

# 动态学习的非负矩阵分解算法

杨志君, 叶东毅

(福州大学 数学与计算机科学学院, 福建 福州 350108)

**摘要:**为在对现有增量型非负矩阵分解算法存在的一些缺陷进行改进, 给出了一个基于误差判断的增量算法有效性准则. 在此基础上, 利用增加样本前的非负矩阵分解结果进行增量分解初始化, 提出了一种新的动态非负矩阵分解算法. 在多个数据集上的实验结果表明该算法可以实现对基矩阵和编码矩阵的即时更新, 且具有较低的计算复杂度, 在处理动态数据集时, 还可有效识别噪声点, 是一个可行有效的动态分解算法.

**关键词:**非负矩阵分解; 动态学习; 初始化; 误差准则

**中图分类号:** TP181 **文献标识码:** A **文章编号:** 1673-4785(2010)04-0320-07

## A dynamic learning algorithm based on non-negative matrix factorization

YANG Zhi-jun, YE Dong-yi

(College of Mathematics and Computer Science, Fu Zhou University, Fu Zhou 350108, China)

**Abstract:** To improve the performance of the incremental non-negative matrix factorization algorithm, error estimation criteria for judging the effectiveness of the incremental algorithm was presented. Then, a new dynamic non-negative matrix factorization algorithm was proposed whereby incremental factorization was initialized with the already factorized matrices before adding new samples. Experimental results on a number of data sets showed that the proposed algorithm is capable of instantly updating both the base matrix and the code matrix. Another benefit of the method is that the computational complexity is relatively low. The proposed algorithm can also identify noise points when dealing with dynamic data. So it is a feasible and effective dynamic factorization algorithm.

**Keywords:** non-negative matrix factorization; dynamic learning; initialization; error criteria

近年来非负矩阵分解方法 (nonnegative matrix factorization, NMF)<sup>[1-4]</sup> 在聚类分析、人脸识别、语音识别、声音分离等领域取得了很好的应用. 但 NMF 也存在一些缺陷, 如不能动态地、实时地处理加入的数据向量, 当加入新的数据向量时只能将所有数据向量重新计算得到基矩阵和编码矩阵, 这在大规模数据的处理中显得不太合理.

针对以上问题, 人们已经提出了若干增量型 NMF 算法. 例如, Bucak 等人<sup>[5]</sup> 提出了一种增量非负矩阵分解算法 (incremental nonnegative matrix factorization, INMF). 该算法在增加新样本时, 保持原样本数据集编码不变, 通过一个加权目标函数来控制矩阵分解的无记忆性. 但在实际应用中样本编码经常

需要随样本数的动态增加做相应的调整. 例如, 在分析疫情的病例时, 将每个病人所需测量的生理参数是否异常作为二值 (0, 1) 样本, 则分解产生的基矩阵和编码矩阵分别为各个症状 (包含若干个有可能异常的生理参数) 和每个病例可能具有的若干症状. 当出现新的病例时, 应重新调整对各个病人的诊断, 即他们的症状. 保持编码矩阵不变显然是不合理的. Cao<sup>[6]</sup> 等人在满足矩阵分解惟一性的条件下, 使用原基矩阵和新样本来更新基矩阵, 得到一种新的动态非负矩阵分解算法 (online nonnegative matrix factorization, ONMF), 但它是以 NMF 满秩分解为前提的. 因此, 该算法不适用于要求基向量的个数小于样本矩阵的秩时的情况. 文献[7]中提出一种动态分块非负矩阵分解算法 BNMF, 该算法运用分块的思想对 NMF 算法进行修改. 将大矩阵按每类进行分块, 即同一类的训练样本形成一块小矩阵, 然后对每

收稿日期: 2009-06-08.

基金项目: 国家自然科学基金资助项目 (60805042).

通信作者: 叶东毅. E-mail: yiedy@fzu.edu.cn.

块小矩阵进行分解,最后再将分解后的小矩阵合成大矩阵.当每类的训练样本数较多时,BNMF也会存在和NMF一样的缺点.

