

Efficient Data Representation Combining with ELM and GNMF

Zhiyong Zeng, YunLiang Jiang, Yong Liu and Weicong Liu

Abstract Nonnegative Matrix Factorization (NMF) is a powerful data representation method, which has been applied in many applications such as dimension reduction, data clustering etc. As the process of NMF needs huge computation cost, especially when the dimensional of data is large. Thus a ELM feature mapping based NMF is proposed [1], which combined Extreme Learning Machine (ELM) feature mapping with NMF (EFM NMF), can reduce the computational of NMF. However, the random parameter generating based ELM feature mapping is nonlinear. And this will lower the representation ability of the subspace generated by NMF without sufficiently constrains. In order to solve this problem, this chapter propose a novel method named Extreme Learning Machine feature mapping based graph regulated NMF (EFM GNMF), which combines ELM feature mapping with Graph Regularized Nonnegative Matrix Factorization (GNMF). Experiments on the COIL20 image library, the CMU PIE face database and TDT2 corpus show the efficiency of the proposed method.

Keywords Extreme learning machine · ELM feature mapping · Nonnegative matrix factorization · Graph regularized nonnegative matrix factorization · EFM NMF · EFM GNMF · Data representation

Z. Zeng (✉)
Hangzhou Dianzi University, Hangzhou 310018, China
e-mail: zzy07053437@163.com

Y. Jiang
Huzhou Teachers College, Huzhou 313000, China

Y. Liu (✉) · W. Liu
Zhejiang University, Hangzhou 310027, China
e-mail: yongliu@iipc.zju.edu.cn

1 Introduction

Nonnegative matrix factorization (NMF) techniques have been frequently applied in data representation and document clustering. Given an input data matrix X , each column of which represents a sample, NMF aims to find two factor matrices U and V using low-rank approximation such that $X \approx UV$. Each column of U represents a base vector, and each column of V describes how these base vectors are combined fractionally to form the corresponding sample in X [2, 3].

Compared to other methods, such as principal component analysis (PCA) and independent component analysis (ICA), the nonnegative constraints lead to a parts-based representation because they allow only additive, not subtractive, combinations. Such a representation encodes the data using few active components, which makes the basis easy to interpret. NMF has been shown to be superior to SVD in face recognition [4] and document clustering [5]. It is optimal for learning the parts of objects.

However, NMF cost huge computing when it disposes high-dimensional data such as image data. ELM feature mapping [6, 7] as an explicit feature mapping techniques was proposed. It is more convenient than kernel function and can get more satisfactory results for classification and regression [8, 9]. NMF is a linear model, using nonlinear feature mapping techniques, it will be able to deal with nonlinear correlation in data. Then, ELM based methods is not sensitive to the number of hidden layer nodes, provided that a large enough number is selected [1]. So, using ELM feature mapping to improve the efficiency of NMF is feasible. Nevertheless, ELM feature mapping NMF (EFM NMF) can not keep generalization performance as NMF. Only the non-negative constraints in NMF unlike other subspace methods (e.g. Locality Preserving Projections (LPP) method [10]), may not be sufficiently understand the hidden structure of the space which transform from the original data. A wide variety of subspace constraints can be formulated into a certain form such as PCA and LPP to enforce general subspace constraints into NMF. Graph Regularized Nonnegative Matrix Factorization (GNMF [11], which discovers the intrinsic geometrical and discriminating structure of the data space by implant a geometrical regularization, is more powerful than the ordinary NMF approach. In order to obtain efficiency and keep generalization representation performance simultaneously, we proposed method named EFM GNMF which combined ELM feature mapping with GNMF.

The rest of the chapter is organized as follows: Sect. 2 gives a brief review of the ELM, ELM Feature mapping, NMF and GNMF. The EFM NMF and EFM GNMF are presented in Sect. 3. The experimental result will be shown in Sect. 4. Finally, in Sect. 5, we conclude the chapter.

2 A Review of Related Work

In this section, a short review of the original ELM algorithm, ELM Feature mapping, NMF and GNMF are given.

