

在线学习非负矩阵分解

何孝文 胡一飞 王海平 陈默

(广东工业大学自动化学院 广州 510006)

摘要 文中提出了一种新的在线形式的非负矩阵分解,即在线学习非负矩阵分解(OLNMF)。OLNMF 算法采用了增量形式的非光滑模型,并采用“选择遗忘法”控制新样品和旧样品的权重,提高了算法的计算效率,减少了计算复杂度。OLNMF 算法能处理大型的实时更新的数据集,并得到稀疏度更高的基矩阵。实验结果表明,在多个人脸数据集中,相对于 INMF,ONMFO,Lp-INMF,OLNMF 具有更好的稀疏性;在 EEG 数据集中,基于 OLNMF 的 SVM 分类方法能得到更好的分类准确率。

关键词 在线学习,非负矩阵分解,特征提取,稀疏约束

中图分类号 TP391 文献标识码 A

Online Learning Nonnegative Matrix Factorization

HE Xiao-wen HU Yi-fei WANG Hai-ping CHEN Mo

(School of Automation,Guangdong University of Technology,Guangzhou 510006,China)

Abstract This paper proposed a new nonnegative matrix factorization of online form ,namely online learning nonnegative matrix factorization(OLNMF). The OLNMF algorithm uses incremental forms of non-smooth model, and adopts “anmesic average method” to control the weight of new and old samples,improving the computational efficiency and reducing the computational complex. OLNMF algorithm can deal with large real-time update data sets,and extract more sparse base matrix. Compared with INMF,ONMFO,Lp-INMF,experiments on face databases show that the proposed method achieves better sparsity,and SVM classification method base on OLNMF achieves better classification accuracy on EEG database.

Keywords Online learning,Nonnegative matrix factorization,Feature extraction,Sparseness constraints

1 引言

特征提取的目的是在高维的数据集中寻找一个适合的低维特征空间,常用的方法有 PCA^[1] 和 VQ^[2],但是 PCA 和 VQ 没有增加非负约束,两种方法提取的特征元素可能存在负值,负值在实验生活中没有对应的物理意义。1999 年, Lee 和 Seung 提出了一种新的特征提取方法——非负矩阵分解(NMF)^[3]。NMF 是一种可以提取局部特征的降维工具,而且对基矩阵和系数矩阵增加了非负约束,这间接导致了 NMF 的分解结果具有稀疏性。NMF 由于其非负性和稀疏性,已经成为一种提取数据特征的工具,已经广泛应用于各个领域,如人脸识别^[4]、盲源分离^[5]、深度神经网络^[6]和光谱解混^[7-9]。

某些领域中的数据维度太大,会导致 NMF 的分解结果不够稀疏,不能提取局部特征。为了增加基矩阵和系数矩阵的稀疏性,Hoyer 在 2002 年和 2004 年分别提出了两篇经典的稀疏约束 NMF:一个是非负稀疏编码(NNSC)^[22],该算法应用欧氏距离平方来进行相似性衡量,并对系数矩阵添加了 L1 范数约束,另一个是在目标函数后面增加稀疏度约束的模型(NMFSC)^[23]。另一方面,在 2006 年,Pascual-Montano 提出了一种 nsNMF^[10],通过在原始 NMF 的模型中增加一个光滑矩阵来同时提高基矩阵和系数矩阵的稀疏度。近几年,Ra-

jabi 等^[12]在原始 NMF 的模型上增加了 L1/2 范数和图约束,得到了新模型,该模型在光谱解混领域中有很好的效果。

在更新大规模数据集的过程中,传统的 NMF 不适合处理这类数据集。一方面,NMF 的计算复杂度和数据集的维度成正比,数据集越大,计算复杂度越高。另一方面,NMF 是一种批处理方式,当采集到新样品时,需要重新计算,导致旧样品的重复计算,浪费计算资源。2007 年,Bucak 和 Günsel 提出了 INMF^[11]的理论,即一种大型的在线更新数据集的处理方法。由于 INMF 十分适用于在线处理大型数据集,研究学者对在线形式的 NMF 进行了深入研究,并取得众多研究成果。Zhao 等提出了 ONMFO^[13],这是一种在离群值影响下依然能提取有良好特征的基矩阵的在线形式 NMF。Dang 等提出了 Lp-INMF^[15],INMF 的模型增加了 Lp 范数,用于提高 SAR 目标识别的准确率。在线形式的 NMF 也被成功运用到实际领域当中,如人脸识别^[14]、视觉捕捉^[16]、前景监测^[17]。