上述增量型NMF算法均未给出判断动态非负矩阵分解算法优劣的标准,使算法的有效性无法得到衡量.鉴于此,本文给出一条误差准则用于判断增量型NMF算法的有效性.详细的定义在第3部分中给出.此外,考虑到在一些应用中,当增加样本时,需对编码矩阵和基矩阵均进行动态更新,且算法的执行时间应尽可能短,同时具有较小的剩余误差.在NMF的许多实际应用中,通常在相邻的2次迭代产生的剩余误差之差小于预设的阈值时中止迭代过程.据此,本文提出一种新的动态非负矩阵分解算法(DNMF).当增加样本时,用原样本集分解产生的编码矩阵和基矩阵对新样本集的分解进行初始化,使迭代从接近于阈值的位置开始.实验表明该方法有效的减少了迭代次数,缩短了收敛时间.

## 1 非负矩阵分解

### 1.1 数学定义

给定一个 $n$ 行 $m$ 列的非负矩阵 $V$ ,将其近似地分解成2个非负矩阵之积,分别记为 $W$ 与 $H$ ,使得 $V \approx WH$ ,即

$$V_{ij} \approx (WH)_{ij} = \sum_{a=1}^r W_{ia} H_{aj}. \quad (1)$$

分解之后的矩阵 $W$ 与 $H$ 的维数分别为 $n \times r$ 和 $r \times m$ . $r$ 的选取满足 $r < \min\{m, n\}$ .

如果用 $v$ 和 $h$ 分别表示矩阵 $V$ 与 $H$ 中对应的列向量,由式(1)可得 $V \approx Wh$ .由此可见,原始矩阵 $V$ 中的列向量 $v$ 以通过矩阵 $W$ 以一定的权重 $h$ 表示出来.因此,矩阵 $W$ 可以看作是基矩阵,其中的每一列代表基向量,而矩阵 $H$ 中的每一列代表与矩阵 $W$ 中基向量相对应的编码系数.为了找到一个矩阵分解 $W$ 与 $H$ ,使得 $\|V - WH\|$ 最小,且满足 $W \geq 0$ , $H \geq 0$ 可以使用如下的规则进行迭代计算<sup>[1]</sup>:

$$H_{aj} \leftarrow H_{aj} \frac{(W^T V)_{aj}}{(W^T W H)_{aj}}, \quad (2)$$

$$W_{ia} \leftarrow W_{ia} \frac{(V H^T)_{ia}}{(W H H^T)_{ia}}. \quad (3)$$

在计算过程中,随机选取非负的初始矩阵 $W$ 与 $H$ ,定义适当的迭代运算次数之后,按照式(2)和(3)进行迭代计算即可得到最优的 $W$ 和 $H$ .

### 1.2 在处理动态数据集上的不足

在处理动态数据集上,当增加样本时,其影响在分解结果中应有适当的表现,即分解产生的基矩阵和编码矩阵应随之更新.NMF算法在诸如此类的批处理应用中存在一些问题.由于 $V$ 的每一列对应于一个不同的样本,每增加一个样本即增加 $V$ 的度.NMF每次迭代的复杂度是 $O(mnr)$ ,显然随着维度增加其复杂度呈线性增长.因此,每增加一个样本即重新对 $V$ 进行NMF分解是不现实的.再者,NMF分解的计算复杂度和迭代次数呈线性关系,增加新样本时迭代次数相应增加使动态处理数据不具有可行性.

## 2 几个典型的增量型算法分析

### 2.1 ONMF<sup>[6]</sup>算法及其不足

该算法对其定义的目标函数加上如下限制,

$$\text{s. t. } h_i \times h_j = 0, j \neq i.$$

并将目标函数表示为

$$J = \frac{1}{2} \|V - WH\|_F^2 + \alpha I^T H H^T.$$

式中: $I$ 为对角元素为零,其余元素均大于零的矩阵, $\alpha$ 是一个正数.后根据KKT条件得迭代公式:

$$w_{ij} \leftarrow w_{ij} \frac{(V H^T)_{ij}}{(W H H^T)_{ij}}, \quad (4)$$

$$h_{ij} \leftarrow h_{ij} \frac{(W^T V)_{ij}}{(W^T W H + \alpha I^T H)_{ij}}. \quad (5)$$

最后算法利用 $U$ 和 $H$ 通过式(4)和式(5)来计算 $\tilde{W}$ 和 $\tilde{H}$ ,实现增量分解.

该算法是以满秩分解为前提的,因此不适用于要求基向量的个数小于样本矩阵的秩时的情况,且算法中要求 $W_1^*$ 可逆,这2点较大地限制了算法的应用范围.另外文中的2个参数 $\alpha$ 和 $I$ 并未给出明确的定义,代码也无法获得,使实验不能重现.

