An unsupervised discriminative extreme learning machine and its applications to data clustering

Yong Peng[a], Wei-Long Zheng[a], Bao-Liang Lu[a,b,*]

[a] *Center for Brain-like Computing and Machine Intelligence, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, P.R.China*

[b] *Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai 200240, P.R.China*

Abstract

Extreme Learning Machine (ELM), which was initially proposed for training single-layer feed-forward networks (SLFNs), provides us a unified efficient and effective framework for regression and multiclass classification. Though various ELM variants were proposed in recent years, most of them focused on the supervised learning scenario while little effort was made to extend it into unsupervised learning paradigm. Therefore, it is of great significance to put ELM into learning tasks with only unlabeled data. One popular approach for mining knowledge from unlabeled data is based on the manifold assumption, which exploits the geometrical structure of data by assuming that nearby points will also be close to each other in transformation space. However, considering the manifold information only is insufficient for discriminative tasks. In this paper, we propose an improved unsupervised discriminative ELM (UDELM) model, whose main advantage is to combine the local manifold learning with global discriminative learning together. UDELM can be efficiently optimized by solving a generalized eigen-value decomposition problem. Extensive comparisons over several state-of-the-art models on clustering image and emotional EEG data demonstrate the efficacy of UDELM.

Keywords: Extreme learning machine (ELM), Unsupervised learning, Manifold information, Discriminative information, Image clustering, EEG

1. Introduction

ELM as an emerging learning technique provides an efficient unified solution to generalized feed-forward networks such as SLFNs. The main merit of ELM is that the network input weights are randomly assigned and independent from specific applications [1, 2], which makes the analytical solution of network output weights be efficiently obtained by solving a least square formula. Despite the fact that the determination of the network hidden layer outputs is based on randomly generated network input weights, it has been proven that SLFNs trained based on ELM algorithm still have the global approximation ability [3, 4]. ELM is a unified framework for regression and multiclass classification [5]. Due to its effectiveness and fast learning process in comparison with gradient descend-based optimization, the ELM model has been adopted in many applications such as face recognition [6], action recognition [7, 8], gesture recognition [9], security assessment [10], EEG signal processing [11], data privacy [12], image quality assessment [13, 14] and remote sensing [15].

Though many ELM variants were proposed in the last few years [16, 17, 18, 19, 20, 21, 8], the extension on ELM research focused mainly on the supervised learning tasks. This greatly limits the applicability of ELM in utilizing unlabeled data. Moreover, in many real world applications, labeled data is usually expensive to obtain but the unlabeled data is relatively easy to collect, which drives us to extend ELM to unsupervised learning by properly harnessing the unlabeled data. On the basis of manifold regularization, Huang and his colleagues proposed two ELM variants, semi-supervised ELM and unsupervised ELM (USELM) [22]. He et al. proposed to do clustering in ELM hidden layer space in view of the good properties of its random feature mapping, which shows better results than clustering in the original data space [23]. The part from hidden layer to output layer of ELM was discarded and the hidden layer representation was used for clustering. The rationality of ELM feature mapping was also analyzed in [23]. A new ELM clustering technique was presented by Akusok et al. [24] by incorporating some prior knowledge into clustering. This method utilizes the prior of the exact number of points in each cluster; however, this requirement is hard to satisfy in real world applications. We are usually provided with imbalance data sets which have different number of points in different clusters. These models greatly enlarge the applicability of ELM.

In this paper, we aim to make improvements on the basis of USELM [22] for the reasons that (1) we want to

[*] Corresponding author

Email addresses: stany.peng@gmail.com (Yong Peng), bllu@sjtu.edu.cn (Bao-Liang Lu) retain the whole architecture of ELM network, from input layer, hidden layer to output layer; (2) we need not to know the exact number of points in each cluster before clustering. USELM, which was designed to exploit the underlying structure of data, shows excellent performance in clustering when comparing with several state-of-the-art unsupervised algorithms [22]. However, it pays only attention to the local structure of data and ignores the discriminative information of different classes. Various studies have shown that both structure information and discriminative information are important in dealing with discriminative tasks such as classification [25, 26, 27] and clustering [28, 29]. Specifically, Guan *et al.* introduced the manifold regularization and the margin maximization into non-negative matrix factorization and obtained the manifold regularized discriminative non-negative matrix factorization [25]. In [26], similar technique was incorporated into the ELM framework for EEG-based emotion recognition. Shu *et al.* included the graph regularization term into the discriminative analysis based on spectral regression [27]. The formulated LocLDA method covers both local and global structure information, which is more effective for face recognition. In [28], Yang *et al.* proposed to exploit the discriminative information in each local data clique based on constructing an elaborate local graph Laplacian and then globally integrate the local models of all the local cliques. The formulated model, local discriminant models and global integration (LDMGI), was put into spectral clustering and promising results were demonstrated in comparison with ordinary normalized cut [30]. In [29], both local manifold learning and global discriminative learning are incorporated into non-negative matrix factorization to learn effective data representation.

Inspired by existing studies, we propose a new unsupervised ELM learning model, unsupervised discriminative ELM, to utilize both the local structure and global discriminative information of data. Our goal is to learn a well-structured data representation for data clustering. On one hand, the learned data representation can preserve the intrinsic structure as much as possible through efficiently exploiting the local manifold information. On the other hand, the global discriminative information is utilized to qualify the learned representation has the discriminative power, e.g., differentiating samples from different clusters.

The main contributions of this paper can be summarized as follows

(1) We propose the *unsupervised discriminative ELM* to derive better data representations for clustering. UDELM utilizes both the local structure and global discriminative information of data.

(2) Different from USELM, which leaves the parameter of the number of output neurons to tune, UDELM defines such value as the number of the clusters. This more coincides with the original ELM definition.

(3) Extensive experiments were conducted to evaluate the clustering performance of UDELM by comparing with several state-of-the-art algorithms. Results on five widely used image data sets and one emotional EEG data set demonstrate the efficacy of UDELM.

The remainder of this paper is organized as follows. Section 2 provides a brief review of ordinary ELM and USELM [22]. Section 3 proposes the model formulation and optimization method of UDELM. Experimental studies to evaluate the performance of UDELM are given in Section 4. Section 5 concludes the paper.

2. Preliminaries

2.1. Extreme learning machine

Denote $\{\mathbf{x}_i, c_i\}_{i=1,\ldots,N}$ a set of N raw feature vectors $\mathbf{x}_i \in \mathbb{R}^D$ and the corresponding class labels $c_i \in \{1,\ldots,C\}$. The task is to train a SLFN with $\{\mathbf{x}_i, c_i\}_{i=1,\ldots,N}$. Such a network consists of D input (the dimensionality of $\mathbf{x}_i$), L hidden and C output (the number of classes) neurons. In ELM, the number of hidden neurons is usually set to be larger than the number of classes to ensure the global approximation ability [5], *i.e.*, $L \gg C$. For each training vector $\mathbf{x}_i$, the corresponding network target vector is $\mathbf{t}_i = [t_{i1},\ldots,t_{iC}]$. Generally, when $\mathbf{x}_i$ belongs to class k, that is $c_i = k$, we have $t_{ij} = 1$ if $j = k$ and $t_{ij} = -1$ otherwise. In ELM, the network input weights $\mathbf{W} \in \mathbb{R}^{L\times D}$ and the hidden layer bias $\mathbf{b} \in \mathbb{R}^L$ are randomly generated, which leads to the analytical calculation of the network output weights $\boldsymbol{\beta} \in \mathbb{R}^{L\times C}$.

Based on the above settings, the network response $\mathbf{o}_i = [o_{i1},\ldots,o_{iC}]$ corresponding to $\mathbf{x}_i$ can be calculated by

$$o_{ik} = \sum_{j=1}^{L} \beta_{jk} h_j(\mathbf{x}_i),\ k = 1,\ldots,C \tag{1}$$

where $\mathbf{h}(\mathbf{x}_i) = [h_1(\mathbf{x}_i),\ldots,h_L(\mathbf{x}_i)] \in \mathbb{R}^{1\times L}$ is the output row vector of the hidden layer corresponding to the input $\mathbf{x}_i$. $\mathbf{h}(\mathbf{x}_i)$ actually maps the sample $\mathbf{x}_i$ from the D-dimensional input space $\mathcal{X}$ to the L-dimensional ELM feature space $\mathcal{H}$. By storing the network hidden layer outputs for all the training vectors in one matrix, we have

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}(\mathbf{x}_1) \\ \mathbf{h}(\mathbf{x}_2) \\ \vdots \\ \mathbf{h}(\mathbf{x}_N) \end{bmatrix} = \begin{bmatrix} h_1(\mathbf{x}_1) & h_2(\mathbf{x}_1) & \cdots & h_L(\mathbf{x}_1) \\ h_1(\mathbf{x}_2) & h_2(\mathbf{x}_2) & \cdots & h_L(\mathbf{x}_2) \\ \vdots & \vdots & \vdots & \vdots \\ h_1(\mathbf{x}_N) & h_2(\mathbf{x}_N) & \cdots & h_L(\mathbf{x}_N) \end{bmatrix}.$$

We can rewrite (1) in a compact form as

$$\mathbf{O} = \mathbf{H}\boldsymbol{\beta}, \tag{2}$$

where $\mathbf{O} \in \mathbb{R}^{N\times C}$ is a matrix containing the network responses for all training samples $\mathbf{x}_i, i = 1,2,\ldots,N.$.