在本文中,我们提出了一种新的在线形式的 NMF,即在线学习非负矩阵分解(OLNMF)。OLNMF 算法采用了 nsNMF 的模型,引入了增量学习的形式,采用选择遗忘法^[18]的加权函数控制新样品和旧样品的权重。与 INMF 不同的是:在线学习非负矩阵分解的梯度下降法采用的是自然梯度。在线学习非负矩阵分解不仅能在线处理大型的实时更新的数

何孝文(1994—),男,硕士生,主要研究方向为在线非负矩阵分解及其应用,E-mail:786551691@qq.com(通信作者);胡一飞(1993—),男,硕士生,主要研究方向为稀疏非负矩阵分解;王海平(1993—),男,硕士生,主要研究方向为心肺音分离的方法及应用;陈默(1995—),男,硕士生,主要研究方向为非负矩阵分解在心电信号中的应用。

据集,而且能提取出稀疏度较高的基矩阵。

2 非负矩阵分解理论及概念

2.1 非负矩阵分解(NMF)

传统的 NMF 方法可以表示为以下的问题:设数据矩阵 $V \in R^{m \times n}$, NMF 将数据矩阵 V 近似地分解成非负的基矩阵 $W \in R^{m \times r}$ 和非负的系数矩阵 $H \in R^{r \times n}$, 目的是使得乘积 WH 无穷接近于 V 。 $r(m, n$ 远大于 $r)$ 是一个预设的降维参数。NMF 数学模型如下:

$$V \approx WH \quad (1)$$

数据矩阵 V 由 n 个列向量组成。每一个列向量在系数矩阵 H 中都有与之相对应的系数向量,如式(2)所示。

$$v_c = Wh_c, c=1, 2, \dots, n \quad (2)$$

欧氏距离的目标函数为:

$$\begin{cases} \min: D(V \| WH) = \frac{1}{2} \| V - WH \|_2^2 \\ \text{s. t. } W, H \geq 0 \end{cases} \quad (3)$$

为了解求目标函数, Lee 等^[19]利用交替迭代的方法依次更新基矩阵 V 和系数矩阵 H , 巧妙地选择学习率, 并且严格证明了迭代算法的收敛性。只要基矩阵 W 和系数矩阵 H 的初始值是非负的, 就可以保证每一步迭代运算得到的结果是非负的。基矩阵 W 和系数矩阵 H 的迭代更新公式如下:

$$\begin{cases} W_{ia} \leftarrow W_{ia} \frac{(VH^T)_{ia}}{(WHH^T)_{ia}} \\ H_{au} \leftarrow H_{au} \frac{(W^T V)_{au}}{(W^T WH)_{au}} \end{cases} \quad (4)$$

2.2 非光滑非负矩阵分解(nsNMF)

Pascual-Montano^[6]提出了一种非光滑形式的非负矩阵分解算法(nsNMF), 该算法模型如下:

$$V = WSH \quad (5)$$

其中, V, W, H 的意义与原始的 NMF 一样。与原始的 NMF 模型不同的是, nsNMF 加入了一个非负平滑矩阵 $S \in R^{r \times r}$, S 的定义如下:

$$S = (1 - \theta)I + \frac{\theta}{r} \mathbf{1}\mathbf{1}^T \quad (6)$$

其中, I 是一个 $r \times r$ 的单位矩阵, $\mathbf{1}$ 是一个全为 1 的 $r \times 1$ 的列向量, 参数 $\theta \in [1, 0]$ 。平滑矩阵 S 的作用是同时提升基矩阵 W 和系数矩阵 H 的稀疏度。

3 在线学习非负矩阵分解

3.1 迭代公式推导过程

OLNMF 的模型采用的是 nsNMF 的模型, 如式(5)所示。设旧数据为 V_k, V_k 分解得到后的基矩阵 W_k 和系数矩阵 H_k , 光滑矩阵 S 的定义如式(6)所示。OLNMF 目标函数 D_k 如式(7)所示。

$$\begin{aligned} D_k &= \frac{1}{2} \| V_k - W_k SH_k \|_2^2 \\ &= \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^k ((V_k)_{ij} - (W_k SH_k)_{ij})^2 \end{aligned} \quad (7)$$

随着样品的数量增加, W_{k+1} 的变化就越越来越小。在大型数据集中, 样品的数量是十分庞大的, 此时, W_k 近似于 W_{k+1} 。基于式(7)可得:

$$D_k \approx \frac{1}{2} \| V_k - W_{k+1} SH_k \|_2^2 \quad (8)$$

另外一方面, H_{k+1} 的前 k 列近似于 H_k , 同理, V_{k+1} 的前 k 列

近似于 V_k 。当采集到新的样品 v_{k+1} 时, 目标函数 D_{k+1} 定义为:

$$\begin{aligned} D_{k+1} &= \frac{1}{2} \| V_{k+1} - W_{k+1} SH_{k+1} \|_2^2 \\ &= \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^{k+1} ((V_{k+1})_{ij} - (W_{k+1} SH_{k+1})_{ij})^2 \\ &= \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^k ((V_{k+1})_{ij} - (W_{k+1} SH_{k+1})_{ij})^2 + \frac{1}{2} \sum_{i=1}^m ((V_{k+1})_{i(k+1)} - (W_{k+1} SH_{k+1})_{i(k+1)})^2 \\ &= \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^k ((V_k)_{ij} - (W_{k+1} SH_k)_{ij})^2 + \frac{1}{2} \sum_{i=1}^m ((v_{k+1})_i - (W_{k+1} SH_{k+1})_i)^2 \\ &= \frac{1}{2} \| V_k - W_{k+1} SH_k \|_2^2 + \frac{1}{2} \| v_{k+1} - W_{k+1} SH_{k+1} \|_2^2 \\ &\approx D_k + d_{k+1} \end{aligned} \quad (9)$$

其中, $d_{k+1} = \frac{1}{2} \| v_{k+1} - W_{k+1} SH_{k+1} \|_2^2$, d_{k+1} 是新样品 v_{k+1} 的目标函数。为了增加每一个新样品的利用效率, D_{k+1} 采用选择遗忘法。如式(10)所示:

$$D_{k+1} = \alpha D_k + \beta d_{k+1} \quad (10)$$

其中, α 和 β 分别对应旧数据和新数据的权重, 其中, $\alpha + \beta = 1$, α 和 β 都大于 0。 W_{k+1} 和 h_{k+1} 迭代更新公式采用梯度下降法^[20], 对 D_{k+1} 求 W_{k+1} 和 h_{k+1} 的偏导, 如式(11)和式(12)所示:

$$\frac{\partial D_{k+1}}{\partial W_{k+1}} = \alpha \frac{\partial D_k}{\partial W_{k+1}} + \beta \frac{\partial d_{k+1}}{\partial W_{k+1}} \quad (11)$$

$$\frac{\partial D_{k+1}}{\partial h_{k+1}} = \alpha \frac{\partial D_k}{\partial h_{k+1}} + \beta \frac{\partial d_{k+1}}{\partial h_{k+1}} = \beta \frac{\partial d_{k+1}}{\partial h_{k+1}} \quad (12)$$

梯度下降法采用自然梯度, Amari^[21]利用黎曼几何的有关理论, 证明梯度下降法的最佳下降方向不是负常规梯度方向, 而是负黎曼梯度。计算其偏导后, 我们可以得到:

$$\begin{aligned} W_{k+1} \leftarrow W_{k+1} - \eta_w \odot \left(\frac{\partial D_{k+1}}{\partial W_{k+1}} W_{k+1}^T + W_{k+1} \right) \\ \leftarrow W_{k+1} + \eta_w [(\alpha V_k H_k^T S^T + \beta v_{k+1} h_{k+1}^T S^T) W_{k+1}^T + \\ \delta - (\alpha W_{k+1} S H_k H_k^T S^T + \beta W_{k+1} S h_{k+1} h_{k+1}^T S^T) W_{k+1}^T + \\ W_{k+1} - \delta] \end{aligned} \quad (13)$$

$\delta = 10^{-9}$, 这是为了防止迭代运算中出现分母为零的情况。为了保证迭代运算能获得一个新的非负基矩阵 W_{k+1} , 取 η_w 为:

$$\eta_w = W_{k+1} \oslash [(\alpha W_{k+1} S H_k H_k^T S^T + \beta W_{k+1} S h_{k+1} h_{k+1}^T S^T) W_{k+1}^T + W_{k+1} + \delta] \quad (14)$$

将式(14)代入式(13)得到 W_{k+1} 的迭代更新公式:

$$\begin{aligned} W_{k+1} \leftarrow \oslash W_{k+1} [(\alpha V_k H_k^T S^T + \beta v_{k+1} h_{k+1}^T S^T) W_{k+1}^T + \delta] \\ \oslash [(\alpha W_{k+1} S H_k H_k^T S^T + \beta W_{k+1} S h_{k+1} h_{k+1}^T S^T) W_{k+1}^T + \\ W_{k+1} + \delta] \end{aligned} \quad (15)$$

\odot 和 \oslash 分别代表矩阵对应位置相乘或相除。为了避免 W_{k+1} 每一列的规格参差不齐, 我们需要对 W_{k+1} 进行归一化处理:

$$(W_{k+1})_{ij} \leftarrow (W_{k+1})_{ij} / \sum_{i=1}^m (W_{k+1})_{ij} \quad (16)$$

另一方面, h_{k+1} 的迭代更新公式如式(17)所示:

$$h_{k+1} \leftarrow h_{k+1} - \eta_h \odot [(S^T W_{k+1}^T v_{k+1} + \delta - S^T W_{k+1}^T W_{k+1} S h_{k+1} - \delta) \beta] \quad (17)$$