### 2.2 INMF<sup>[5]</sup>算法及其不足

算法用参数 $\alpha$ 和 $(1 - \alpha)$ 来控制前 $k$ 个样本和第 $(k + 1)$ 个样本对目标函数的影响:

$$F_{k+1} = \alpha F_k + (1 - \alpha) f_{k+1}.$$

接着使用梯度下降算法得出 $h_{k+1}$ 和 $W_{k+1}$ 的更新规则.

INMF算法是在假定新样本对前 $k$ 个样本的编码影响不大的前提下提出的,而在许多实际应用中,如视频监控,在原来静止的画面中,出现一个移动的物体,则在少量的几个新增样本出现后,应对基矩阵

和编码矩阵均做适度的调整,以使剩余误差尽量小. DNMF 算法在调整基矩阵的同时,将新增样本的影响即时施加到原样本的编码中,即随着样本的持续加入,编码矩阵随之做相应的调整.

INMF 算法使用  $\alpha$  和  $(1-\alpha)$  来控制前  $k$  个样本和第  $(k+1)$  个样本对目标函数的影响. 若  $\alpha$  接近于 1, 则第  $(k+1)$  个样本的影响可以忽略. 而文献[5]所进行的实验中,均削弱了新增样本的影响. 如在第 1 项实验中,为比较 NMF 和 INMF 算法分解背景样本数据集所产生的剩余误差,将  $\alpha$  分别设为 0.2 及 0.3, 均接近于 0, 降低了新增样本的影响. 显然,即使比较结果 INMF 的剩余误差较 NMF 小,也是没多大意义的. 文献[5]中第 2 和第 3 项实验也存在同样的问题.

### 3 本文算法的基本思想

DNMF 的目标是增加新样本时,将其影响即时施加到编码矩阵  $\mathbf{W}$  和基矩阵  $\mathbf{H}$  上,且尽量减少迭代次数,即具有较低的计算复杂度. 另外,增加样本时,对  $\mathbf{W}$  和  $\mathbf{H}$  的更新应使 DNMF 分解具有较低的重构误差. 且算法不需要满秩分解的限制,即应具有更广泛的应用. 以下描述 DNMF 的基本思想及算法步骤.

#### 3.1 相关定义

假设  $\mathbf{W}_k$  和  $\mathbf{H}_k$  为原  $k$  个样本进行 NMF 分解产生的编码矩阵和基矩阵,  $C_k$  为代价函数,即重构误差.

$$C_k = \|\mathbf{V} - \mathbf{W}_k \mathbf{H}_k\|^2 = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^k (V_{ij} - (\mathbf{W}_k \mathbf{H}_k)_{ij})^2.$$

增加一个样本时,将编码矩阵和基矩阵定义为  $\mathbf{W}_{k+1}$  和  $\mathbf{H}_{k+1}$ ,  $\mathbf{V}'$  为样本矩阵. 此时代价函数为

$$C_{k+1} = \|\mathbf{V}' - \mathbf{W}_{k+1} \mathbf{H}_{k+1}\|^2 = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^k (V'_{ij} - (\mathbf{W}_{k+1} \mathbf{H}_{k+1})_{ij})^2.$$

为判断增量算法的有效性,进行如下定义:

#### 定义 1 最大样本差

$$e_{\max} = \max \|v_i - \mathbf{W} \mathbf{h}_i\|, i = 1, 2, \dots, n.$$

式中:  $v_i$  是第  $i$  个样本向量,  $\mathbf{W}$  是分解产生的基矩阵,  $\mathbf{h}_i$  是第  $i$  个样本的编码向量,  $n$  为样本数.

当增加一个样本时,对新的样本数据矩阵重新进行 NMF 分解,设得到的最大样本差为  $e^*$ . 同时,设新样本数据矩阵  $\mathbf{V}'$  在原样本或者原样本分解结

果的基础上通过对原编码矩阵和原基矩阵进行某种修正得到的新基矩阵、新编码矩阵为  $\tilde{\mathbf{W}}$  和  $\tilde{\mathbf{H}}$ . 则可得增量差的定义如下:

#### 定义 2 增量差

$$\hat{e} = \|\mathbf{V}_{n+1} - \tilde{\mathbf{W}} \tilde{\mathbf{h}}_{n+1}\|.$$

式中:  $\mathbf{v}_{n+1}$  为新增样本向量,  $\tilde{\mathbf{h}}_{n+1}$  为  $\tilde{\mathbf{H}}$  中  $\mathbf{v}_{n+1}$  所对应的编码向量.