The original ELM assumes that $\mathbf{o}_i = \mathbf{t}_i$, $i = 1,\ldots,N$ or $\mathbf{O} = \mathbf{T}$ in matrix form, where $\mathbf{T} = [\mathbf{t}_1;\ldots;\mathbf{t}_N]$ is a matrix containing the network target vectors. By using (2), the closed form of the network output weights is

$$\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger}\mathbf{T}, \tag{3}$$

where $\mathbf{H}^{\dagger}$ is the Moore-Penrose generalized inverse of $\mathbf{H}$. If $\mathbf{H}^T\mathbf{H}$ is nonsingular, $\mathbf{H}^{\dagger} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T$; or when $\mathbf{H}\mathbf{H}^T$ is nonsingular, $\mathbf{H}^{\dagger} = \mathbf{H}^T(\mathbf{H}\mathbf{H}^T)^{-1}$ [5]. Once the network output weights are obtained, the network response for an unseen vector $\mathbf{x}_{new}$ is given by

$$\mathbf{o}_{new} = \mathbf{h}(\mathbf{x}_{new})\boldsymbol{\beta}. \tag{4}$$

To avoid the singularity problem when calculating the inverse of $\mathbf{H}^T\mathbf{H}$, a regularization term is introduced to minimize the norm of the network output weights, which results in the following objective of regularized ELM as

$$\begin{aligned} \arg\min_{\boldsymbol{\beta}} \mathcal{J}_{RELM} &= \frac{1}{2}\|\boldsymbol{\beta}\|^2 + \frac{\lambda}{2}\sum_{i=1}^{N}\|\xi_i\|_2^2, \\ s.t., \ \xi_i &= \mathbf{t}_i - \mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta}, \ i = 1, \ldots, N \end{aligned} \tag{5}$$

where $\xi_i \in \mathbb{R}^{1\times C}$ is the error vector corresponding to $\mathbf{x}_i$ and $\lambda > 0$ is a regularization parameter. Therefore, the network output weights in regularized ELM can be estimated as

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{H}^T\mathbf{H} + \frac{\mathbf{I}}{\lambda}\right)^{-1}\mathbf{H}^T\mathbf{T}. \tag{6}$$

This regularization method is a general way to make the solution of least square regression stable, which is called "ridge regression" [31] in statistics.

As a whole, training a SLFN using ELM algorithm can be summarized in Algorithm 1.

Algorithm 1 ELM-based SLFN training

```
Input: training set $\mathcal{X} = \{\mathbf{x}_i, \mathbf{t}_i\}_{i=1,\ldots,N}$, activation function $g(\cdot)$, number of hidden neurons $L$ and regularization parameter $\lambda$;
Output: Output weight matrix $\boldsymbol{\beta}$;
1: Generate the network input weight matrix $\mathbf{W}$ and hidden layer bias $\mathbf{b}$ randomly;
2: Calculate the hidden layer output matrix $\mathbf{H}$;
3: Calculate the network output weight matrix $\hat{\boldsymbol{\beta}}$ according to (3) or (6).
```

2.2. Unsupervised extreme learning machine

In [22], ELM was extended to cluster data based on manifold regularization. The formulated USELM aims at exploiting the underlying structure of data, having the following objective

$$\begin{aligned} &\arg\min_{\boldsymbol{\beta}} \ \mathrm{Tr}\left(\boldsymbol{\beta}^T\mathbf{H}^T\mathbf{L}\mathbf{H}\boldsymbol{\beta}\right) + \lambda\|\boldsymbol{\beta}\|^2, \\ &s.t., \ (\mathbf{H}\boldsymbol{\beta})^T\mathbf{H}\boldsymbol{\beta} = \mathbf{I} \end{aligned} \tag{7}$$

where $\mathbf{L}$ is the graph Laplacian matrix constructed in the input data space $\mathcal{X}$, and the constraint is introduced to avoid a degenerated solution. Therefore, the optimal solution to problem (7) can be obtained by solving the following generalized eigen-value decomposition problem

$$(\mathbf{H}^T\mathbf{L}\mathbf{H} + \lambda\mathbf{I})\mathbf{v} = \gamma\mathbf{H}^T\mathbf{H}\mathbf{v}. \tag{8}$$

The network output weight matrix $\boldsymbol{\beta}$ can be formed by stacking the eigenvectors as its columns.

3. Unsupervised discriminative extreme learning machine

3.1. Analysis of USELM

USELM aims at finding the linear embedding of hidden layer representation by maximally preserving the local structure of data, which can be seen as a subspace learning process based on manifold assumption. The network response is the learned data representation for clustering. Therefore, USELM is essentially a two-stage algorithm in which the ELM random feature mapping is followed by the locality preserving projection [32].

Though USELM has shown excellent performance in clustering task, it still leaves room for improvement: (1) From the hidden layer to output layer, it acts actually as an unsupervised subspace learning process and the dimensionality of subspace is arbitrary, which needs to be tuned. This deviates from the ordinary ELM definition in which the number of output neurons is usually set as the number of classes/clusters. (2) Recent studies [25, 33] demonstrate that both manifold structure and discriminative information of data are beneficial for classification and clustering; however, USELM is only based on the manifold assumption, which neglects to consider the discriminative information of data.

The proposed UDELM can deal with unlabeled data more effectively by utilizing both *local manifold learning* and *global discriminative learning*.

3.2. Local manifold learning

In many real-world applications, data points are more likely to reside on a low-dimensional manifold, which leads to the manifold learning research in recent years [34, 35, 36, 37]. A natural assumption in manifold learning is that nearby data points will be likely to have similar properties and should be categorized into the same cluster [38]. USELM is directly based on such 'manifold assumption', which can preserve the local structure of data when projecting the hidden layer data representation to the output layer. Therefore, it is expected that if $\mathbf{x}_i$ and $\mathbf{x}_j$ are within a small neighborhood, the corresponding learned USELM representations $\mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta}$ and $\mathbf{h}(\mathbf{x}_j)\boldsymbol{\beta}$, will share similar properties.

The widely used method to preserve the intrinsic data structure is the graph-based manifold method [39, 40]. Generally, data points are modeled as a graph with N vertices, each edge is established if two data points are

k nearest neighbors. There are many choices to define the affinity matrix $\mathbf{S}$ on the graph such as binary weights, Gaussian weights and dot product weights. Similar to USELM, we adopt the Gaussian weights, *i.e.*,

$$s_{ij} = \begin{cases} \exp\left(\frac{-\|\mathbf{x}_i-\mathbf{x}_j\|^2}{\sigma}\right), & \text{if } \mathbf{x}_i \in \mathcal{N}_k(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in \mathcal{N}_k(\mathbf{x}_i) \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

where σ is the bandwidth parameter and $\mathcal{N}_k(\mathbf{x}_i)$ denotes the set of k nearest neighbors of $\mathbf{x}_i$. The graph Laplacian matrix is $\mathbf{L} = \mathbf{D} - \mathbf{S}$, where $d_{ii} = \sum_j s_{ij}$.

Preserving the local structure of data is equivalent to minimize the following objective

$$\arg\min_{\boldsymbol{\beta}} \sum_{i,j=1}^{N} s_{ij}\|\mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} - \mathbf{h}(\mathbf{x}_j)\boldsymbol{\beta}\|^2, \tag{10}$$

which can be simplified as $\text{Tr}(\boldsymbol{\beta}^T\mathbf{H}^T\mathbf{L}\mathbf{H}\boldsymbol{\beta})$. Here $\text{Tr}(\cdot)$ is the trace operator. In this way, the nearby points are encouraged to have similar representations.

3.3. Global discriminative learning

USELM utilizes the local structure for clustering and the Laplacian matrix is only based on k nearest neighbors relationship of input data. As was pointed out in [41], emphasizing local structure only may induce overfitting and thus degrade the clustering performance. In order to make the learned data representation characterize the discriminative power, we attempt to discover the global discriminative information of data. Before introducing the discriminative regularization term, we briefly review the definitions of scaled indicator matrix, between-cluster scatter and total scatter [42].

Denote the indicator matrix by $\mathbf{Y} \in \{0,1\}^{N\times C}$, then the corresponding scaled indicator matrix is

$$\mathbf{F} = \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1/2}, \tag{11}$$

where each column $\mathbf{F}_j$ in $\mathbf{F}$ is $[0,\ldots,0,\underbrace{1,\ldots,1}_{n_j},0,\ldots,0]^T/\sqrt{n_j}$, and n_j is the number of points in the j-th cluster. In UDELM, we expect that the learned data representation $\mathbf{H}\boldsymbol{\beta}$ can characterize the structure of $\mathbf{F}$ to capture the discriminative ability. Then the representation will yield promising clustering performance in the embedded space $\mathbb{R}^{N\times C}$. Obviously, $\mathbf{H}\boldsymbol{\beta}$ should approach $\mathbf{F}$, which can be measured by a small constant ε, *i.e.*, $\|\mathbf{H}\boldsymbol{\beta} - \mathbf{F}\|^2 \leq \varepsilon$.