2.1 ELM

For N arbitrary distinct samples (x_i, t_i) , where $x_i = [x_{i1}, x_{i2}, \dots, x_{iD}]^T \in R^D$ and $t_i = [t_{i1}, t_{i2}, \dots, t_{iK}]^T \in R^K$, standard SLFNs with L hidden nodes and activation function $h(x)$ are mathematically modeled as:

$$\sum_{i=1}^L \beta_i h_i(x_j) = \sum_{i=1}^L \beta_i h_i(w_i \cdot x_j + b_i) = o_j \quad (1)$$

where $j = 1, 2, \dots, N$. Here $w_i = [w_{i1}, w_{i2}, \dots, w_{iD}]^T$ is the weight vector connecting the i th hidden node and the input nodes, $\beta_i = [\beta_{i1}, \dots, \beta_{iK}]^T$ is the weight vector connecting the i th hidden node and the output nodes, and b_i is the threshold of the i th hidden node. The standard SLFNs with L hidden nodes with activation function $h(x)$ can be compactly written as [12–15]:

$$H\beta = T \quad (2)$$

where

$$H = \begin{bmatrix} h_1(w_1 \cdot x_1 + b_1) & \dots & h_L(w_L \cdot x_1 + b_L) \\ \vdots & & \vdots \\ h_1(w_1 \cdot x_N + b_1) & \dots & h_L(w_L \cdot x_N + b_L) \end{bmatrix} \quad (3)$$

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_L^T \end{bmatrix} \text{ and } T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix} \quad (4)$$

Different from the conventional gradient-based solution of SLFNs, ELM simply solves the function by

$$\beta = H^+ T \quad (5)$$

H^+ is the Moore-Penrose generalized inverse of matrix H .

2.2 ELM Feature Mapping

As show in Sect. 2.1 above, $h(x)$ as the ELM feature mapping, maps the sample x_1 from the D -dimensional input space to the L -dimensional hidden-layer feature space which is called ELM feature space. The ELM feature mapping process is shown in Fig. 1.

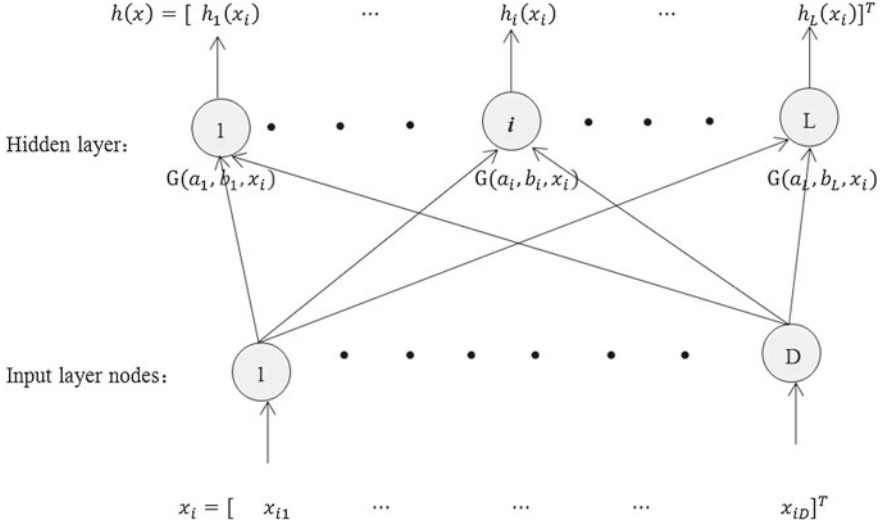


Fig. 1 ELM feature mapping process (cited from [1])

The ELM feature mapping can be formally described as:

$$h(x_i) = [h_1(x_i), \dots, h_L(x_i)]^T = [G(a_1, b_1, x_i), \dots, G(a_L, b_L, x_i)]^T \quad (6)$$

where $G(a_i, b_i, x_i)$ is the output of the i -th hidden node. The parameters which need not to be tuned, $(a_i, b_i)_{i=1}^L$, can be randomly generated according to any continuous probability distribution. It is that ELM feature mapping is very convenient. Huang in [6, 7] has proved that almost all almost all nonlinear piecewise continuous functions can be used as the hidden-node output functions directly [1].

