\beta}} \tag{4}$$

2.2 考虑项目的度和项目的权值之和的影响

大多数推荐算法都倾向于给用户推荐流行商品,但是给用户推荐冷门商品比推荐流行商品更有意义。如果两个用户同时选择一个大度项目(流行商品),这并不能意味着两个用户的兴趣是相似的;相反,如果两个用户同时选择一个小度的项目,这就意味着他们有相似的独特的兴趣。算法中应该考虑项目的度,增强小度项目的推荐能力,相反,降低大度项目的推荐能力,以提高多样性。

同时,如果两个用户同时选择两个项目 o_i 和 o_j,且两个项目度相同,但是项目 o_i 的评分基本大于等于 3,即项目的权值之和相对较高,说明选择过它的用户基本都喜欢它;相反,项目 o_j 的评分基本都小于 3,项目的权值之和相对较低,那么应该减弱这种选择过它的用户基本都不喜欢的项目的推荐能力。在改进的算法中,计算用户相似性系数时,考虑项目度和项目的权值之和的比值 θ,即 $\theta(o_i)=\frac{k(o_i)}{d(o_i)}$,$k(o_i)$ 表示项目 o_i 的度。当选择过该项目的所有用户都不喜欢该项目时,θ 取最大值 $1/\lambda$;当选择过该项目的所有用户都喜欢该项目时,θ 取最小值 1。即 θ 越大,用户越不喜欢该项目。算法中应该降低这种项目的推荐能力。计算用户间相似性系数的最终公式如式(5)所示。

$$S_{\alpha\beta}=\frac{1}{d(u_\beta)}\sum_{i=1}^{m}\frac{\omega_{\alpha i}\omega_{\beta i}}{d^{f(\theta)}(o_i)} \tag{5}$$

其中:$d(u_\beta)=\sum_{i=1}^{m}\omega_{\beta i}$,$d(o_i)=\sum_{l=1}^{n}\omega_{li}$;$\theta=\frac{k(o_i)}{d(o_i)}$ 为项目的度和项目的权值之和的比值,$k(o_i)$ 为项目 o_i 的度,则 θ 取值为 $(1,1/\lambda)$。如果直接让 $f(\theta)=\delta+\theta$,其中 δ 为可调参数。虽然这也满足 θ 与 $S_{\alpha\beta}$ 的关系,但是这样会对 $S_{\alpha\beta}$ 产生过大的影响,对推荐准确性产生过大影响。由于算法在提高多样性的同时要保证准确性,所以要减小 θ 对 $S_{\alpha\beta}$ 的影响,通过多次实验总结比较发现 θ 以函数 $f(x)=\delta+\mathrm{e}^{\lambda(x-1)}$ 的形式加入,可以减小 θ 对 $S_{\alpha\beta}$ 的影响,提高推荐准确性和多样性。所以式(5)中的

$f(\theta)$的定义为

$$f(\theta)=\delta+e^{\lambda(\theta-1)} \tag{6}$$

其中:θ 的取值范围为$(1,\ 1/\lambda)$;δ 为一个可调参数,用于调节项目度对推荐的影响。当 $\delta>0$ 时表示大度项目的推荐能力被压制;当 $\delta<0$ 时表示大度项目的推荐能力得到提高。

3　算法的详细步骤

改进的基于加权二部图网络结构的推荐算法的详细步骤如下:

输入:用户和项目的评分矩阵 R,目标用户 u_α。

输出:目标用户 u_α 的推荐列表。

a)根据用户和项目的评分矩阵构造表示用户与项目关系的二部网络 G。其中边的权值 ω 通过用户对项目的评分确定。评分大于等于 3,则 $\omega=1$;评分小于 3,则 $\omega=\lambda$;用户未选择过项目则 $\omega=0$。

b)通过调节式(4)中的参数 λ 的值调节低分在推荐中的作用,确定 λ 的值得到更精确的相似性 $S_{\alpha\beta}$。

c)根据用户间的相似性 $S_{\alpha\beta}$ 和 θ(项目度与项目的权值和的比值)的关系确定函数 $f(\theta)$ 的具体形式,以得到更准确的相似度 $S_{\alpha\beta}$,保证推荐准确性并提高多样性;同时调节参数 δ 的值,以调节项目度对推荐质量的影响。