Given a centering matrix $\mathbf{M} = \mathbf{I}_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^T$, where $\mathbf{1}_N$ is a column vector with all ones and $\mathbf{I}_N$ is an identity matrix, the between-cluster scatter and total scatter matrices are respectively defined as [43]

$$\begin{aligned} \mathbf{S}_b &= \tilde{\mathbf{X}}\mathbf{F}\mathbf{F}^T\tilde{\mathbf{X}}^T, \\ \mathbf{S}_t &= \tilde{\mathbf{X}}\tilde{\mathbf{X}}^T, \end{aligned} \tag{12}$$

where $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{M}$ is the centered data matrix. According to the principle of discriminative analysis, the distance between data points from different clusters should be as large as possible while the distance between data points from the same cluster should be as small as possible. To this end, it is reasonable to maximize the following objective

$$\arg\max_{\mathbf{F}} \ \text{Tr}\left[(\mathbf{S}_t + \mu\mathbf{I})^{-1}\mathbf{S}_b\right], \tag{13}$$

where $\mu > 0$ is a small constant to avoid the singularity in calculating the inverse of $\mathbf{S}_t$. Since $\text{Tr}(\mathbf{F}^T\mathbf{M}\mathbf{F}) = C - 1$ is a constant [41], problem (13) is equivalent to

$$\arg\min_{\mathbf{F}} \ \text{Tr}\left[\mathbf{F}^T(\mathbf{M} - \tilde{\mathbf{X}}^T(\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T + \mu\mathbf{I})^{-1}\tilde{\mathbf{X}})\mathbf{F}\right]. \tag{14}$$

Therefore, we can obtain the data representation characterizing the discriminative power by minimizing the above objective.

3.4. UDELM model formulation

Treating the discriminative information as a regularizer, we combine problems (10) and (14) together to formulate the UDELM objective as

$$\begin{aligned} &\arg\min_{\boldsymbol{\beta},\mathbf{F}} \ \text{Tr}(\boldsymbol{\beta}^T\mathbf{H}^T\mathbf{L}\mathbf{H}\boldsymbol{\beta}) + \lambda\text{Tr}(\mathbf{F}^T\mathbf{Q}\mathbf{F}), \\ &\quad s.t.\ \|\mathbf{H}\boldsymbol{\beta} - \mathbf{F}\|^2 \leq \varepsilon, \end{aligned} \tag{15}$$

where $\mathbf{Q} = \mathbf{M} - \mathbf{U}$ and $\mathbf{U} = \tilde{\mathbf{X}}^T(\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T + \mu\mathbf{I})^{-1}\tilde{\mathbf{X}}$. Recall that $\mathbf{F}$ is the scaled indicator matrix. According to its definition, in each row of $\mathbf{F}$, there is only one positive element and the others are 0, which makes (15) an NP-hard problem. Therefore, we relax $\mathbf{F}$ into continuous domain and make this problem tractable. It is easily to verify that

$$\mathbf{F}^T\mathbf{F} = (\mathbf{Y}^T\mathbf{Y})^{-1/2}\mathbf{Y}^T\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1/2} = \mathbf{I}_C. \tag{16}$$

Therefore, objective (15) becomes

$$\begin{aligned} &\arg\min_{\boldsymbol{\beta},\mathbf{F}} \ \text{Tr}(\boldsymbol{\beta}^T\mathbf{H}^T\mathbf{L}\mathbf{H}\boldsymbol{\beta}) + \lambda\text{Tr}(\mathbf{F}^T\mathbf{Q}\mathbf{F}), \\ &\quad s.t.\ \|\mathbf{H}\boldsymbol{\beta} - \mathbf{F}\|^2 \leq \varepsilon,\ \mathbf{F}^T\mathbf{F} = \mathbf{I}. \end{aligned} \tag{17}$$

Similar to [29], we set $\mathbf{H}\boldsymbol{\beta} = \mathbf{F}$, which is equivalent to ε=0; then we have the following objective

$$\begin{aligned} &\arg\min_{\boldsymbol{\beta}} \ \text{Tr}\left[(\mathbf{H}\boldsymbol{\beta})^T(\mathbf{L} + \lambda\mathbf{Q})(\mathbf{H}\boldsymbol{\beta})\right], \\ &\quad s.t.\ (\mathbf{H}\boldsymbol{\beta})^T(\mathbf{H}\boldsymbol{\beta}) = \mathbf{I} \end{aligned} \tag{18}$$

where the constraint is introduced to avoid a degenerated solution. The solution to problem (18) is given by the following theorem.

Theorem 1. *An optimal solution to (18) can be obtained via generalized eigen-value decomposition. Specifically, columns of $\boldsymbol{\beta}$, $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_C] \in \mathbb{R}^{N\times C}$, are eigenvectors corresponding to the first C smallest eigenvalues of the generalized eigenvalue problem as*

$$\mathbf{H}^T(\mathbf{L} + \lambda\mathbf{Q})\mathbf{H}\beta_i = \gamma_{ii}\mathbf{H}^T\mathbf{H}\beta_i,\ i = 1, 2, \ldots, C. \tag{19}$$

Proof. The Lagrangian function to problem (18) is

$$\begin{aligned}\mathcal{L}(\boldsymbol{\beta}) =& \mathrm{Tr}\left[(\mathbf{H}\boldsymbol{\beta})^T(\mathbf{L}+\lambda\mathbf{Q})(\mathbf{H}\boldsymbol{\beta})\right] \\ &- \mathrm{Tr}\left[\boldsymbol{\Gamma}\left((\mathbf{H}\boldsymbol{\beta})^T(\mathbf{H}\boldsymbol{\beta})-\mathbf{I}\right)^T\right].\end{aligned} \tag{20}$$

Taking the derivative of $\mathcal{L}(\boldsymbol{\beta})$ w.r.t. $\boldsymbol{\beta}$, and setting the derivative to zero, we have

$$\frac{\partial\mathcal{L}(\boldsymbol{\beta})}{\partial\boldsymbol{\beta}} = 2\mathbf{H}^T(\mathbf{L}+\lambda\mathbf{Q})\mathbf{H}\boldsymbol{\beta} - 2\mathbf{H}^T\mathbf{H}\boldsymbol{\beta}\boldsymbol{\Gamma} = 0, \tag{21}$$

$$\mathbf{H}^T(\mathbf{L}+\lambda\mathbf{Q})\mathbf{H}\boldsymbol{\beta} = \mathbf{H}^T\mathbf{H}\boldsymbol{\beta}\boldsymbol{\Gamma}. \tag{22}$$

For each column in $\boldsymbol{\beta}$, we have the following eigenvalue decomposition problem

$$\mathbf{H}^T(\mathbf{L}+\lambda\mathbf{Q})\mathbf{H}\beta_i = \gamma_{ii}\mathbf{H}^T\mathbf{H}\beta_i,\ i=1,2,\ldots,C,$$

where γ_{ii} is the i-th element in diagonal matrix $\boldsymbol{\Gamma}$. □

3.5. Discussions

To outline the whole process of UDELM-based image clustering, we show the overview in Figure 1. Specifically, we first extract the raw features from input images. Then we learn well-structured image representation by feeding the raw features into UDELM. Finally, K-means will be applied on the learned representations to do clustering.

Local structure and global discrimination information are beneficial for both supervised and unsupervised tasks . Naturally, in ELM framework, it is necessary to introduce both properties to enhance its performance [21, 22, 26]. Figure 2 shows the model evolution of several ELM variants. We can clearly see the connection among these four ELM variants.

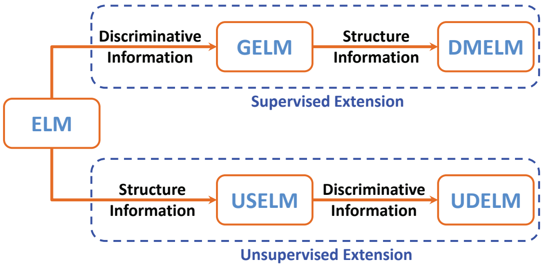


Figure 2: Model evolution of ELM variants.

4. Experimental studies

In this section, we conduct experiments on image and EEG data clustering tasks to demonstrate the effectiveness of UDELM. The source code will be available from `http://bcmi.sjtu.edu.cn/~pengyong` to help reproducing the experimental results.

4.1. Evaluation metrics

Following the convention of clustering study, we use the clustering accuracy (ACC) and normalized mutual information (NMI) as evaluation metircs.

Clustering Accuracy (ACC). Given a data set with n points; for $\mathbf{x}_i$, let r_i be the clustering result from the clustering algorithm and s_i the ground truth label. ACC is defined as

$$\mathrm{ACC} = \frac{\sum_{i=1}^{N}\delta(s_i, \mathrm{map}(r_i))}{N}, \tag{23}$$

where $\delta(x,y)=1$ if $x=y$; $\delta(x,y)=0$ otherwise, and $\mathrm{map}(r_i)$ is the optimal mapping function that permutes clustering labels to match the ground truth labels. The best mapping can be found by using the Kuhn-Munkres algorithm [44]. A larger ACC indicates better performance.