2.3 GNMF

NMF [16–18] is a matrix factorization algorithm that focuses on the analysis of data matrices whose elements are nonnegative. Consider a data matrix $X = [x_1, \dots, x_D] \in \mathbb{R}^{D \times M}$ each column of X is a sample vector which consists of D features. Generally, NMF can be presented as the following optimization problem:

$$C(X \approx UV), \text{ s.t. } U, V \geq 0 \quad (7)$$

NMF aims to find two non-negative matrices $U = [u_{ij}] \in \mathbb{R}^{D \times K}$ and $V = [v_{ij}] \in \mathbb{R}^{K \times M}$ whose product can well approximate the original matrix X . $C(\cdot)$ denotes the cost function.

NMF performs the learning in the Euclidean space which cover the intrinsic geometrical and discriminating. To find a compact representation which uncovers the hidden semantics and simultaneously respects the intrinsic geometric structure, Cai et al. [11] proposed construct an affinity graph to encode the information and seek a matrix factorization to respects the graph structure in GNMF.

$$O_{GNMF} = \|X - UV\|_F^2 + \lambda \text{tr} \left(V^T L V \right) \quad \text{st. } U \geq 0, V \geq 0 \quad (8)$$

where L is graph Laplacian. The adjacent graph, which each vertex corresponding to a sample and the weight between vertex $\vec{x}_i$ and vertex $\vec{x}_j$, is defined as [19]

$$S_{ij} = \begin{cases} 1, & \text{if } \vec{x}_i \in N_k(\vec{x}_j) \text{ or } \vec{x}_j \in N_k(\vec{x}_i) \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

where $N_k(\vec{x}_i)$ signifies the set of k nearest neighbors of $\vec{x}_i$. Then L is written as $L = T - W$, where T is a diagonal matrix whose diagonal entries are column sums of S , i.e., $T_{ii} = \sum_j W_{ij}$.

3 EFM GNMF

In this section, we will present our EFM GNMF. EFM NMF will improve computational efficiency by reducing the feature number. But ELM feature mapping, which using random parameter, is a nonlinear feature mapping technique. This will lower the ability of representation of the subspace generating from NMF without sufficiently constrains. In order to solve this problem, this chapter propose a novel method EFM GNMF, combined ELM feature mapping with Graph Regularized Non-negative Matrix Factorization (GNMF). Graph constrain guarantee that using ELM feature space in NMF can also has the local manifold feature. The proposed algorithm puts as follows:

- (1) Setting the number of hidden-layer nodes $L < D$ and threshold $\varepsilon > 0$.
- (2) Calculate the weight matrix W on the nearest neighbor graph of the original data.
- (3) Using ELM feature mapping $h(x) = [h_1(x), \dots, h_i(x), \dots, h_L(x)]^T$ transform the original data into ELM feature space. The original data with D -dimensional will transform into L -dimensional

$$X = [x_1, \dots, x_D] \in \mathbb{R}^{D \times M} \rightarrow H = [h_1, \dots, h_L] \in \mathbb{R}^{L \times M}$$

- (4) Initialize $U \in \mathbb{R}^{L \times K}$, $V \in \mathbb{R}^{K \times M}$ and the regularization λ with nonnegative values.
- (5) Using W as the weight matrix on the nearest neighbor graph of the ELM feature space data H
- (6) Iterate for each i, j , and i until convergence ($\text{err} < \varepsilon$) or reached the maxiamal iterations [10]

$$\begin{aligned} (a) \quad U_{ij}^{t+1} &\leftarrow U_{ij}^t \frac{(H V^T)_{ij}}{(U^t V V^T)_{ij}} \\ (b) \quad V_{ij}^{t+1} &\leftarrow V_{ij}^t \frac{(H^T U + \lambda W V^T)_{ij}}{((U^t V V^T)^T + \lambda D V^T)_{ij}} \\ (c) \quad \text{err} &\leftarrow \max \left\{ \frac{\|U^{t+1} - U^t\|}{\sqrt{LK}}, \frac{\|V^{t+1} - V^t\|}{\sqrt{KM}} \right\} \end{aligned}$$