在实际应用中,为保证增量型 NMF 算法的有效性,应保证分解结果有一定的准确度,避免因算法本身导致分解的质量下降或下降太大. 为此提出判断增量型 NMF 算法有效性的误差准则:

**准则 1** 当增量型 NMF 算法的分解结果满足  $\hat{e} \leq e^*$  时,算法是比较合理的.

由于  $e^*$  是对  $\mathbf{V}'$  重新进行 NMF 分解得到的最大样本差,因此最好的情况是通过某种增量型算法得到的最大样本差不超过  $e^*$ ,而通常增量型算法都是对原编码矩阵和原基矩阵进行某种修正得到新的分解结果,难以达到这样的准确度. 鉴于此,提出以上准则,当增量差满足  $\hat{e} \leq e^*$  时,结果即是可接受的.

#### 3.2 算法的设计思路

对于动态数据集,新增样本和原先的样本集通常具有一定的相似性,即具有许多共同特征. 因此,新增样本对原先  $k$  个样本分解产生的基矩阵  $\mathbf{W}_k$  和编码矩阵  $\mathbf{H}_k$  进行的更新通常是不显著的. 鉴于此,将  $\mathbf{W}_k$  用于  $(k+1)$  个样本时  $\mathbf{W}_{k+1}$  的初始化,  $\mathbf{H}_k$  用于  $\mathbf{H}_{k+1}$  前  $k$  列的初始化. 相对于随机初始化,该初始化方法使迭代过程从较接近于局部最优收敛点的地方开始进行,且保证增加样本前后算法能以较大概率收敛于同一收敛点,有效的减少了迭代次数. 同时,保证了分解的准确度,实现对  $\mathbf{W}_{k+1}$  和  $\mathbf{H}_{k+1}$  的即时更新.

#### 3.3 算法设计

本节具体给出 DNMF 算法. 该算法是在 NMF 算法的基础上对其初始化方式进行改进得到的. 当新增一个样本时,

- 1) 保存  $k$  个样本时分解产生的基矩阵  $\mathbf{W}_k$  和编码矩阵  $\mathbf{H}_k$ .
  - 2) 用  $\mathbf{W}_k$  初始化  $\mathbf{W}_{k+1}$ ,  $\mathbf{H}_k$  初始化  $\mathbf{H}_{k+1}$  前  $k$  列.
  - 3) 对  $\mathbf{H}_{k+1}$  第  $k+1$  列进行随机初始化.
  - 4) 使用 2)、3) 的迭代规则进行计算.
  - 5) 符合迭代停止条件,算法结束,否则转 4).
- 新增样本时,和 INMF 相比, DNMF 对编码矩阵

和基矩阵均进行了调整,减小了剩余误差.且从 3.2 节的分析可知,算法第 2)步使得 DNMF 从较接近于收敛点的位置开始迭代过程,降低了计算复杂度.当增加样本时,若该类的训练样本数较大,相比 DNMF,BNMF 的计算复杂度较高.

4 实验结果与分析

为评价 DNMF 算法在动态数据集上的效果,进行了 5 项实验.第 1 个实验的数据集由 Matlab 生成;接下来的 3 个实验分别使用 PET 2001<sup>[8]</sup>、temporal image<sup>[6]</sup>和 swimmers<sup>[9]</sup>数据集;第 5 个实验展示了在分离的 swimmers 数据集上 DNMF 识别噪声点的能力.

实验在 Windows XP 环境下进行,Cpu 为赛扬 2.40 GHz,内存 512 M,Matlab 为 7.0 版.

4.1 使用 Matlab 生成的不同规模数据集

DNMF 算法使用前  $k$  个样本分解产生的基矩阵  $W_k$  和编码矩阵  $H_k$  来初始化  $W_{k+1}$  和  $H_{k+1}$ ,可使迭代以较大概率收敛于原收敛点,同时使迭代从较接近于收敛点的位置开始,减少了中止所需的迭代次数.实验中用 Matlab 随机生成函数生成不同规模的数据矩阵.当增加一个随机样本时分别测试 DNMF 的增量差和 NMF 的最大样本差.设样本数为  $m$  时,第  $i$  次迭代各样本所产生的平均误差为

$$T_i = \|V - W_i H_i\|^2 / m.$$

设置迭代中止条件为  $T_i - T_{i+1} \leq 0.0001$ .