Normalized Mutual Information (NMI). For two arbitrary variables P and Q, NMI is defined as [45]

$$\mathrm{NMI}(P,Q) = \frac{I(P,Q)}{\sqrt{H(P)H(Q)}}, \tag{24}$$

where $I(P,Q)$ is the mutual information between P and Q, $H(P)$ and $H(Q)$ respectively denote the entropies of P and Q. Obviously, if P is identical with Q, $\mathrm{NMI}(P,Q)$ will be equal to 1; if P is independent from Q, $\mathrm{NMI}(P,Q)$ will become 0. Let t_l be the number of samples in the cluster $\mathcal{C}_l$ $(1 \le l \le C)$ obtained by clustering algorithm and $\tilde{t}_h$ be the number of samples in the h-th ground truth cluster $(1 \le h \le C)$. NMI is defined as [45]

$$\mathrm{NMI} = \frac{\sum_{l=1}^{C}\sum_{h=1}^{C} t_{l,h}\log(\frac{n\cdot t_{l,h}}{t_h\tilde{t}_h})}{\sqrt{(\sum_{l=1}^{C} t_l\log\frac{t_l}{n})(\sum_{h=1}^{C}\tilde{t}_h\log\frac{\tilde{t}_h}{n})}}, \tag{25}$$

where $t_{l,h}$ is the number of samples, which are in the intersection between the cluster $\mathcal{C}_l$ and the h-th ground truth cluster. Similarly, a larger NMI indicates better clustering results.

4.2. Image clustering

4.2.1. Data corpora

- **ORL**[1]. This data set contains ten different images of each of 40 distinct subjects. The images were taken at different times, varying the lighting, facial expressions and facial details. Each image is manually cropped and normalized to size of 64×64 pixels.

- **Yale**[2]. This data set contains 64×64 gray scale images of 15 subjects. There are 11 images per subject facial expression or configuration.

[1] http://www.uk.research.att.com/facedatabase.html

[2] http://cvc.yale.edu/projects/yalefaces/yalefaces.html

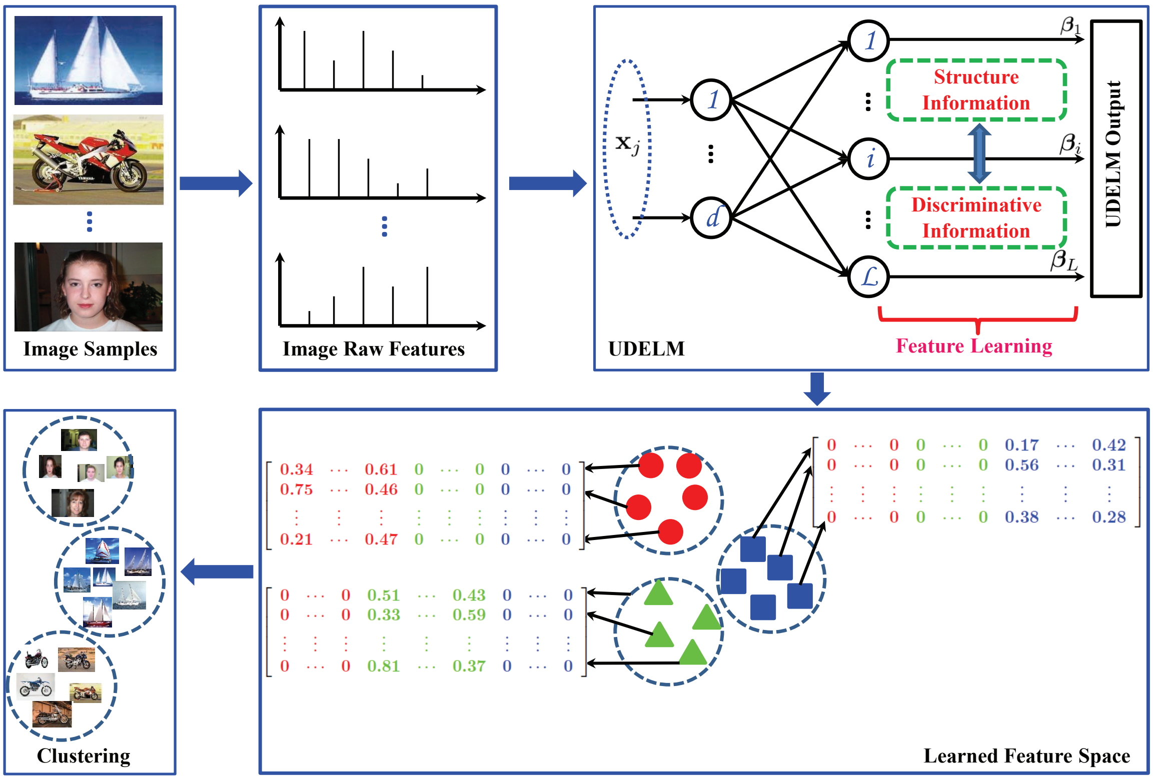


Figure 1: UDELM-based image clustering framework overview.

- **COIL20**[3]. This data set has 1440 images of 20 objects, with each object containing 72 images from different views. We have resized each image to 32×32 pixels.

- **MNIST**[4]. This data set has a training set of 60,000 samples, and a test set of 10,000 samples with each image size of 28×28. We use a subset of 4,000 samples for clustering.

- **Caltech101**[5]. This data set contains 101 object categories with about 40 to 800 images for each category. Each image is roughly with size 300×200 pixels. Except the BACKGROUND GOOGLE category, we choose the subset with 10 largest categories, which contains 3044 images in total. By extracting the SIFT descriptors and then generating the codewords as the features, we get a 500-dimensional frequency histogram for each image.

Table 1 summarizes the detailed information of these five data sets in terms of the total number of samples, feature dimension and clusters. Several sample images from these five data sets are shown in Figure 3.

Table 1: Statistics of the five data sets.

Dataset	Size(N)	Dimensionality(D)	# Classes (C)
ORL	400	4096	40
Yale	165	4096	15
COIL20	1440	1024	20
MNIST	4000	784	10
Caltech101	3044	500	10

4.2.2. Experimental settings

To show the effectiveness of the clustering performance by UDELM, we compared UDELM with the following five popular algorithms:

- Canonical K-means clustering method (Kmeans);

- Graph Regularized Non-negative Matrix Factorization (GNMF [39]);

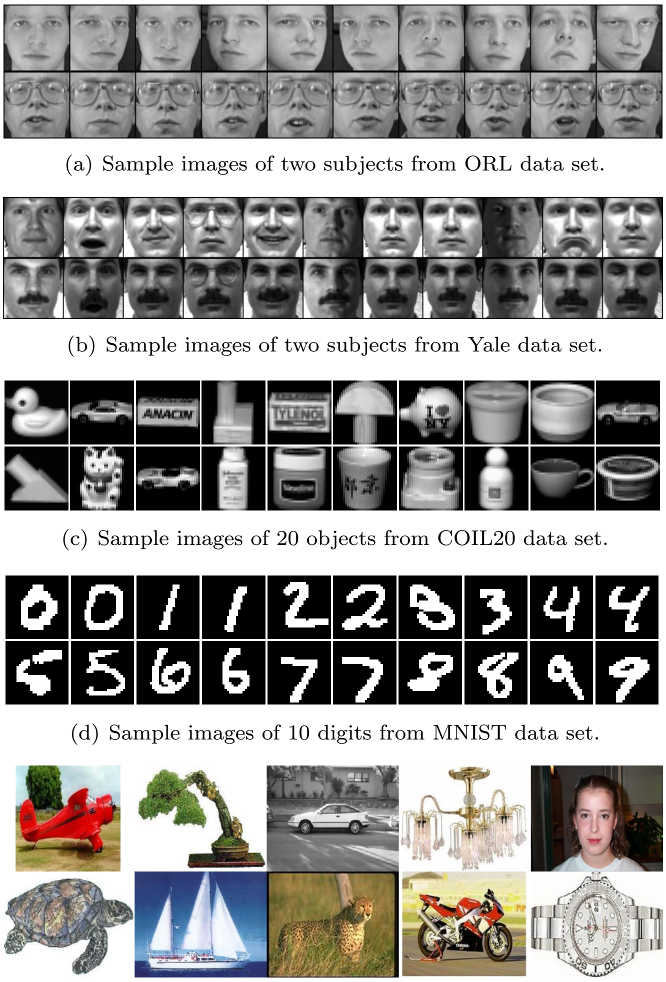
- Locally Consistent Concept Factorization (LCCF [40]);

- Local Discriminative Models and Global Integration (LDMGI [28]);

- Unsupervised Extreme Learning Machine (USELM [22]);

The evaluations were conducted with different numbers of clusters. For each given cluster number, 20 test runs were conducted on different randomly chosen clusters.

[3] http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php
[4] http://yann.lecun.com/exdb/mnist/
[5] http://www.vision.caltech.edu/Image_Datasets/Caltech101/



(a) Sample images of two subjects from ORL data set.

(b) Sample images of two subjects from Yale data set.

(c) Sample images of 20 objects from COIL20 data set.

(d) Sample images of 10 digits from MNIST data set.

(e) Sample images of 10 largest categories from Caltech101 data set.