d)根据步骤 b)和 c)确定 λ、δ 的值和函数 $f(\theta)$ 的形式后,利用用户相似性计算公式(5)计算用户 u_α 和其他用户 u_β 之间的相似性 $S_{\alpha\beta}$。

e)利用式(4)计算目标用户 u_α 对未评分项目 o_i 的评分值 $v_{\alpha i}$。

f)将评分值进行降序排列,并把列表中的评分最高的 N 个项推荐给目标用户 u_α,完成推荐,算法结束。

下面对本文算法时间复杂度进行分析:推荐系统中有 n 个用户,m 个项目。步骤 b)和 c)主要是用来确定参数 λ、δ 的值和函数 $f(\theta)$ 的形式。由于这两步的计算过程可以离线进行,因此省略了在线计算的时间,提高了推荐效率;步骤 d)用来计算用户间的相似性,其时间复杂度为 $O(m^2\cdot n)$,此计算过程同样可以离线进行;步骤 e)用来计算用户对未评分项目的评分,其时间复杂度为 $O(n)$。总的来说,算法的时间复杂度为 $O(n)$,n 为系统的用户数目。

4　实验数据选取和算法评价指标

在实验中,采用 Movielens 数据集来测试算法的准确性和多样性。该数据集包含了 Movielens 推荐系统的 943 个用户(即 $n=943$)对 1 682 部电影(即 $m=1\ 682$)的 10 万个评分。其中,每个用户至少评价过 20 部电影,评分值为 1 ~ 5 之间的整数,评分越高表示用户越喜欢该电影。评分大于等于 3 表示用户喜欢该部电影,评分小于 3 表示用户不喜欢该部电影。

实验中,将数据集划分成训练集和测试集两部分,其中训练集占总数的 80%。用平均排名分数(rank score)来衡量算法推荐准确性。除此之外,用汉明距离(Hamming distance)评价多样性,用推荐项目的平均度$\langle K\rangle$评价流行性。

4.1　平均排名分数

对于用户 u_i,推荐算法会给用户一个排序的长度为 L 的推荐列表。根据测试集,如果用户 u_i 选择了项目 o_j,而 o_j 在推荐列表中排在第 $R_{i,j}$ 位,则认为项目 o_j 的相对位置为

$$r_{i,j}=\frac{R_{i,j}}{L}$$

因为测试集中的项目是用户实际选择过的,越准确的算法给测试集中的项目越靠前的相对位置,即 $r_{i,j}$ 越小。将测试集中的所有用户—项目数据的相对位置求平均,得到平均值$\langle r\rangle$,即平均排名分数。$\langle r\rangle$越小算法准确性越好。

4.2　多样性和流行性

对于任意的两个用户 u_i 和 u_j,其推荐列表之间的距离为

$$H_{i,j}=1-\frac{Q_{i,j}}{L}$$

其中:L 表示推荐列表的长度;$Q_{i,j}$ 代表用户 u_i 和用户 u_j 长度为 L 的推荐列表中相同的项目数目。计算出任意两个用户之间的汉明距离,然后计算其平均值 H,用 H 衡量算法的多样性。H 最大为 1,表示所有用户项目推荐列表全不相同,推荐多样性最好,即个性化程度最好;当 $H=0$ 时,表示所有用户的推荐列表完全相同。

用推荐列表中的 L 个项目的平均度$\langle K\rangle$来评价算法所推荐项目的流行性。平均度越小,说明不是非常流行的项目也能被推荐。

5　实验结果及分析

本文中设计了三组实验,首先分别对参数 λ 和 δ 对实验结果的影响进行实验,然后将 λ 和 δ 取一定值时的本文算法与文献[5]中提出的 NBI (network-based inference)算法[5] 和文献[6]中的 SA-CF(spreading activation approach for collaborative filtering)算法[6] 进行对比。

实验 1　参数 λ 对平均排名分数$\langle r\rangle$和 Hamming 距离 H 的影响,实验中 λ 的取值为 0 ~ 1 之间。实验结果如图 1 和 2 所示。