表 1 中的次数指算法中止时经过的迭代次数.基向量个数取 10.

表 1 不同数据规模 DNMF 和 NMF 的样本差和迭代次数  
Table 1 Sample error and iteration times of DNMF and NMF with different data scales

未增加样 本前的数 据规模	增加后 NMF 所 需次数	增加后 NMF 最大 样本差	增加后 DNMF 所 需次数	增加后 DNMF 增量差
100 × 100	524	8.623	67	7.285
300 × 300	732	26.29	72	22.50
500 × 500	894	44.17	43	41.81
800 × 800	1 005	71.51	76	67.69
1 500 × 1 000	1 247	129.8	63	124.4
3 000 × 1 000	1 660	261.1	80	244.2

表 1 的实验结果表明 DNMF 在该数据集上符合准则 1,是有效算法.且由于迭代从接近于收敛点的位置开始,使 DNMF 和 NMF 相比减少了迭代次数,缩短了算法执行时间.表 1 中所示的最大样本差

平均到样本向量中的各个元素(元素个数为向量维数),误差约为 0.088,相较于上限 1,误差较小.

4.2 PET2001 数据集

PET2001 数据集是某道路视频监控的各帧图像的集合,其中包含不同光照条件下的背景图像和某些时刻人、车等的移动画面.静态背景和随时间变化的移动物体使该数据集较适合于评价非负分解算法的动态学习能力.

为便于处理,从中选取 800 张图像(dataset1, 0350-1150)并将其转换为 134 × 100 的 jpg 格式的灰度图.由于相邻帧的图像没有太大的变化,从每 10 帧中取一帧以更好的评价算法的动态学习能力,共 80 帧.

实验中,以前 35 张为初始样本集,接着每隔 6 张分别使用 DNMF 和 INMF 对图片构成的数据矩阵进行增量分解,取  $r$  为 20,迭代中止条件为  $T_i - T_{i+1} \leq 5$ ,INMF 的  $\alpha$  参数取 0.3.

图 1 为实验中 NMF 的最大样本差、DNMF 的增量差和 INMF 的增量差随样本数的变化情况.可看出在该数据集中 INMF 的增量差均高于 NMF 的最大样本差,随着样本数持续增加,INMF 算法的增量差大幅增长.而 DNMF 算法的增量差均低于 NMF 的最大样本差,证实 DNMF 的有效性.DNMF 相对于 INMF 具有更好的准确率是由于 INMF 在增加新图像时,原先的编码矩阵没有随之变化,而基图像已经更新,使增量差加大.另外,随着样本数的增加,DNMF 的增量差同样有较大增长,这和  $r$  有着密切的关系,具体研究留待以后的工作.

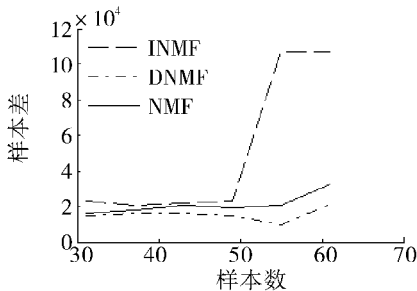


图 1 最大样本差和增量差随帧数的变化情况  
Fig. 1 The biggest sample error and incremental error changing with the frame number

图 2 为实验中各算法执行时间随帧数的变化情况.在实验的大部分过程中,DNMF 算法的执行时间低于 INMF.由于 DNMF 总是以接近于收敛点的位置开始迭代,因此有效的减少了算法的执行时间.随着

样本数的持续增加, DNMF 的执行时间超过 INMF, 这种情况同样和  $r$  的取值有较为密切的关系.