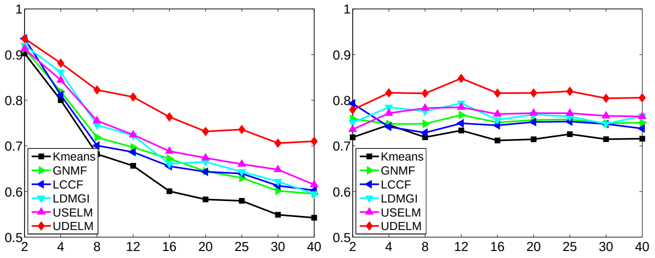
Figure 3: Sample images from five data sets. a) ORL, b) Yale, c) COIL20, d) MNIST and e) Caltech101.

The final performance is recorded by averaging the performance of the 20 tests. For fair comparison, we record the randomly selected cluster indices, and fix them for all compared algorithms. For GNMF and LCCF, we set the number of basis columns to be the number of clusters and use the obtained coefficient to determine the cluster label of each data point. The K-means is repeated 100 times with different initializations and the best result in terms of the objective function of K-means is recorded.

For all the graph-based methods, the number of nearest neighbors k is set to a small number 5 as suggested in [39]. The 'Gaussian' function is used to measure the affinity for all related algorithms in which the bandwidth parameter is set as the average value of Euclidean distance among the points. In LCCF, 'Gaussian' kernel is used and the bandwidth parameter is set in the same way. The regularization parameter λ in GNMF, LCCF and USELM is searched from $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$. The parameter λ used to avoid singularity problem in LDMGI is searched from $\{10^{-8}, 10^{-6}, \ldots, 10^{8}\}$. The parameters μ and λ in UDELM are respectively selected from $\{10^{-8}, 10^{-6}, \ldots, 10^{8}\}$ and $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$. For USELM and UDELM, the activation function is 'sigmoid' function and the number of hidden layer neurons is three times of the input dimension. We report the best results from the optimal ones.

4.2.3. Image clustering results

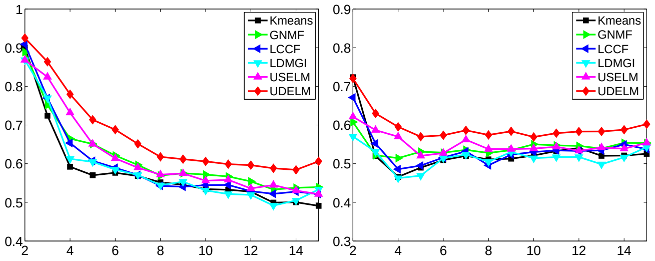
Figure 4 shows the effectiveness of the proposed UDELM on ORL. The detailed results are described in Table 2. It is clear that UDELM outperforms all the other algorithms expect when $K = 2$. Based on the quantitative comparison results, UDELM averagely achieves 6.26% improvement in accuracy and 4.45% improvement in normalized mutual information than the best results of the other algorithms. Similar results can be found in the other Yale face data set, which are shown in Figure 5 and Table 3. The improvements on accuracy and normalized mutual information are respectively 5.41% and 4.45%.



(a) Accuracy

(b) NMI

Figure 4: Clustering performance on ORL.



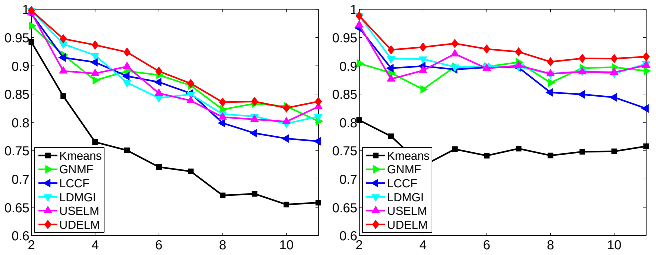
(a) Accuracy

(b) NMI

Figure 5: Clustering performance on Yale.

Figure 6 and Table 4 show the clustering results on COIL20 by graphical form and numerical form, respectively. GNMF performs quite well on this data set, which yields similar results as UDELM when K increases. However, for the average performance, UDELM can still obtain 2.10% improvement on accuracy and 2.70% improvement on normalized mutual information when comparing with the best results of other algorithms.

The clustering results on MNIST is shown in Figure 7 and Table 5. Obviously, UDELM outperforms all the other algorithms in all cases. The performance of LDMGI algorithm is close to that of UDELM and the differences on ac-

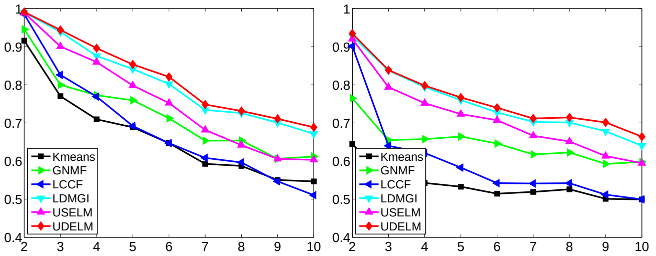


(a) Accuracy

(b) NMI

Figure 6: Clustering performance on COIL20.

curacy and normalized mutual information are only 1.15% and 1.10%. LDMGI also takes the local structure and discriminative information of data into consideration, which further shows the importance of these two techniques.

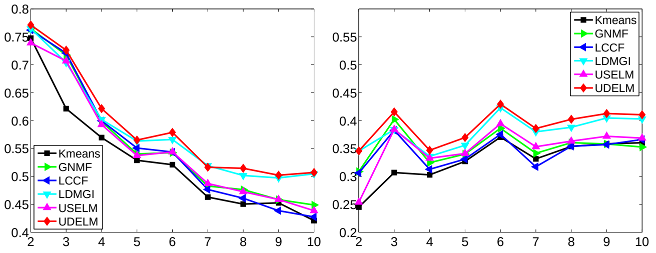


(a) Accuracy

(b) NMI

Figure 7: Clustering performance on MNIST.

Clustering on Caltech101 is a relatively difficult task because this is an imbalance data set in which each cluster contains different number of samples. Figure 8 and Table 6 present the results on this data set. We can see that both LDMGI and UDELM obtain promising results, while the UDELM still has 0.89% improvement on accuracy and 1.10% improvement on normalized mutual information with respect to LDMGI.



(a) Accuracy

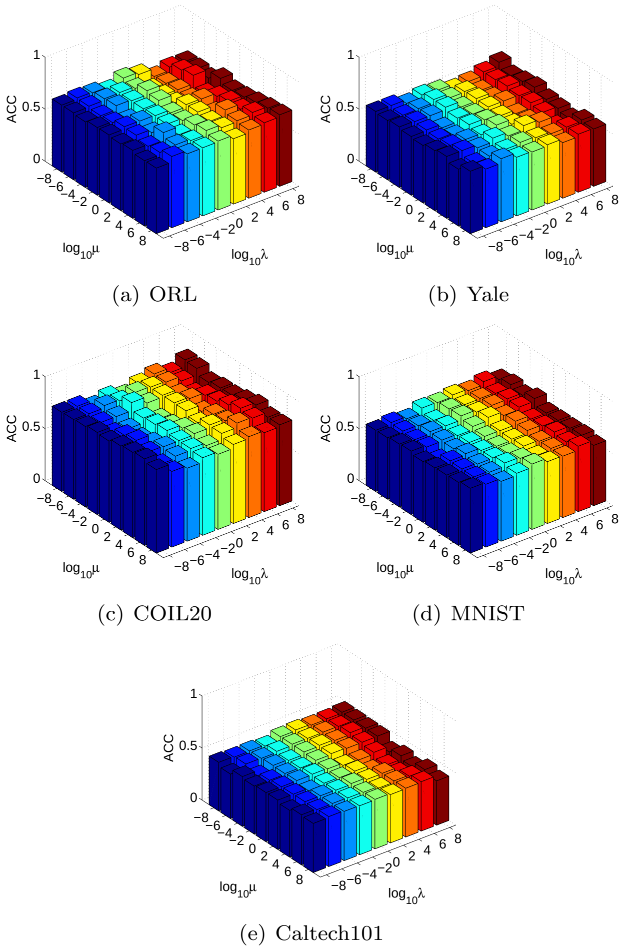
(b) NMI

Figure 8: Clustering performance on Caltech101.

4.2.4. *Parameter sensitivity analysis*

There are two hyper parameters, the regularization parameters μ and λ, in UDELM. The other two parameters: the number of nearest neighbors k and the bandwidth σ of Gaussian function are set the same values of all related algorithms in the above experiments, which should be fair comparison. Thus we only investigate the performance of UDELM in terms of the parameters combination (μ, λ). Figures 9 and 10 show how the average accuracy and normalized mutual information of UDELM varies with the parameters combination (μ, λ), respectively.

As we can see, the performance of UDELM is very stable with respect to the parameters combination (μ, λ). UDELM consistently achieves good performance when μ is relatively small and λ is large. The main role of μ is to avoid the singularity problem when calculating matrix inverse. Thus a small value can satisfy this. A large λ reflects that the global discriminative information is important for learning a well-structured data representation.