同理, 为了保证每次迭代运算更新得到一个非负的 h_{k+1} , 学习率 η_h 为:

$$\eta_h = h_{k+1} \oslash [(S^T W_{k+1}^T W_{k+1} S h_{k+1} + \delta) \beta] \quad (18)$$

最终, 我们可以得到 h_{k+1} 的迭代更新公式, 如式(19)所示:

$$h_{k+1} \leftarrow h_{k+1} \oslash [S^T W_{k+1}^T v_{k+1} + \delta] \oslash [S^T W_{k+1}^T W_{k+1} S h_{k+1} + \delta] \quad (19)$$

为了避免重复计算和减少算法的计算量,我们需要储存上一次迭代运算留下来的信息,并将其运用到下一次迭代运算中。每一次我们将储存的信息如式(20)、式(21)所示:

$$\mathbf{V}_{k+1} \mathbf{H}_{k+1}^T \mathbf{S}^T = \alpha \mathbf{V}_k \mathbf{H}_k^T \mathbf{S}^T + \beta \mathbf{v}_{k+1} \mathbf{h}_{k+1}^T \mathbf{S}^T \quad (20)$$

$$\mathbf{S} \mathbf{H}_{k+1} \mathbf{H}_{k+1}^T \mathbf{S}^T = \alpha \mathbf{S} \mathbf{H}_k \mathbf{H}_k^T \mathbf{S}^T + \beta \mathbf{S} \mathbf{h}_{k+1} \mathbf{h}_{k+1}^T \mathbf{S}^T \quad (21)$$

每一次迭代运算都会储存相应的信息 $\mathbf{V}_{k+1} \mathbf{H}_{k+1}^T \mathbf{S}^T$ 和 $\mathbf{S} \mathbf{H}_{k+1} \mathbf{H}_{k+1}^T \mathbf{S}^T$ 。举个例子,当算法计算到第 $k+1$ 个样品时,储存 $\mathbf{V}_{k+1} \mathbf{H}_{k+1}^T \mathbf{S}^T$ 和 $\mathbf{S} \mathbf{H}_{k+1} \mathbf{H}_{k+1}^T \mathbf{S}^T$,当采集到第 $k+2$ 个样品后,由于我们在上一次迭代运算中已经计算好 $\mathbf{V}_{k+1} \mathbf{H}_{k+1}^T \mathbf{S}^T$ 和 $\mathbf{S} \mathbf{H}_{k+1} \mathbf{H}_{k+1}^T \mathbf{S}^T$,在这次计算 $\mathbf{V}_{k+2} \mathbf{H}_{k+2}^T \mathbf{S}^T$ 和 $\mathbf{S} \mathbf{H}_{k+2} \mathbf{H}_{k+2}^T \mathbf{S}^T$ 的迭代运算中,不需要重复计算,可以直接运用。

3.2 OLNMF 算法的流程

OLNMF 算法的实验流程如下:

(1)初始化:取原始数据 \mathbf{V} 的前 k 列作为训练集 \mathbf{V}_k ,对 \mathbf{V}_k 进行 nsNMF 处理,得到 \mathbf{W}_k 和 \mathbf{H}_k 。

(2)归一化: $[\mathbf{v}_{k+1}]_i = [\mathbf{v}_{k+1}]_i / \sum_{j=1}^n [\mathbf{v}_{k+1}]_j$ 。

(3)在线学习[内循环 s 次($s=10 \sim 50$)]:

1)令 $\mathbf{W}_{k+1} = \mathbf{W}_k$,迭代更新公式 \mathbf{h}_{k+1} ,如式(19)所示。

2)迭代更新公式 \mathbf{W}_{k+1} ,如式(15)和式(16)所示。

(4)停止:令 $k=k+1$,若 k 超过整体样品的数量,则停止算法,若没有超过,则返回到步骤 2)。

(5)得到 \mathbf{W}_{k+1} 和 \mathbf{h}_{k+1} 。

4 实验与结果分析

稀疏度量方法是采用 Hoyer 提出的基于向量的 L1 范数和 L2 范数的关系来计算的,如式(20)所示:

$$\text{Sparseness}(\mathbf{w}) = \frac{\sqrt{m} - (\sum |\omega_i|) / \sqrt{\sum \omega_i^2}}{\sqrt{m} - 1} \quad (20)$$

其中, ω_i 是向量 \mathbf{w} 的其中一个元素, m 是向量 \mathbf{w} 的维度。在下面实验中,计算矩阵的稀疏度,定义矩阵的“平均稀疏度”是矩阵每一列向量的平均稀疏度。

