Neurocomputing 000 (2018) 1–9

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Incremental general non-negative matrix factorization without dimension matching constraints

Zigang Chen[a,b], Lixiang Li[a,*], Haipeng Peng[a], Yuhong Liu[c], Yixian Yang[a]

[a] Information Security Center, State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, P.O. Box 145, Beijing 100876, China
[b] College of Computer and Information Engineering, Nanyang Institute of Technology, Nanyang 473004, China
[c] Department of Computer Engineering, Santa Clara University, Santa Clara, CA 95053, USA

ARTICLE INFO

Article history:
Received 15 October 2017
Revised 2 April 2018
Accepted 21 May 2018
Available online xxx

Communicated by Deng Cai

Keywords:
General non-negative matrix factorization
Incremental general non-negative matrix factorization
Semi-Tensor Product (STP)
Dimensionality reduction

ABSTRACT

In this paper, we propose a General Non-negative Matrix Factorization based on the left Semi-Tensor Product (lGNMF) and the General Non-negative Matrix Factorization based on the right Semi-Tensor Product (rGNMF), which factorize an input non-negative matrix into two non-negative matrices of lower ranks based on gradient method. In particular, the proposed models are able to remove the dimension matching constraints required by conventional NMF models. Both theoretical derivation and experimental results show that the conventional NMF is a special case of the proposed lGNMF and rGNMF. We find the method for the best efficacy of the image restoration in lGNMF and rGNMF by experiments on baboon and lenna images. Moreover, inspired by the Incremental Non-negative Matrix Factorization (INMF), we propose the Incremental lGNMF (IlGNMF) and Incremental rGNMF (IrGNMF), We also conduct the experiments on JAFFE database and ORL database, and find that IlGNMF and IrGNMF realize saving storage space and reducing computation time in incremental facial training.

1. Introduction

Face recognition technology has attracted many attentions due to its broad applications on security systems, user authentications, smart phone unlocking, etc. With the continuous improvement of image resolution and the prevalence of web cameras and smart phones, a vast amount of high resolution facial images has been generated every day, resulting in massive storage space and high computational complexity. There is an increasing demand for facial recognition models that can dynamically adopt large scale data through online training while requiring less storage space and computational time.

There is psychological [1] and physiological [2,3] evidences for parts-based representations in the brain, and certain computational theories of object recognition are relying on such representations [4,5]. But little is known about how brains or computers might learn the parts of objects. Non-negative matrix factorization (NMF) [6], which factorizes an input non-negative matrix into two non-negative matrices of lower ranks and is able to learn parts of faces, has recently been adopted by many face recognition studies [7,8]. However, conventional NMF are not specifically designed to reduce storage space or computational complexity. In addition, it requires lots of computational efforts for model updates, making it very challenging to dynamically adopt new data samples through online training. Bucak and Gunsel [9] propose the incremental non-negative matrix factorization (INMF) method to overcome the difficulties that conventional NMF confronts in online processing of large scale data.

Many later studies based on INMF are then proposed to further reduce the required storage space and computational complexity in various applications. For example, Zheng et al. [10] propose an incremental locality preserving nonnegative matrix factorization (ILPNMF) method to discover the manifold structure embedded in high-dimensional space that deals well with large scale data. Liu et al. [11] propose an online graph regularized non-negative matrix factorization for large-scale datasets. Yu et al. [12] propose an incremental graph regularized nonnegative matrix factorization (IGNMF) algorithm which imposes manifold into INMF to preserve the geometric structure in the data under incremental study framework, and proposed Batch-IGNMF algorithms (B-IGNMF) for implementing incremental study in batches. Zhou et al. [13] derive an INMF with volume constraints for solving online Blind Source Separation (BSS).

Recently, online analysis or learning approaches about INMF had emerged. For example the incremental learning method. Rebhan et al. [14] propose an incremental learning method to

* Corresponding author.
E-mail address: li_lixiang2006@163.com (L. Li).

https://doi.org/10.1016/j.neucom.2018.05.067
0925-2312/

cover the (possibly growing) input space and to enable NMF to incrementally and continuously adopt new data. Amin and Mahmoudi [15] use INMF to learn a linear part-based subspace in an online fashion. Wang and Lu [16] propose an incremental orthogonal projective non-negative matrix factorization algorithm (IOP-NMF) to learn a parts-based subspace that reveals dynamic data streams.

INMF has also been used for visual tracking (object tracking). Qian et al. [17] propose an appearance model based on extended INMF for visual tracking. Dou et al. [18] propose an incremental discriminative projective non-negative matrix factorization for robust visual tracking. Zhang et al. [19] propose a constrained INMF for visual tracking. Liu et al. [20] propose an incremental robust nonnegative matrix factorization (IRNMF) for object tracking. Dang et al. [21] propose the INMF with L-p sparse constraint for SAR target recognition.

Inspired by INMF, we aim to propose an incremental general non-negative matrix factorization to further save storage space and improve computing performance.