Table 1 Statistics of the three data sets

Datasets	Size (N)	Dimensionality (M)	# of classes (K)
COIL20	1440	1024	20
PIE	2856	1024	68
TDT2	9394	36771	30

4 Experiments Results

In this section, three of the mostly used datasets COIL20 image library, the CMU PIE face database and TDT2 corpus are used to prove the efficiency of the proposed algorithm. The important statistics of these data sets are summarized below (see Table 1). To make the results valid, every algorithm run 20 times on each data set and obtains the average result. This chapter chooses the sigmoid function as the ELM feature mapping activation function for it is most used. To obtain the efficiency and performance of these algorithms with different numbers of hidden nodes, we adopt the integrated data sets. K-mean is used as the cluster to test the generalization performance. The clustering result is evaluated by comparing the obtained label of each sample with the label provided by the data set. Two metrics, the accuracy (AC) and the normalized mutual information metric (NMI) are used to measure the clustering performance [11]. Please see [20] for the detailed definitions of these two metrics. All the algorithms are carried out in MATLAB 2011 environment running in a Core 2, 2.5 GHZ CPU.

4.1 Compared Algorithms

To demonstrate how the efficiency of NMF can be improve by our method, we compare the computing time of four algorithms (NMF, GNMFEFM NMF, EFM GNMFF). The hidden nodes number is set as 1, 2, 4, 6, $\dots$, 18 within 18; 20, 30, $\dots$, 100 from 20 to 100; 125, 150, $\dots$, 600 from 125 to 600; 650, 700, $\dots$, 1000 from 600 to 1000. Comparing the clustering performance of these methods is also revealed (The values of clustering performance change little when nodes number surpass 100, that is, only the result of the hidden nodes number from 1 to 100 is shown). The max iterations in NMF, GNMFF, EFM NMF and EFM GNMFF are 100.

Figure 2 show the time comparing results on the COIL20, PIE, and TDT2 data sets respectively. Over all, we can see that ELM feature mapping methods (EFM NMF, EFM GNMFF) is faster than NMF and GNMFF when hidden nodes number is low. With the hidden nodes number increased, the computation time is monotone increasing. When the number is high, the computation time of EFM NMF or EFM NMF will exceed NMF and GNMFF. Comparing the computation time of EFM NMF with EFM GNMFF, we can see that EFM NMF is faster than EFM GNMFF. However, by increasing the hidden nodes number, the time difference between EFM NMF