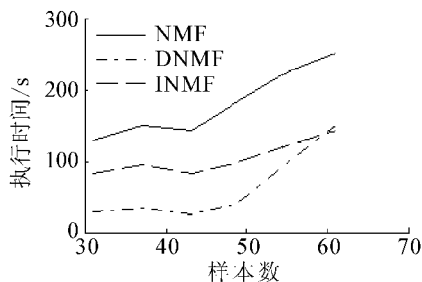


图2 各算法执行时间随帧数的变化情况

Fig.2 Run time of the algorithms changing with the frame number

#### 4.3 temporal image 数据集

文献[6]中使用 temporal image 数据集来验证 ONMF 算法在满秩分解条件下处理动态数据集的有效性. 由于暂时无法获得 ONMF 算法的代码进行实验对比, 因此本文也采用该数据集来验证 DNMF 的动态学习能力. 图3给出了数据集的部分生成因子, 每个生成因子对应于一张  $10 \times 10$  像素的图片, 图片中包含一条横线或竖线. 该数据集中的每张图片均是生成因子的某种线性组合. 为验证算法的动态学习能力, 图3中最左边的2个生成因子随时间左右或上下移动, 其他生成因子保持不变, 如图4.

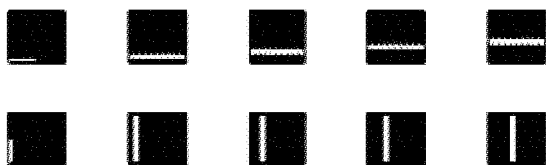


图3 temporal image 数据集的生成因子

Fig.3 Factors of temporal image dataset

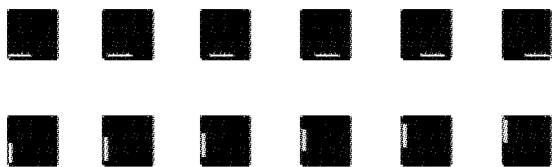


图4 随时间变化的生成因子

Fig.4 Factors changing with the time

根据图4中6种不同的情况各生成50张图片, 共300张. 实验中  $r$  取30, 迭代中止条件为  $T_i - T_{i+1} \leq 0.0001$ . 将前50张图片作为初始样本集, 剩下的250张图片每增加1张均进行 DNMF 分解, 观察分解产生因子的变化过程. 当样本集增至100张时, 分解产生的部分基图像如图5所示.



图5 样本数为100时的部分生成因子

Fig.5 Some of the factors with 100 samples

可看出图4中最左边的4个生成因子均出现在 DNMF 分解产生的基图像中. 说明 NMF 具有较强的提取线性样本集各组成成份的能力. 而 DNMF 在处理动态数据集时同样具有这样的功能. 当样本集增至200时, 分解产生的部分基图像如图6.

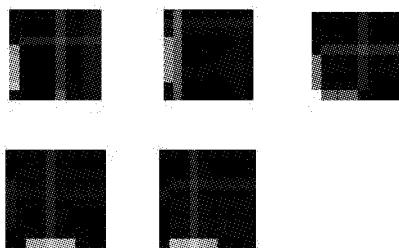


图6 样本数为200时的部分生成因子

Fig.6 Some of the factors with 200 samples

同样可看出有8个生成因子出现在基图像中, 其中图6的第一个图像和最后一个图像是由2个生成因子部分重叠产生的. 当样本集为300时算法生成的部分基图像如图7.

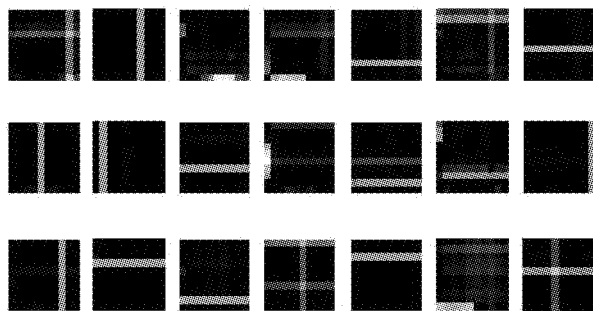


图7 样本数为300时的部分基图像

Fig.7 Some of the factors with 300 samples

可以看出, 在分解结果中, 各生成因子均有不同程度的体现. 另外, 最后分解得到 DNMF 算法的最大样本差为0.4425, 对300张图片进行 NMF 分解得到的最大样本差为0.3034. 可见 DNMF 分解具有较高的准确率.