(a) ORL

(b) Yale

(c) COIL20

(d) MNIST

(e) Caltech101

Figure 9: Clustering accuracy of UDELM vs. μ and λ.

4.3. *EEG signal clustering*

EEG signals reflect the electrical activities along the scalp, which can provide us a reliable channel to investigate the human emotional states. In this section, we apply UDELM on EEG-based emotional states clustering task for further evaluating its effectiveness.

Table 2: Clustering performance on ORL.

K	Accuracy (%)					
	Kmeans	GNMF	LCCF	LDMGI	USELM	UDELM
2	90.25±14.46	91.25±14.95	**93.50±11.01**	92.00±10.93	91.25±13.36	**93.50±10.40**
4	80.00±10.42	81.88±10.54	81.12±10.84	86.13±8.53	84.38±9.10	**88.13±8.58**
8	68.19±8.93	71.87±8.86	70.06±10.47	74.56±9.19	75.50±6.74	**82.25±8.84**
12	65.63±6.79	69.67±7.09	68.63±8.23	72.25±8.35	72.42±7.01	**80.71±9.73**
16	60.06±6.71	67.19±6.03	65.53±6.82	66.06±6.24	68.84±5.43	**76.34±6.98**
20	58.27±3.92	64.45±4.54	64.32±5.70	66.50±4.59	67.35±4.00	**73.15±5.23**
25	57.96±4.07	62.98±3.97	63.94±3.74	64.34±4.68	66.00±3.58	**73.58±5.02**
30	54.90±3.12	60.15±2.97	61.27±4.31	62.20±3.81	64.80±2.65	**70.60±3.38**
40	54.25	59.50	60.25	59.50	61.50	**71.00**
Avg.	65.50	69.88	69.85	71.50	72.45	**78.81**
K	Normalized Mutual Information (%)					
	Kmeans	GNMF	LCCF	LDMGI	USELM	UDELM
2	71.86±37.30	75.94±38.38	**79.27±32.92**	75.09±31.36	73.60±36.15	77.90±30.79
4	74.59±11.19	74.79±13.78	74.17±12.19	78.46±11.23	77.14±11.21	**81.61±12.46**
8	71.89±8.08	74.84±7.36	72.88±9.36	77.58±8.11	78.27±5.74	**81.50±7.75**
12	73.37±5.61	76.76±5.26	74.94±5.48	79.34±6.48	78.46±5.26	**84.77±6.73**
16	71.18±4.85	75.13±4.80	74.49±4.85	75.70±4.05	76.96±4.33	**81.55±5.00**
20	71.44±2.53	75.65±3.08	75.22±3.83	76.86±2.91	77.16±2.91	**81.58±3.79**
25	72.56±3.00	75.78±2.21	75.32±2.30	76.42±2.93	77.15±2.53	**81.95±3.06**
30	71.46±2.42	74.92±2.27	74.79±2.87	74.82±3.26	76.56±2.20	**80.42±2.39**
40	71.57	75.08	73.79	76.54	76.43	**80.54**
Avg.	72.21	75.43	74.99	76.76	76.86	**81.31**

Table 3: Clustering performance on Yale.

K	Accuracy (%)					
	Kmeans	GNMF	LCCF	LDMGI	USELM	UDELM
2	90.00±17.05	88.86±11.47	90.91±10.84	86.82±10.62	86.82±16.48	**92.50±8.51**
3	72.42±21.38	75.15±11.36	76.97±13.28	76.97±12.41	82.42±9.71	**86.36±4.66**
4	59.20±17.32	66.48±11.03	65.34±12.40	61.25±11.24	73.18±11.48	**77.95±10.94**
5	57.00±9.05	65.09±10.71	60.73±15.69	60.45±12.33	65.09±12.99	**71.36±9.80**
6	57.65±6.19	62.12±8.02	58.94±9.04	58.48±7.62	61.36±8.08	**68.79±7.18**
7	56.82±5.51	59.68±5.05	57.08±3.51	57.08±5.68	58.96±4.31	**65.13±5.20**
8	55.17±5.88	56.93±6.37	54.26±4.75	54.43±5.35	57.16±3.60	**61.76±5.03**
9	54.55±5.52	57.53±4.20	54.04±5.61	55.35±4.69	57.47±3.81	**61.16±4.53**
10	53.41±6.30	57.18±4.65	54.45±4.93	53.09±4.53	55.59±4.36	**60.59±5.05**
11	53.26±3.53	56.69±3.83	54.50±3.88	52.15±4.27	55.83±3.72	**59.88±3.14**
12	52.77±5.33	55.42±3.75	52.88±4.01	51.93±3.81	53.60±3.59	**59.62±3.40**
13	49.93±4.82	53.32±3.88	52.20±3.30	49.16±2.88	54.48±2.85	**58.81±3.21**
14	50.03±3.06	53.77±3.84	52.76±2.73	50.49±2.06	53.05±3.53	**58.41±2.75**
15	49.09	53.94	52.12	53.33	52.12	**60.61**
Avg.	57.95	61.58	59.80	58.64	61.94	**67.35**
K	Normalized Mutual Information (%)					
	Kmeans	GNMF	LCCF	LDMGI	USELM	UDELM
2	72.36±35.46	60.77±29.47	67.11±29.18	57.04±25.07	62.14±37.54	**72.06±26.71**
3	52.01±28.79	52.16±13.27	55.23±18.23	53.08±14.07	58.68±15.88	**63.00±10.77**
4	46.60±22.50	51.43±15.68	48.57±13.86	46.25±11.70	56.98±13.34	**59.53±12.94**
5	48.98±12.41	53.09±14.52	49.50±16.77	46.90±10.89	52.02±13.38	**56.97±9.49**
6	50.93±7.96	52.86±8.93	51.60±8.70	51.44±7.94	52.75±8.03	**57.34±5.61**
7	52.02±6.09	53.56±6.26	53.20±5.06	52.34±6.39	56.20±5.31	**58.62±5.13**
8	51.11±7.75	52.77±6.70	49.57±6.59	50.52±6.45	53.77±5.78	**57.43±5.07**
9	51.30±5.79	53.50±5.15	52.35±5.45	53.01±5.44	53.78±4.87	**58.37±4.78**
10	52.12±6.06	55.01±5.63	53.02±4.99	51.42±5.75	53.91±5.35	**56.91±5.20**
11	53.27±3.41	54.73±3.67	53.44±3.41	51.71±4.40	54.36±3.61	**57.89±3.04**
12	54.07±4.31	54.59±3.85	53.29±2.98	51.72±3.72	53.34±3.56	**58.31±2.86**
13	52.03±3.27	53.92±3.55	53.44±2.57	49.85±3.07	54.16±2.26	**58.33±3.15**
14	52.10±3.45	55.38±2.82	54.97±2.39	51.65±2.31	53.87±2.19	**58.78±2.21**
15	52.55	55.34	53.75	54.28	55.38	**60.22**
Avg.	52.96	54.22	53.50	51.51	55.10	**59.55**

Table 4: Clustering performance on COIL20.