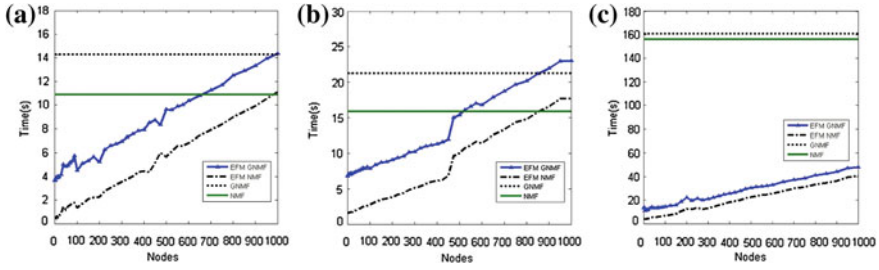


Fig. 2 Computation time on **a** COIL20 **b** PIE **c** TDT2

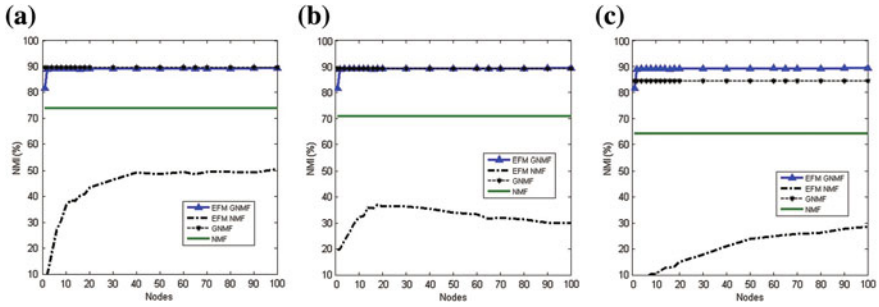


Fig. 3 NMI measure clustering performance. **a** COIL20 **b** PIE **c** TDT2

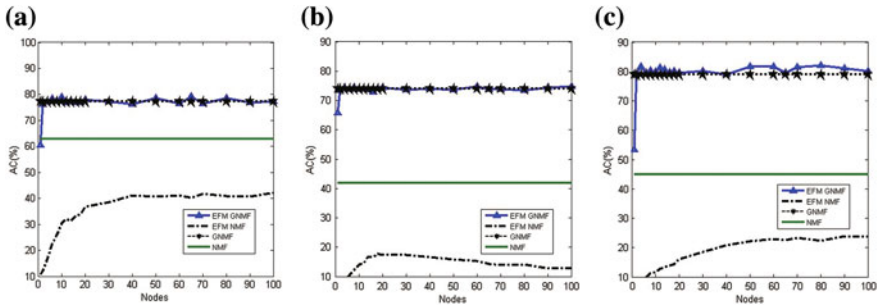


Fig. 4 AC measure clustering performance. **a** COIL20 **b** PIE **c** TDT2

and EFM GNMF close to a constant. That is because GNMF need to compute the weight matrix W .

Figures 3 and 4 show clustering performance comparing results on data sets respectively. Obviously, EFM NMF can not get the approximate clustering performance as NMF. Nevertheless, EFM GNMF can reach approximate clustering performance as GNMF, provided that a large enough hidden nodes number is selected.

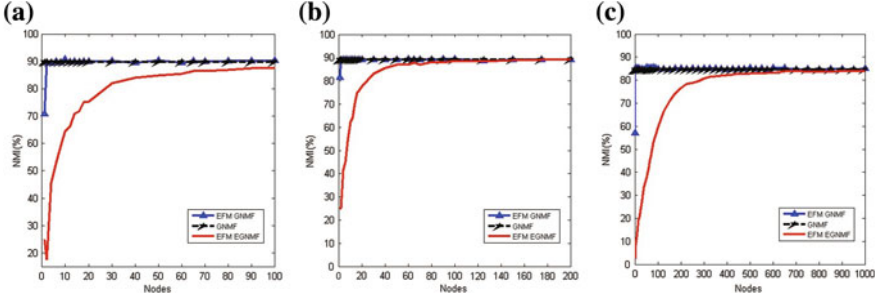


Fig. 5 NMI measure clustering performance. Comparing EFM GNMF with EFM EGNMF
a COIL20 **b** PIE **c** TDT2

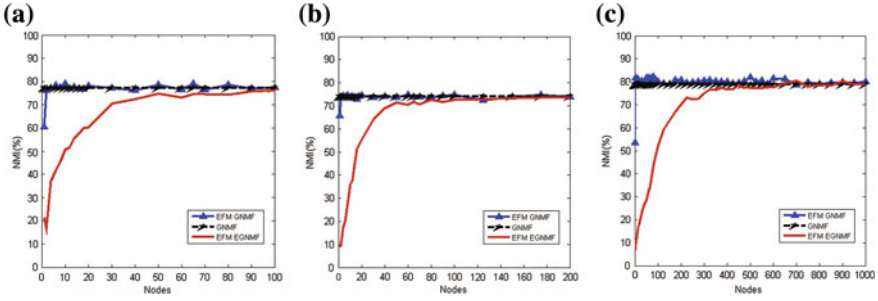


Fig. 6 AC measure clustering performance. Comparing EFM GNMF with EFM EGNMF
a COIL20 **b** PIE **c** TDT2

4.2 Original Graph Versus ELM Feature Space Graph

We denote the method that uses ELM feature space neighbor graph to replace the original space neighbor graph in the EFM GNMF as ELM feature mapping with ELM space graph NMF (EFM EGNMF).

As show in Figs. 5 and 6, EFM EGNMF can also reach similar clustering performance as GNMF. However, comparing with EFM GNMF, EFM EGNMF need more hidden nodes number to reach similar clustering performance as GNMF. So, ELM feature mapping may be can simulate the local manifold of the original data.