本节中,为了更好地量化每一次迭代更新后 \mathbf{W}_{k+1} 的变化,定义代价函数 $F(\mathbf{W}_{k+1})$:

$$F(\mathbf{W}_{k+1}) = \frac{\|\mathbf{W}_{k+1} - \mathbf{W}_k\|_F^2}{\|\mathbf{W}_k\|_F^2} \quad (21)$$

4.1 稳定性实验

4.1.1 计算机生成的随机数据集

本节中,计算机用式(6)的模型生成一个非负且稀疏度很高的数据集 $\mathbf{V} \in R^{300 \times 500}$,其中 $\mathbf{W} \in R^{300 \times 25}$ 和 $\mathbf{H} \in R^{25 \times 500}$ 的稀疏度都超过 0.8,平滑矩阵 \mathbf{S} 的参数 $\theta=0.5$ 。本次实验中,初始化步骤取数据集的前 50 列进行 50 次 nsNMF 运算;在线学习步骤对剩下的数据集的每一列依次进行 OLNMF 运算,最终获得 \mathbf{W}_{k+1} ,并计算代价函数 $F(\mathbf{W}_{k+1})$ 和基矩阵 \mathbf{W}_{k+1} 的平均稀疏度。其中,OLNMF 的参数设定如下: $\alpha=0.9, \beta=0.1, r=25, s=30$ 。

为了验证算法的收敛性,对 θ 设定了 4 个值,分别进行了 4 次实验,得到 4 条代价函数曲线和 4 条稀疏度曲线,如图 1 和图 2 所示。在图 1 中,4 条代价函数曲线是收敛的,验证了算法在不同 θ 下依然可能分解得到稳定的基矩阵 \mathbf{W}_{k+1} 。随着 θ 的增加,代价函数 $F(\mathbf{W}_{k+1})$ 曲线的收敛速率加快,基矩阵 \mathbf{W}_{k+1} 能更快地达到稳定。

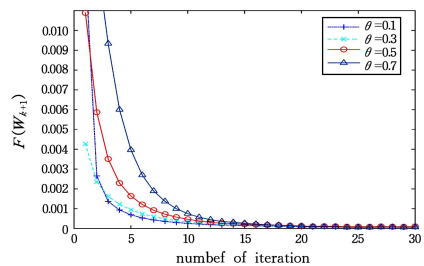


图 1 4 条代价函数曲线

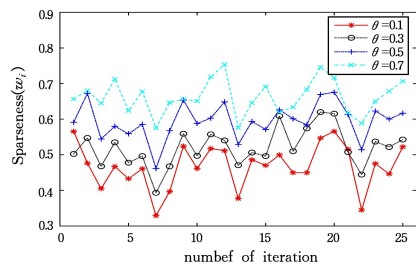


图 2 \mathbf{W}_{k+1} 的列向量稀疏度曲线

4.2 收敛性实验

4.2.1 计算机生成的随机数据集

在 4.1 节的实验结果中,计算得到了 4 条矩阵 \mathbf{W}_{k+1} 的列向量稀疏度曲线,如图 2 所示。 \mathbf{W}_{k+1} 的每一列的稀疏度都会随着 θ 的增加而增加。

为了体现 OLNMF 算法在稀疏度上面的优势,OLNMF 和 3 种在线形式的 NMF 算法进行了比较。在线形式的 NMF 的初始化步骤是要提取部分数据进行批处理 NMF,并分解得到 \mathbf{W}_k ,把 \mathbf{W}_k 作为在线学习步骤中的初始矩阵。4 种在线形式的 NMF 的初始化步骤中批处理 NMF 分别采用 nsNMF, NMF, NMFO, Lp-NMF。4 种在线 NMF 的实验流程如图 3 所示。

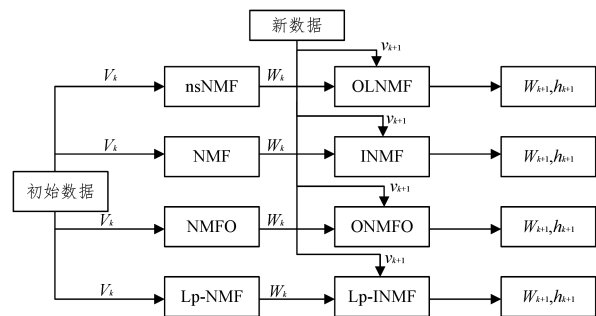


图 3 4 种在线 NMF 实验流程

在初始化步骤,取随机数据集的前 50 列进行批处理 NMF;在线学习步骤,对剩下的 450 列进行在线 NMF 处理 (OLNMF, INMF, ONMFO, Lp-INMF);最后计算基矩阵 \mathbf{W}_{k+1} 的平均稀疏度。本次实验同样设定了 10 种不同的初始状态,参数设置如下:

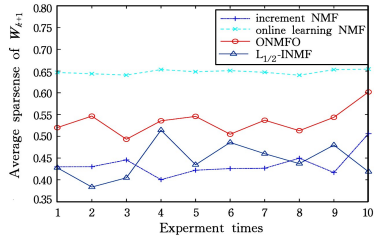
(1)在 OLNMF 中, $\alpha=0.9, \beta=0.1, r=25, s=30, \theta=0.5$;

(2)在 INMF 中, $\alpha=0.9, \beta=0.1, r=25$;

(3)在 ONMFO 中, $\lambda=0.01, r=25$;

(4)在 Lp-INMF 中, $p=0.5, \lambda=0.1, r=25$ 。

实验中,设置了 10 种不同的初始状态,生成了 4 条稀疏度曲线。实验结果如图 4 所示,其中 OLNMF 的基矩阵 \mathbf{W}_{k+1} 的平均稀疏度是最高的。

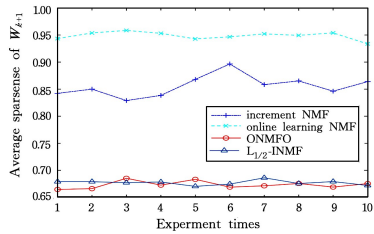
图4 随机数据集中4种在线NMF的 W_{k+1} 平均稀疏度曲线

4.2.2 CBCL数据集

为了验证算法应用到实际数据集时在稀疏度方面也有优势,采用CBCL数据集进行稀疏度比较。CBCL数据集由16个人在不同的角度、光线、尺寸下拍摄的2429张人脸图像组成的。在初始化步骤,取CBCL数据集的前100列(100张人脸图像)进行批处理NMF;在在线学习步骤,对后面的1900列进行在线NMF(OLNMF, INMF, ONMFO, L_p-INMF);最后计算基矩阵 W_{k+1} 的平均稀疏度。本次实验同样设定了10种不同的初始状态,参数如下:

- (1)在OLNMF中, $\alpha=0.9, \beta=0.1, r=49, s=10, \theta=0.5$;
- (2)在INMF中, $\alpha=0.9, \beta=0.1, r=49$;
- (3)在ONMFO中, $\lambda=0.01, r=49$;
- (4)在L_p-INMF中, $p=0.5, \lambda=0.1, r=49$ 。

实验结果如图5所示,OLNMF的基矩阵 W_{k+1} 的平均稀疏度依然是最高的,验证了OLNMF算法应用到人脸数据集中稀疏度方面依然具有优势。

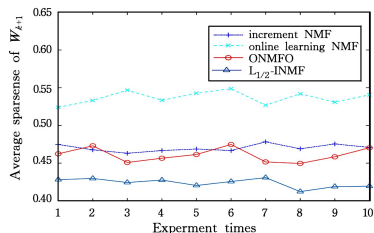
图5 CBCL数据集中4种在线NMF的 W_{k+1} 平均稀疏度曲线

4.2.3 ORL数据集

为了验证算法的鲁棒性,实验采用另一种人脸数据集——ORL数据集。ORL数据集由40个人在不同的角度、光照拍摄出来的400张人脸灰度图像组成。本次实验同样设定了10种不同的初始状态,在初始化步骤中,取ORL数据集的前40列进行4种批处理NMF;在在线学习步骤中,对剩下的360列进行在线NMF(OLNMF, INMF, ONMFO, L_p-INMF);最后计算基矩阵 W_{k+1} 的平均稀疏度。参数如下:

- (1)在OLNMF中, $\alpha=0.9, \beta=0.1, r=25, s=50, \theta=0.5$;
- (2)在INMF中, $\alpha=0.9, \beta=0.1, r=25$;
- (3)在ONMFO中, $\lambda=0.01, r=25$;
- (4)在L_p-INMF中, $p=0.5, \lambda=0.1, r=25$ 。

实验结果如图6所示,OLNMF的基矩阵 W_{k+1} 的平均稀疏度依然是最高的,这说明算法具有不错的鲁棒性。

图6 ORL数据集中4种在线NMF的 W_{k+1} 的平均稀疏度曲线

4.3 分类实验

4.3.1 EEG数据集

EEG数据集^[24]是来自2003年BCI比赛的其中一组左手和右手的想象运动脑电信号数据集,一共有140次实验,数据集从C3, Cz, C4通道中采集。本次实验我们只选取C3, C4通道采集到的信号。由于EEG数据集是脑电信号,数据集会有负值出现,因此需要对EEG数据集进行CSP预处理^[25],将脑电信号转换成非负数据。经过处理后,非负数据 $V \in R^{140 \times 440 \times 2}$ 是一个两通道,也是140次实验的数据集。