#### 4.4 Swimmer 数据集

Swimmer 数据集包含一系列的黑白图像, 每个图像有4个可移动的部分(四肢), 每个部分可以旋转到4个不同的位置(关节). 且每个黑白图像的中心有一个包含12个像素的躯干和4个包含6个像

素的可移动部分,每个可移动部分可位于4个不同位置.因此总共有256张图像包括所有的肢体位置组合.每张图像包含 $32 \times 32$ 个像素.图8给出了该数据集的一个子集.

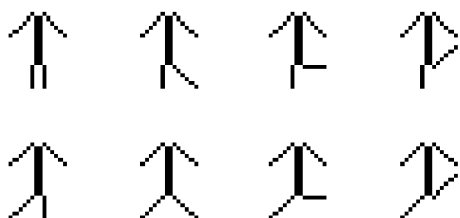


图8 Swimmer数据集

Fig. 8 Swimmer dataset

该数据集在文献[9]中是用来展示NMF算法发现局部特征(四肢)的能力.它遵守一些预定义的规则,如可分性和完全因子取样,即它是通过一种具有NMF样式的模型产生的.

首先从4个可移动的部分中去掉1个,则剩余部分可产生 $4^3 = 64$ 种位置组合,用 $V_{64}$ 表示.

对 $V_{64}$ 进行NMF分解,取 $r$ 为17,可产生17个基图像,如图9.

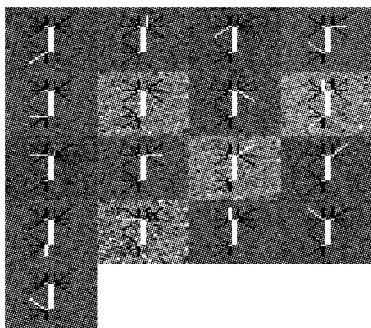


图9 NMF分解产生的基图像

Fig. 9 Base images decomposed by NMF

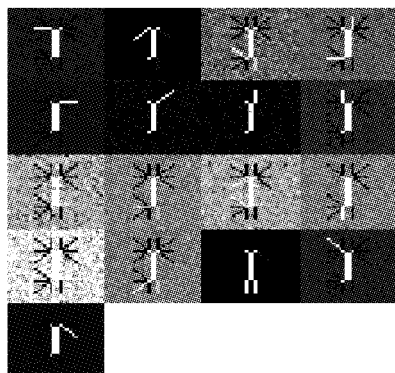


图10 DNMF分解产生的基图像

Fig. 10 Base images decomposed by DNMF

显然基图像中不包含去掉的部分.接着从图8所示的数据集中抽取一个样本加入 $V_{64}$ ,产生新的

数据矩阵 $V_{65}$ .对 $V_{65}$ 使用DNMF进行分解,所需迭代次数为591次,各样本产生的平均误差为0.996. DNMF产生的基图像如图10.

可以看出图9中去掉的部分完整的出现在若干个基图像中.表明DNMF对基矩阵和编码矩阵进行了即时更新.

#### 4.5 DNMF识别噪声的能力

文献[10]中分析了NMF和K-means聚类之间的关系,得出2种算法具有相同的目标函数. DNMF作为NMF的改进算法同样具有聚类效果.在这个小节里检验了DNMF在识别噪声方面的效果.和上个实验一样从swimmers数据集的4个活动部分中去掉其中一个,产生64种组合 $V_{64}$ .把4个活动部分组成的256张图像设为 $V$ .2组图像若定义为图像识别问题,显然它们应分别对应于不同的聚类.

对其中一组 $V_{64}$ 进行NMF分解,设为2,产生的编码矩阵中每一列有2个元素,将其作为2维空间的坐标得图11.从图中可以看出64张图像根据其编码分成了4类,表明NMF具有聚类效果.

从另一组图像 $V$ 中取出一副图像放入 $V_{64}$ ,得到新的一组图像 $V'$ .该图像包含4个可活动的部分,显然,可以将其视为噪声点.对 $V'$ 进行DNMF分解,产生的编码矩阵,同样可得其2维空间的坐标如图12.