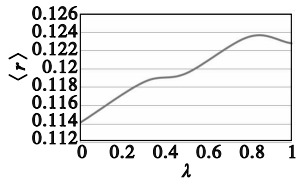

图1　参数λ对平均排名分数⟨r⟩的影响

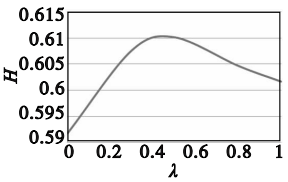

图2　参数λ对Hamming距离H的影响

图 1 和 2 分别显示了参数 λ 对算法的推荐准确性和多样性的影响。从图 1 中可以观察得到,随着 λ 的下降,$\langle r\rangle$值也有所下降,即推荐准确性得到提高。结合图 2 的结果可知,在该数据集中取 $\lambda=0.5$ 最优。

实验 2　参数 δ 对推荐准确性的影响,δ 用于调节大度节点的推荐能力。在该实验中 λ 取 0.5。实验结果如图 3 所示。

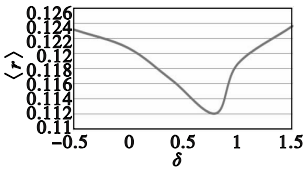

图3　参数δ对推荐准确性⟨r⟩的影响

从图 3 中可以看出,在计算用户相似性系数时,适当降低大度项目的推荐能力,并考虑项目的所有权值之和,降低用户不喜欢的项目的推荐能力,能够提高推荐准确性。在 $\delta=0.8$

时准确性最好,相对于未改进的算法(NBI 算法)准确性提高了 9.52%。

实验 3 对比实验

将本文提出的改进算法与 NBI 和 SA-CF 两种算法在多样性和流行性两个方面进行对比。实验结果如图 4 和 5 所示。其中,本文算法取 $\lambda = 0.5$,$\delta = 0.8$;L 表示推荐列表长度。

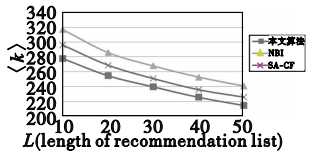

图4 本文算法与其他算法的推荐多样性的比较

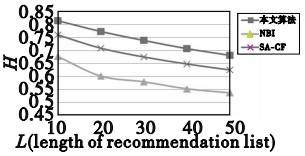

图5 本文算法与其他算法的项目流行性的比较

从图 4 和 5 中可以看出,本文的改进算法能够提高推荐多样性,并能够降低推荐项目的流行性,说明个性化程度得到了提高,推荐系统中不是非常流行的商品也能得到推荐。

6 结束语

针对基于网络结构的推荐算法没有区分用户对项目的评分的高低和大多数推荐算法都倾向于给用户推荐流行商品的问题,本文提出了一种改进的基于二部图网络结构的推荐算法。该算法不仅判断用户是否选择过项目,还要区分高低分,构成一个带权的网络。实验分析发现,适当降低低分在推荐系统中的作用能够提高推荐准确性和多样性。同时,算法在计算用户间的相似性时,综合考虑项目的度和项目的所有权值之和对推荐结果的影响。引入项目度与权值之和的比值 θ,调节 θ 与相似性系数的关系,以提高推荐质量。实验结果表明,在计算用户相似性时适当减弱大度项目的推荐能力,并降低大部分用户都不喜欢的项目(θ 较大)的推荐能力能够提高算法的性能。推荐准确性比未改进的算法提高了 9.52%,个性化程度也比 NBI 和 SA-CF 算法有了明显提高。而且本文算法还降低了推荐项目的流行性,表明该算法能够给目标用户推荐相对冷门的商品。

参考文献:

[1] 许海玲,吴潇,李晓东,等. 互联网推荐系统比较研究[J]. 软件学报, 2009,20(2):1-10.

[2] 刘建国,周涛,汪秉宏. 个性化推荐系统的研究进展[J]. 自然科学进展,2009,19(1):1-12.

[3] SCHAFER J B, KONSTAN J, RIEDL J. Recommender systems in E-commerce[C]//Proc of E-COMMERCE. 1999:158-166.