本次实验在初始化步骤中,取EEG数据集的前40段脑电信号进行批处理NMF,得到基矩阵 W_{k+1} 和系数矩阵 H_{k+1} ,对系数矩阵 H_{k+1} 进行SVM训练得到SVM分类器(svmtrain);在在线学习步骤中,对剩下的100段脑电信号进行了在线NMF处理(OLNMF, INMF, ONMFO, L_p-INMF),每次获得一个新的 h_{k+1} ,并代入到SVM分类器(svmtrain)中分类;最后计算4种在线NMF的分类准确率。上述4种在线NMF的参数设定如下:

- (1)在OLNMF中, $\alpha=0.9, \beta=0.1, r=2, s=10, \theta=0.5$;
- (2)在INMF中, $\alpha=0.9, \beta=0.1, r=2$;
- (3)在ONMFO中, $\lambda=0.01, r=2$;
- (4)在L_p-INMF中, $p=0.5, \lambda=0.1, r=2$ 。

如图7所示,左手和右手想象运动基本被分离曲线分离在曲线两侧,分类效果很显著。对该实验反复运行5次,记录其分类准确率和实验时间,并计算其平均分类准确率和平均实验时间。从表1可以看出:在C3和C4通道中,在线学习非负矩阵分解的分类准确率是最高的。由此得出,OLNMF得到特征比较明显的系数向量 h_{k+1} 。如表2所列,OLNMF的实验时间大约是另外3种在线NMF的3倍。对于速度方面的性能,OLNMF不占优势。

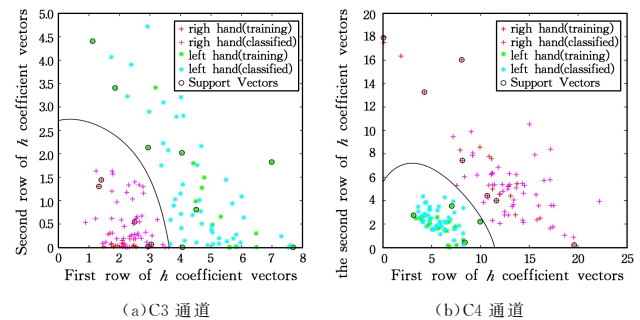
图7 在EEG数据集C3, C4通道下,OLNMF的系数矩阵 h_{k+1} 的分布

表1 4种在线NMF结合SVM分类在EEG数据集的分类准确率(单位:%)

实验次数	通道	算法	OLNMF	INMF	ONMFO	L _{1/2} -INMF
1	C3		95	91	82	90
	C4		96	93	71	91
2	C3		94	92	83	89
	C4		95	93	72	91
3	C3		95	91	81	89
	C4		95	93	72	90
4	C3		95	92	82	89
	C4		95	93	72	91
5	C3		95	92	81	90
	C4		96	93	73	91
平均准确率	C3		94.8	91.6	81.8	89.6
	C4		95.4	93.0	72.0	90.2

表2 4种在线NMF结合SVM分类在EEG数据集的实验时间
(单位:s)

实验次数	通道算法	OLNMF	INMF	ONMFO	$L_{1/2}$ -INMF
1	C3	3.3260	0.8349	0.9480	0.8120
	C4	3.3520	0.9000	1.0280	0.8650
2	C3	3.3580	0.8680	1.1640	0.9110
	C4	3.4290	0.8960	1.1230	0.9480
3	C3	3.4230	0.9480	1.1790	1.0220
	C4	3.4980	1.0010	1.2130	1.2360
4	C3	3.6130	1.0580	1.2560	1.1000
	C4	4.5800	1.0510	1.5490	1.3120
5	C3	3.6130	1.1330	1.3750	1.2170
	C4	4.6190	1.3700	1.6000	1.3900
平均时间	C3	3.4666	0.9682	1.1944	1.0124
	C4	3.8956	1.0436	1.3026	1.1082

结束语 在线学习非负矩阵分解能在线处理大型的实时更新的数据集,并分解得到稀疏性较高的基矩阵。3种实验分别验证了:1)在不同参数 θ 下,OLNMF算法依然可以分解得到稳定的基矩阵。2)与INMF,ONMFO, $L_{1/2}$ -INMF相比,OLNMF算法得到的基矩阵有更高的稀疏度。3)与INMF,ONMFO, $L_{1/2}$ -INMF相比,OLNMF算法在EEG脑电信号数据集有不错的分类准确率。