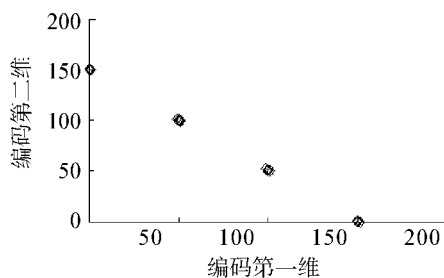


图11 NMF分解产生的聚类

Fig. 11 Clusters decomposed by NMF

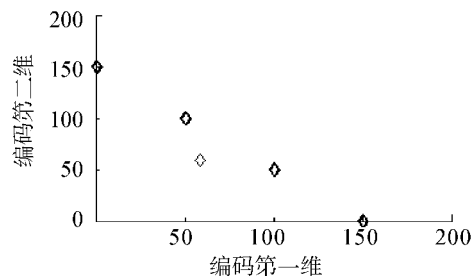


图12 DNMF分解产生的聚类

Fig. 12 Clusters decomposed by DNMF

观察图 12 可知新加入的图像和其他聚类相比处于分离的位置,可以判断该点为噪声点.另外,从实验结果可以看出新增图像对其他图像的编码并未产生较大影响,表明 DNMF 在动态数据集上具有抗噪性.

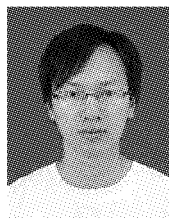
## 5 结束语

在 NMF 算法的基础上提出了一种新的动态学习算法 DNMF.实验表明该算法符合本文提出的准则 1,合理有效,且较 INMF 提高了分解准确率.由于 ONMF 算法的 2 个参数未给出明确的定义,代码无法获得,不能重现文中的实验,因此本文未能对 ONMF 和 DNMF 进行实验对比,但验证了 DNMF 在 temporal image 数据集上的有效性.在动态数据集的处理中,DNMF 对基矩阵和编码矩阵均进行了即时更新.此外,DNMF 在识别噪声、抗噪方面具有一定的用途.

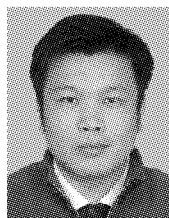
## 参考文献:

- [1] LEE D D, SEUNG H S. Learning the parts of objects by nonnegative matrix factorization[J]. Nature, 1999 (401): 788-791.
- [2] XU Wei, LUU Xin, GONG Yihong. Document clustering based on non-negative matrix factorization[C]//proceedings of the 26th annual international ACM SIGIR conference, Toronto, Canada, 2003: 267-273.
- [3] GUILLAMET D, VITRIA J. Nonnegative matrix factorization for face recognition[C]//Proc Conf Topics in Artificial Intelligence. Alberta, Canada, 2002: 336-344.
- [4] VIRTANEN T. Monaural Sound source separation by non-negative matrix factorization with temporal continuity and sparseness criteria [J]. IEEE Transactions on Audio, Speech, and Language Processing, 2007, 15 (3): 1066-1074.
- [5] BUCAK S S, GUNSEL B, GURSOY O. Incremental non-negative matrix factorization for dynamic background modeling[C]//ICEIS International Workshop on Pattern Recognition in Information Systems. Funchal, Portugal, 2007: 107-116.
- [6] CAO Bin, SHEN Dou, SUN Jiantao, et al. Detect and track latent factors with online nonnegative matrix factorization[C]//The 20th International Joint Conference on Artificial Intelligence. Hyderabad, India, 2007: 2689-2694.
- [7] CHEN W S, PAN B B, FANG B, et al. A novel constraint non-negative matrix factorization criterion based incremental learning in face recognition[C]//Proceedings of International Conference on Wavelet Analysis and Pattern Recognition. Hong Kong, 2008: 292-297.
- [8] PETS Video Database[EB/OL]. <http://ftp.pets.rdg.ac.uk/>.
- [9] DONOHO D, STODDEN V. When does non-negative matrix factorization give a correct decomposition into parts? [C]//Proceedings of the Seventeenth Annual Conference on Neural Information Processing Systems, Vancouver and Whistler. British Columbia, Canada, 2003: 101-108.
- [10] LI T, DING C. The relationships among various nonnegative matrix factorization methods for clustering[C]//Sixth IEEE International Conference on Data Mining. Hong Kong, 2006: 362-371.

### 作者简介:



杨志君,男,1985年生,硕士研究生,主要研究方向为数据挖掘.



叶东毅,男,1964年生,教授,博士生导师,主要研究方向为计算智能、数据挖掘.曾获得 1988 年度国家科技进步二等奖(主要成员)、1 项福建省科学技术二等奖和 2 项福建省科学技术三等奖.出版著作和教材 6 部,发表论文 70 余篇.