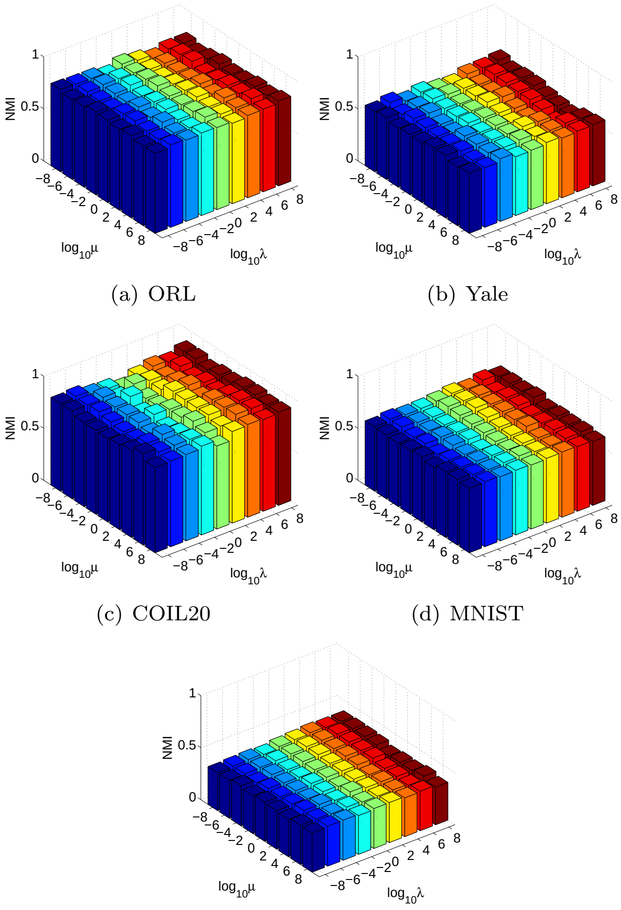
K	Accuracy (%)					
	Kmeans	GNMF	LCCF	LDMGI	USELM	UDELM
2	94.20±8.24	97.08±6.85	99.20±2.58	**99.76±1.09**	99.34±2.38	**99.76±1.09**
4	84.64±13.45	91.94±12.98	91.46±14.16	93.84±12.15	89.08±15.20	**94.77±10.32**
6	76.53±10.74	87.42±10.47	90.62±9.38	91.85±8.71	88.63±10.86	**93.67±8.11**
8	75.06±11.15	89.05±6.90	88.17±8.99	87.02±7.74	89.87±8.41	**92.40±8.17**
10	72.11±7.41	88.37±8.42	87.11±7.21	84.32±5.45	85.15±8.62	**89.03±8.21**
12	71.34±6.93	86.51±6.83	85.12±8.46	84.94±6.35	83.83±5.83	**86.86±6.93**
14	67.10±4.05	82.27±4.01	79.88±5.19	81.46±4.52	80.93±5.19	**83.56±4.23**
16	67.38±4.26	83.32±4.00	78.10±5.54	81.05±4.51	80.53±3.03	**83.69±3.59**
18	65.49±3.24	**82.84±3.68**	77.14±4.27	79.78±4.44	80.13±3.24	82.55±3.21
20	65.83	80.21	76.67	81.04	82.78	**83.68**
Avg.	73.97	86.90	85.35	86.51	86.03	**89.00**
K	Normalized Mutual Information (%)					
	Kmeans	GNMF	LCCF	LDMGI	USELM	UDELM
2	80.40±25.09	90.42±20.79	96.69±10.43	**98.82±5.30**	97.21±9.45	**98.82±5.30**
4	77.54±17.20	88.73±15.71	89.57±15.53	91.25±14.96	87.65±16.05	**92.81±12.44**
6	72.32±11.58	85.83±10.45	89.94±8.10	91.20±8.37	89.17±8.57	**93.29±6.38**
8	75.27±9.26	89.81±5.26	89.35±6.21	89.79±5.02	92.10±4.86	**93.94±5.33**
10	74.12±6.13	89.84±5.68	89.71±4.48	89.93±3.29	89.53±4.49	**92.96±3.89**
12	75.37±5.65	90.62±3.79	89.69±4.66	89.82±4.50	90.04±3.30	**92.46±3.72**
14	74.15±2.88	87.01±3.08	85.29±3.22	88.56±2.66	88.59±2.31	**90.70±2.32**
16	74.81±2.26	89.59±2.95	84.94±3.65	89.07±2.70	88.93±1.59	**91.30±1.76**
18	74.88±2.07	89.74±1.63	84.42±2.22	88.62±2.53	88.84±1.47	**91.27±1.64**
20	75.78	89.09	82.45	90.36	90.09	**91.61**
Avg.	75.46	89.07	88.21	90.74	90.22	**92.92**

Table 5: Clustering performance on MNIST.

K	Accuracy (%)					
	Kmeans	GNMF	LCCF	LDMGI	USELM	UDELM
2	91.57±7.52	94.54±6.01	98.51±1.34	99.01±0.96	98.89±1.03	**99.10±0.94**
3	77.01±15.65	79.99±17.54	82.59±14.61	93.90±8.58	90.07±15.28	**94.39±8.29**
4	70.94±11.44	77.25±12.99	76.90±9.81	87.57±11.52	85.96±10.64	**89.60±9.59**
5	68.84±10.26	75.94±11.06	69.19±7.91	83.13±11.08	79.81±13.50	**85.34±8.94**
6	64.69±7.16	71.21±10.11	64.69±7.85	80.22±9.06	75.23±11.46	**82.09±8.95**
7	59.27±5.29	65.34±5.25	60.82±7.09	73.39±7.86	68.15±7.82	**74.83±7.43**
8	58.72±3.55	65.38±5.14	59.63±5.71	72.59±7.01	64.18±7.27	**73.12±6.24**
9	55.03±2.02	60.58±3.21	54.68±4.99	70.05±3.98	60.56±3.12	**71.08±4.78**
10	54.65	61.18	51.05	67.13	60.32	**68.85**
Avg.	66.75	72.38	68.67	80.89	75.91	**82.04**
K	Normalized Mutual Information (%)					
	Kmeans	GNMF	LCCF	LDMGI	USELM	UDELM
2	64.49±23.74	76.46±21.58	90.14±7.73	92.80±6.12	92.02±6.31	**93.46±5.90**
3	54.28±15.92	65.50±17.78	63.99±16.41	83.78±13.94	79.39±16.82	**83.86±14.98**
4	54.25±10.52	65.75±11.92	62.11±11.49	79.42±9.98	75.18±10.16	**79.79±10.31**
5	53.27±8.09	66.42±8.46	58.28±8.07	76.00±9.91	72.32±10.68	**76.67±8.22**
6	51.44±5.32	64.59±7.56	54.21±7.08	72.77±7.32	70.71±8.19	**73.98±6.87**
7	51.91±3.18	61.74±3.40	54.11±4.93	70.27±4.40	66.64±5.02	**71.16±5.04**
8	52.59±2.44	62.23±4.17	54.19±4.00	70.08±4.47	65.13±4.70	**71.43±4.20**
9	50.09±2.16	59.25±2.32	51.18±3.82	67.82±2.10	61.31±2.29	**70.13±2.31**
10	49.92	59.77	49.95	64.02	59.45	**66.39**
Avg.	53.58	64.63	59.80	75.22	71.35	**76.32**

Table 6: Clustering performance on Caltech101.

K	Accuracy (%)					
	Kmeans	GNMF	LCCF	LDMGI	USELM	UDELM
2	74.77±14.25	76.46±17.49	76.21±17.39	76.45±19.12	73.91±16.97	**77.12±18.42**
3	62.13±15.79	71.73±13.30	72.07±11.51	70.39±12.59	70.67±11.29	**72.61±12.05**
4	56.95±11.96	59.81±11.45	59.96±11.68	60.18±12.40	59.27±12.09	**62.14±12.02**
5	52.88±8.59	54.03±10.23	55.08±7.89	56.33±8.86	53.73±10.23	**56.51±9.75**
6	52.09±8.11	54.24±7.41	54.37±6.50	56.61±8.40	54.39±7.87	**57.88±9.28**
7	46.29±8.19	48.27±7.13	47.65±6.12	**51.92±7.19**	48.72±7.42	51.64±6.10
8	45.04±4.71	47.64±4.70	46.09±8.06	50.16±4.46	47.26±5.24	**51.45±3.89**
9	45.26±4.63	45.84±4.20	43.86±3.63	49.73±2.50	45.86±4.64	**50.21±2.91**
10	42.08	44.88	42.74	50.46	43.86	**50.70**
Avg.	53.05	55.88	55.34	58.03	55.30	**58.92**
K	Normalized Mutual Information (%)					
	Kmeans	GNMF	LCCF	LDMGI	USELM	UDELM
2	24.50±26.61	30.87±30.99	30.62±30.35	34.55±30.61	25.37±28.93	**34.57±30.53**
3	30.68±19.20	40.21±18.67	38.26±17.19	38.40±18.12	38.42±16.63	**41.55±16.00**
4	30.28±14.50	32.45±13.75	31.27±12.94	33.59±13.58	33.25±13.62	**34.68±13.64**
5	32.69±10.31	34.00±10.26	33.10±8.75	35.58±9.98	34.10±11.44	**36.96±11.01**
6	37.03±8.21	38.62±7.71	37.53±6.78	42.49±7.66	39.43±8.50	**42.94±8.35**
7	33.14±7.57	34.14±6.95	31.73±6.49	37.98±7.23	35.31±7.86	**38.58±7.76**
8	35.37±4.40	36.04±4.87	35.40±6.47	38.80±4.44	36.32±4.75	**40.23±4.36**
9	35.84±3.14	35.81±3.31	35.70±3.28	40.46±3.18	37.19±3.92	**41.26±3.49**
10	36.02	35.26	36.61	40.28	36.86	**41.04**
Avg.	32.84	35.27	34.47	37.99	35.14	**39.09**



(a) ORL

(b) Yale

(c) COIL20

(d) MNIST

(e) Caltech101

Figure 10: Clustering normalized mutual information vs. μ and λ.

4.3.1. Data sets

Three types of emotional states (positive, neutral and negative) were evoked by watching different types of movie clips. A 62-channel electrode cap according to the extended international 10-20 and ESI NeuroScan System were used to recored the EEG signal with sampling rate 1000Hz. 15 subjects aged between 20 to 27 participated in the EEG signal collection experiment. Each of them participated three times, at the interval of about one week. The detailed description of EEG signal collection including *stimuli*, *subjects*, *experimental procedure* can be found in [26].

After preprocessing the EEG signal, the differential entropy (DE) feature defined as

$$\begin{aligned} h(X) &= \int_{-\infty}^{+\infty} \frac{-1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right) dx \\ &= \frac{1}{2}\log(2\pi e\sigma^2), \end{aligned}$$

is extracted from five frequency bands of EEG. They are δ(1-3Hz), θ(4-7Hz), α(8-13Hz), β(14-30Hz) and γ(31-50Hz). Short-time Fourier transform with 1s non-overlapping Hanning window was used to calculate the average DE features of each channel on these bands. Each frequency band of EEG signal has 62 channels and thus 310 dimensional features were obtained for each sample. S-ince the effective experimental time was about 57 minutes, we finally got about 3400 samples for each experiment. To get reliable samples, Linear Dynamic Systems was used to remove the rapid changes of EEG features.