4.3 The Geometric Structure of ELM Feature Space

Prompt by Sect. 4.2, we speculate that ELM feature mapping can keep approximated geometric of original data when transforming the original data space into ELM feature space with a large number of hidden nodes. In order to discover whether

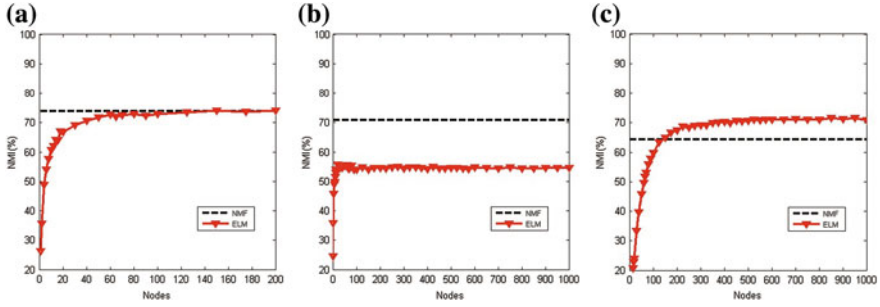


Fig. 7 NMI measure clustering performance. Comparing ELM with NMF **a** COIL20 **b** PIE **c** TDT2

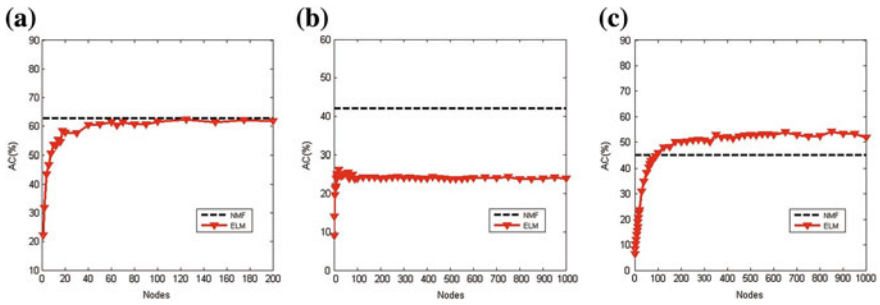


Fig. 8 AC measure clustering performance. Comparing ELM with NMF **a** COIL20 **b** PIE **c** TDT2

ELM can keep approximated geometric structure of original data, we compare the cluster performance of ELM with NMF under different hidden nodes number.

As showed in Figs. 7a, c and 8a, c, after transform into ELM feature space, the data can reach similar clustering performance as NMF, provided the hidden nodes number is enough. Figure 8b Even the hidden nodes number is huge, the data has an approximate constant gap with NMF in clustering performance. We can find that the number of samples for each class is 72 in COIL20 data set, 42 in PIE data set, 313 in TDT2 data set. So, for ELM feature mapping, it may be that having more samples for each class can get better performance. ELM feature mapping can keep approximated geometric of original data not only need enough hidden nodes, but also need enough samples for each class. This need more experiments to confirm.

4.4 Combining ELM and NMF with Other Constrains

In this chapter, neighbor graph based constrain has been proved powerful. Then, NMF can combine with a wide variety of subspace constraints that can be formulated into a certain form such as PCA and LPP. ELM feature mapping combined with general subspace constrained NMF(GSC NMF) can be the future work.

5 Conclusions

This chapter proposes a new method named EFM GNMF, which applies ELM feature mapping and graph constraints to solve computational problem in NMF without lose generalization performance. Experiments show that when dispose with high-dimensional data, the efficiency of EFM GNMF is better than directly using NMF or GNMF. Also, EFM GNMF is compared with GNMF in clustering performance. Unlike EFM NMF get efficiency without keep generalization performance, EFM GNMF can reach similar result as GNMF. Moreover, the difference of using the neighbor graph of the original data space with ELM feature space is raised. ELM feature mapping can keep approximated geometric structure hidden in the original data.