Our contributions are as follows:

1. To save storage space, we propose the General Non-negative Matrix Factorization based on the left Semi-Tensor Product (lGNMF) and the General Non-negative Matrix Factorization based on the right Semi-Tensor Product (rGNMF) to factorize a matrix $C \in R_+^{s\times t}$ into basis images (or basis matrix) $A \in R_+^{m\times n}$ and coefficient matrix $B \in R_+^{p\times q}$, where, $m = s/(l/n)$ and $q = t/(l/p)$, the variable l is the least common product of n and p.
2. Experiments on both baboon and lenna images are conducted to analyze the performance of the proposed lGNMF and rGNMF.
3. Inspired by INMF, we further propose the Incremental lGNMF (IlGNMF) and Incremental rGNMF (IrGNMF), with the design goal as to make the above proposed lGNMF and rGNMF more suitable for dynamically adopting new data for online learning.
4. Experiments on JAFFE database and ORL database are conducted to analyze the performance of the proposed IlGNMF and IrGNMF in incremental facial training. Experiment results show that compared to INMF, the proposed IlGNMF and IrGNMF can further reduce storage space and computation time for incremental facial training.

The rest of this paper is organized as follows. Section 2 introduces conventional NMF, INMF and semi-tensor product of matrices. Section 3 presents lGNMF, rGNMF, IlGNMF and IrGNMF. Experiments on the performance of the proposed lGNMF and rGNMF are presented in Section 4. Experiments on the performance of the proposed IlGNMF and IrGNMF are presented in Section 5. Section 6 concludes the paper.

2. Preliminaries

2.1. Non-negative matrix factorization

NMF factorizes an input non-negative matrix $C \in R^{s\times t}$ into two non-negative matrices of lower ranks, which are $A \in R^{s\times p}$ and $B \in R^{p\times t}$, such that

$$C_+^{s\times t} \approx A_+^{s\times p} B_+^{p\times t}. \quad (1)$$

In general, matrix A is known as the basis matrix, and matrix B is known as the coefficient matrix. Obviously, matrices A and B should meet the dimension matching condition that matrix A's column number should equal to matrix B's row number.

Generally, the loss function of NMF can be calculated by the Euclidean distance or the Kullback–Leibler divergence. When the loss function is determined by the Euclidean distance, the optimization function of NMF is given by

$$L_{NMF}(A, B) = \|C - AB\|_F^2 = \sum_{i=1}^{s}\sum_{j=1}^{t}[C_{ij} - (AB)_{ij}]^2. \quad (2)$$

2.2. Incremental non-negative matrix factorization

INMF [9] is proposed to make the online updating of NMF much more efficient when new data are available. Specifically, the sample set C_k is factorized into the basis matrix A_k and the coefficient matrix B_k using NMF, where k is the number of the current existence sample set. Meanwhile, the optimization function of NMF is given by

$$L_{NMF}(A_k, B_k) = \|C_k - A_kB_k\|_F^2 = \sum_{i=1}^{s}\sum_{j=1}^{t}[(C_k)_{ij} - (A_kB_k)_{ij}]^2. \quad (3)$$

When a new sample c_{k+1} is added, the sample set becomes C_{k+1}, at this moment, the optimization function of incremental NMF (INMF) is given by

$$\begin{aligned} L_{INMF}(A_{k+1}, B_{k+1}) &= \|C_{k+1} - A_{k+1}B_{k+1}\|_F^2 \\ &\approx \sum_{i=1}^{s}\sum_{j=1}^{t}[(C_k)_{ij} - (A_kB_k)_{ij}]^2 + \sum_{i=1}^{s}[(c_{k+1})_i - (A_{k+1}b_{k+1})_i]^2 \\ &= L_{NMF}(A_k, B_k) + L_{NMF}(A_{k+1}, b_{k+1}). \end{aligned} \quad (4)$$

When INMF is applied in the pattern recognition process, the change of basis matrix is not good for pattern recognition. Thus, when a new sample c_{k+1} is added, b_{k+1} can be obtained by

$$b_{k+1} = ((A_k^TA_k)^{-1}A_k^T)c_{k+1}, \quad (5)$$

where A_k^T is used to respect transpose of matrix A_k, and $(A_k^TA_k)^{-1}$ is used to respect inverse of matrix $(A_k^TA_k)$.

2.3. Semi-Tensor Product of matrices

In this section, we provide some necessary preliminaries on the Semi-Tensor Product (STP) [22–24] of matrices.

Definition 1. Given two matrices $A \in R^{m\times n}$ and $B \in R^{p\times q}$, the variable l is the least common product of n and p. The left STP denoted by $\ltimes$, and

$$A \ltimes B = (A \otimes I_{l/n})(B \otimes I_{l/p}) \in R^{(m\cdot l/n)\times(l/p\cdot q)}, \quad (6)$$

where $\otimes$ represents the right Kronecker product [25].

Definition 2. Given two matrices $A \in R^{m\times n}$ and $B \in R^{p\times q}$, the variable l is the least common product of n and p. The right STP denoted by $\rtimes$, and

$$A \rtimes B = (I_{l/n} \otimes A)(I_{l/p} \otimes B) \in R^{(m\cdot l/n)\times(l/p\cdot q)}, \quad (7)$$

Here are a few simple examples:

Example 1. Let $A = (4 \quad -3 \quad 2)$, $B = \begin{pmatrix} -1 \\ 2 \end{pmatrix}$, then

$$\begin{aligned} A \ltimes B &= \left((4 \quad -3 \quad 2) \otimes I_2\right)\left(\begin{pmatrix} -1 \\ 2 \end{pmatrix} \otimes I_3\right) \\ &= \begin{pmatrix} 4 & 0 & -3 & 0 & 2 & 0 \\ 0 & 4 & 0 & -3 & 0 & 2 \end{pmatrix}\begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \\ 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix} \\ &= \begin{pmatrix} -4 & 4 & 3 \\ -6 & -4 & 4 \end{pmatrix}. \end{aligned}$$