[4] HUANG Zan, ZENG D, CHEN H. A comparison of collaborative-filtering recommendation algorithms for E-commerce[J]. IEEE Intelligent Systems,2007,22(5):68-78.

[5] ZHOU Tao, REN Jie, MEDO M, *et al*. Bipartite network projection and personal recommendation[J]. Physical Review E, 2007, 76(4):046115.

[6] LIU Jian-guo, WANG Bing-hong, GUO Qiang. Improved collaborative filtering algorithm via information transformation[J]. International Journal of Modern Physics C,2009,20(2):285-293.

[7] ZHANG Yi-cheng, BLATTNER M, YU Yi-kuo. Heat conduction process on community networks as a recommendation model[J]. Physical Review Letters,2007,99(15):154301.

[8] LIU Jian-guo, ZHOU Tao, CHE Hong-an, *et al*. Effects of high-order correlations on personalized recommendations for bipartite networks[J]. Physica A,2010,389:881-886.

[9] SHANG Ming-sheng, LV Lin-yuan, ZHANG Yi-cheng, *et al*. Empirical analysis of Web-based user-object bipartite networks[J]. Europhysics Letters,2010,90(4):48006.

[10] HUANG Zan, ZENG D D, CHEN H. Analyzing consumer-product graphs: empirical findings and applications in recommender systems[J]. Management Science,2007, 53(7):1146-1164.

[11] ZHOU Tao, KUSCSIK Z, LIU Jian-guo, *et al*. Solving the apparent diversity-accuracy dilemma of recommender systems[J]. PNAS, 2010,107(10):4511-4515.

[12] ZHOU Tao, SU Ri-qi, LIU Run-ran, JIANG Luo-luo, *et al*. Accurate and diverse recommendations via eliminating redundant correlations[J]. New Journal of Physics,2009,11:123008.

[13] PAN Xin, DENG Gui-shi, LIU Jian-guo. Weighted bipartite network and personalized recommendation[J]. Physics Procedia, 2010, 3(5):1867-1876.

(上接第 770 页)

[3] LEE D D, SEUNG H S. Algorithms for non-negative matrix factorization[C]//Advances in Neural Information Processing 13 (Proc NIPS). 2000:556-562.

[4] YANG Zhi-rong, LAAKSONEN J. Multiplicative updates for non-negative projections[J]. Neurocomputing,2007,71(1-3):363-373.

[5] DING C, LI Tao, PENG Wei, *et al*. Orthogonal nonnegative matrix trifactorizations for clustering[C]//Proc of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press,2006:126-135.

[6] LI Zhao, WU Xin-dong, PENG Hong. Nonnegative matrix factorization on orthogonal subspace[J]. Pattern Recognition Letters, 2010,31(9):905-911.

[7] CHEN S S, DONOHO D L, SAUDERS M A. Atomic decomposition by basis pursuit[J]. SIAM Journal on Scientific Computing, 1999,20(1):33-61.

[8] MALLAT S, ZHANG Z. Matching pursuit with time-frequency dictionary[J]. IEEE Trans on Signal Processing,1993,41(12): 3397-3415.

[9] HOSEN M G, BABAIE-ZADEH M, JUTTEN C. A fast approach for overcomplete sparse decomposition based on smoothed l_0 norm[J]. IEEE Trans on Signal Processing,2009,57(1): 289-301.

[10] ZDUNEK R, CICHOCKI A. Nonnegative matrix factorization with constrained second order optimization[J]. Signal Processing, 2007,87(8):1904-1916.

[11] YANG Zhi-rong, OJA E. Linear and nonlinear projective nonnegative matrix factorization[J]. IEEE Trans on Neural Networks,2010,2(5):734-747.

[12] KIM H, PARK H. Sparse non-negative matrix factorizations via alternating nonnegativity constrained least squares for microarray data analysis[J]. Bioinformatics,2007,23(12):1495-1502.

[13] HOYER P O. Nonnegative matrix factorization with sparseness constraints[J]. Journal of Machine Learning Research,2004,5(9): 1457-1469.