Acknowledgments We want to thank Dr. Huang Guangbin from NTU and Dr. Jin Xin from Chinese Academy of Sciences. They provide us with some codes and details of Extreme Learning Machine. This work was supported by the National Natural Science Foundation Project of China (61173123) and the Natural Science Foundation Project of Zhejiang Province (LR13F030003).

References

1. Q. He, X. Jin, C. Du, F. Zhuang, and Z. Shi, Clustering in extreme learning machine feature space, *Neurocomputing* (2013)
2. Z. Li, X. Wu, H. Peng, Nonnegative matrix factorization on orthogonal subspace. *Pattern Recogn. Lett.* **31**(9), 905–911 (2010)
3. S. B. S. L, Nonnegative matrix factorization clustering on multiple manifolds, in *AAAI* (2010)
4. S. Li, X. Hou, H. Zhang, Q. Cheng, Learning spatially localized, parts-based representation, *Computer Vision and Pattern Recognition*, 2001. CVPR 2001. in *Proceedings of the 2001 IEEE Computer Society Conference*, vol. 1, pp. 207–212, 2001
5. W. Xu, X. Liu, Y. Gong, Document clustering based on non-negative matrix factorization, in *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pp. 267–273, 2003
6. G.-B. Huang, L. Chen, Convex ncremental extreme learning machine. *Neurocomputing* **70**, 3056–3062 (2007)
7. G.-B. Huang, L. Chen, A generalized growing and pruning RBF (GGAP-RBF) neural network for function approximation. *IEEE Trans. Neural Networks* **71**, 3460–3468 (2008)
8. Q. Liu, Q. He, and Z. Shi, Extreme support vector machine classifier, in *Knowledge Discovery and Data Mining* (Springer, Berlin, 2008), pp. 222–233
9. G. Huang, X. Ding, H. Zhou, Optimization method based extreme learning machine for clas-sification. *Neurocomputing* **74**(1), 155–163 (2010)
10. N. X, Locality preserving projections. *Neural Inf. proc. syst.* **16**, 153 (2004)
11. D. Cai, X. He, J. Han, Graph regularized nonnegative matrix factorization for data representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(8), 1548–1560 (2011)
12. G.B. Huang, O.Y. Zhu, C.K. Siew, Extreme learning machine: theory and applications. *Neurocomputing* **70**(1), 489–501 (2006)
13. G.B. Huang, D.H. Wang, Y. Lan, Extreme learning machines: a survey. *Int. J. Mach. Learn. Cybern.* **2**(1), 107–122 (2011)
14. G.-B. Huang, L. Chen, C.-K. Siew, Universal approximation using incremental constructive feedforward networks with random hidden nodes. *IEEE Trans. Neural Networks* **17**(4), 879–892 (2006)

15. G. Zhou, A. Cichocki, S. Xie, Fast nonnegative matrix/tensor factorization based on low-rank approximation. *IEEE Trans. Signal Process.* **60**(6), 2928–2940 (2012)
16. P.M. Rossini, A.T. Barker, A. Berardelli, Non-invasive electrical and magnetic stimulation of the brain and spinal cord and roots: basic principles and procedures for routine clinical application. *Electroencephalogr. Clin. Neurophysiol. Suppl.* **91**(2), 79–92 (1994)
17. S. Nikitidis, A. Tefas, N. Nikolaidis, Subclass discriminant nonnegative matrix factorization for facial image analysis. *Pattern Recognit.* **45**(12), 4080–4091 (2012)
18. W. Y and Z. Y., “Non-negative matrix factorization: a comprehensive review”, *Pattern Recognition*, vol. 1, no. 1, 2011.
19. Z. Luo, N. Guan, D. Tao, Non-negative patch alignment framework. *IEEE Trans. Neural Networks* **22**(8), 1218–1230 (2011)
20. D. Cai, X. He, J. Han, Document clustering using locality preserving indexing. *IEEE Trans. Knowl. Data Eng.* **17**(12), 1624–1637 (2005)

Extreme Learning Machines 2013: Algorithms and
Applications

Sun, F.; Toh, K.-A.; Romay, M.G.; Mao, K. (Eds.)

2014, VI, 225 p. 100 illus., 74 illus. in color., Hardcover

ISBN: 978-3-319-04740-9