参考文献

- [1] JOLLIFFE I T. Principal Component Analysis [J]. Journal of Marketing Research, 2002, 25(4): 513.
- [2] Gersho A, Gray R M. Vector quantization and signal compression [J]. Springer International, 1992, 159(1): 407-485.
- [3] LEE D D, SEUNG H S. Learning the parts of objects by non-negative matrix factorization [J]. Nature, 2001, 401(6755): 788-791.
- [4] CHEN W S, ZHAO Y, PAN B, et al. Supervised kernel nonnegative matrix factorization for face recognition [J]. Neurocomputing, 2016, 205(C): 165-181.
- [5] KITAMURA K, BANDO Y, ITOYAMA K, et al. Student's t multichannel nonnegative matrix factorization for blind source separation [C] // IEEE International Workshop on Acoustic Signal Enhancement. IEEE, 2016: 1-5.
- [6] KANG T G, KWON K, SHIN J W, et al. NMF-based Target Source Separation Using Deep Neural Network [J]. IEEE Signal Processing Letters, 2015, 22(2): 229-233.
- [7] FÉVOTTE C, DOBIGEON N. Nonlinear hyperspectral unmixing with robust nonnegative matrix factorization [J]. IEEE Trans. Image Processing A Publication of the IEEE Signal Processing Society, 2015, 24(12): 4810.
- [8] GILLIS N, KUANG D, PARK H. Hierarchical Clustering of Hyperspectral Images Using Rank-Two Nonnegative Matrix Factorization [J]. IEEE Transactions on Geoscience & Remote Sensing, 2013, 53(4): 2066-2078.
- [9] YOKOYA N, CHANUSSOT J, IWASAKI A. Nonlinear Unmixing of Hyperspectral Data Using Semi-Nonnegative Matrix Factorization [J]. IEEE Transactions on Geoscience & Remote Sensing, 2014, 52(2): 1430-1437.
- [10] PASCUAL-MONTANO A, CARAZO J M, KOCHI K, et al. Nonsmooth nonnegative matrix factorization (nsNMF) [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2006, 28(3): 403-415.
- [11] BUCAK S S, GUNSEL B. Incremental subspace learning via non-negative matrix factorization [J]. Pattern Recognition, 2009, 42(5): 788-797.
- [12] RAJABI R, GHASSEMIAN H. Sparsity Constrained Graph Regularized NMF for Spectral Unmixing of Hyperspectral Data [J]. Journal of the Indian Society of Remote Sensing, 2014, 43(2): 269-278.
- [13] ZHAO R, TAN V Y F. Online nonnegative matrix factorization with outliers [C] // IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2016: 1-1.
- [14] YANG H L, HE G P. Online Face Recognition Algorithm via Nonnegative Matrix Factorization [J]. Information Technology Journal, 2010, 9(8): 1719-1724.
- [15] DANG S, CUI Z, CAO Z, et al. SAR target recognition via incremental nonnegative matrix factorization with Lp sparse constraint [J]. Science Technology and Engineering, 2017, 10(3): 0530-0534.
- [16] WU Y, SHEN B, LING H. Visual Tracking via Online Nonnegative Matrix Factorization [J]. IEEE Transactions on Circuits & Systems for Video Technology, 2014, 24(3): 374-383.
- [17] CHEN R, LI H. Online algorithm for foreground detection based on incremental nonnegative matrix factorization [C] // International Conference on Control, Automation and Robotics. IEEE, 2016.
- [18] WENG J, ZHANG Y, HWANG W S. Candid Covariance-Free Incremental Principal Component Analysis [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2003, 25(8): 1034-1040.
- [19] LEE D D, SEUNG H S. Algorithms for non-negative matrix factorization. In NIPS [J]. Advances in Neural Information Processing Systems, 2000, 13(6): 556-562.
- [20] KIVINEN J, WARMUTH M K. Additive versus exponentiated gradient updates for linear prediction [C] // Twenty-Seventh ACM Symposium on Theory of Computing. Las Vegas, Nevada, USA, DBLP, 1995: 209-218.
- [21] AMARI S I. Natural Gradient Works Efficiently in Learning [M] // Unsupervised Learning. Bradford Company, 1999: 251-276.
- [22] HOYER P O. Non-negative sparse coding [C] // IEEE Workshop on Neural Networks for Signal Processing. IEEE, 2002: 557-565.
- [23] HOYER P O. Non-negative Matrix Factorization with Sparseness Constraints [J]. Journal of Machine Learning Research, 2004, 5(1): 1457-1469.
- [24] BLANKERTZ B, MÜLLER K R, CURIO G, et al. The BCI Competition 2003: progress and perspectives in detection and discrimination of EEG single trials [J]. IEEE Transactions on Bio-medical Engineering, 2004, 51(6): 1044.
- [25] LAHIRI R, RAKSHIT A, KONAR A. Discriminating Motor Imagery EEG signals using an improvised regularised CSP algorithm [C] // IEEE International Conference on Power Electronics, Intelligent Control and Energy Systems. IEEE, 2017: 1-6.