4.3.2. Experimental settings and results

As was reported in [29], we found similar results that LDMGI performed worse in other data modalities such as text data and EEG data; therefore, the results of LDMGI on EEG signal clustering are not included. The affinity matrix based on 'Heatkernel' equation defined in (9) is reasonable for image data [39]; here we use '0-1' weight scheme for EEG signal clustering for simplicity. The parameters are set the same values as those in section 4.2.2.

In previous studies [46, 47, 26], we found that the DE features of β and γ frequency bands are more relevant to the transition of emotional states; therefore, we experiment on three types of features: β band features, γ band features and all frequency bands features.

Tables 7, 8 and 9 respectively present the clustering results of different algorithms on β, γ and all frequency bands features extracted from emotional EEG signal. The best results are shown in boldface. These results are across different subjects and different experiments. From the results, we can see that: (1) UDELM consistently performs better than the other algorithms, which means that both geometrical structure and discriminative information are helpful in exploring the regularity contained in the EEG data. UDELM is a competitive algorithm for emotional EEG data clustering. (2) The results obtained from γ band feature, which are shown in Table 8, are obviously more promising that those shown in Tables 7 and 9. This suggests that γ band is the key relevant frequency band to emotion, which is consistent with the conclusion in [46].

Table 7: Clustering performance on β frequency band features.

Metric	ACC(%)	NMI(%)
Kmeans	58.28±12.03	30.54±18.34
GNMF	60.89±9.53	32.73±16.23
LCCF	60.23±10.99	33.97±17.80
USELM	59.02±14.31	30.30±21.13
UDELM	**77.33±11.24**	**57.15±16.62**

Table 8: Clustering performance on γ frequency band features.

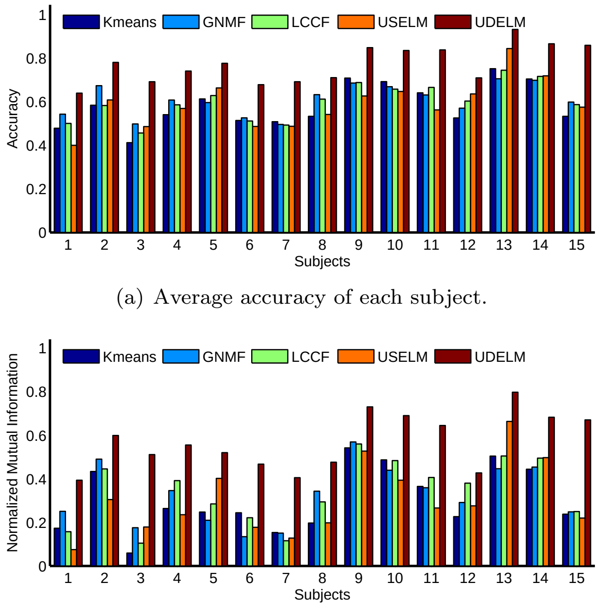
Metric	ACC(%)	NMI(%)
Kmeans	62.31±10.56	37.18±16.84
GNMF	69.13±12.06	43.02±19.30
LCCF	64.17±10.05	39.53±16.77
USELM	64.15±13.66	38.19±21.22
UDELM	**79.64±11.09**	**61.06±16.78**

Table 9: Clustering performance on all frequency bands features.

Metric	ACC(%)	NMI(%)
Kmeans	59.85±11.89	32.59±18.80
GNMF	62.73±9.72	33.48±15.90
LCCF	64.11±11.04	37.31±21.02
USELM	60.11±14.76	33.47±21.02
UDELM	**78.60±11.16**	**60.61±17.52**

For each subject, we show the clustering performance of different algorithms in Figures 11 and 12, which are respectively based on β and γ bands features. Obviously, for EEG data, the accuracy metric is more stable than the normalized mutual information metric.

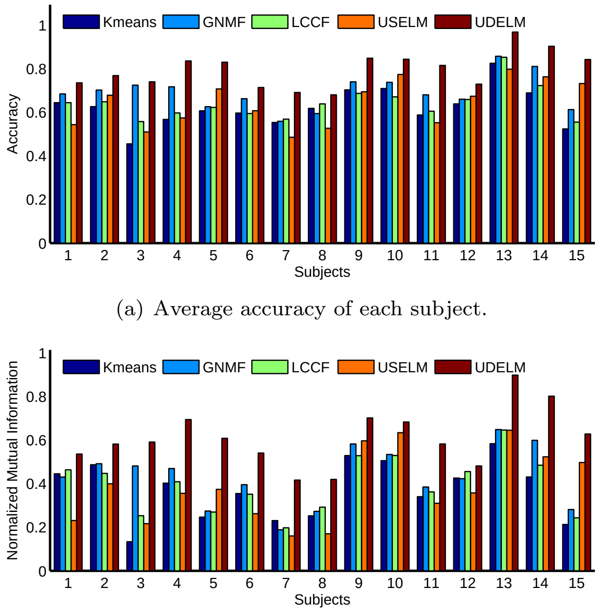
Each subject has three sets of EEG data which were collected in different times. For each time EEG data of each subject, we show the clustering results in Figure 13, in which all the frequency bands features are used.



(a) Average accuracy of each subject.

(b) Average normalized mutual information of each subject.

Figure 11: Average performance of each subject on β band feature.



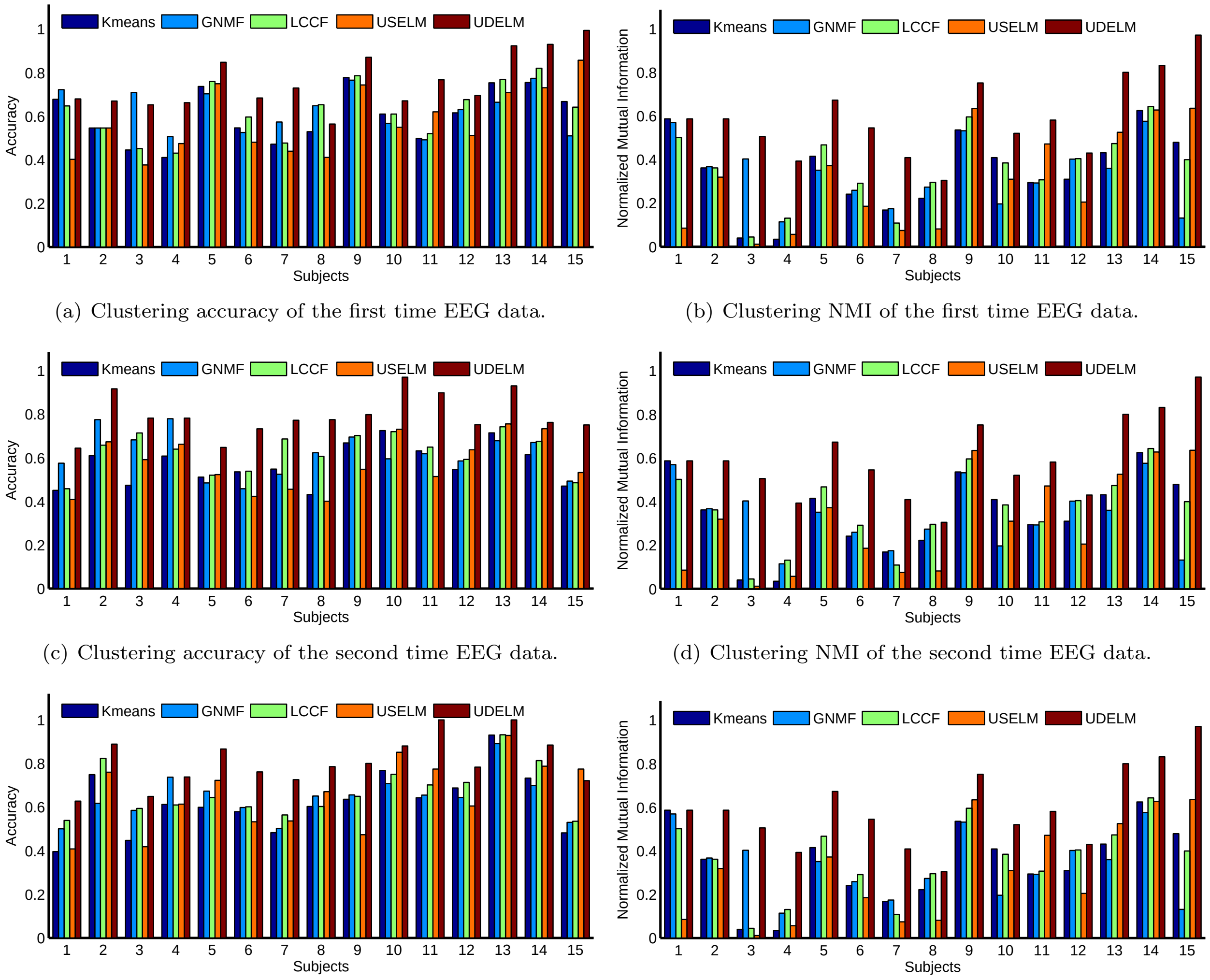
(a) Average accuracy of each subject.

(b) Average normalized mutual information of each subject.

Figure 12: Average performance of each subject on γ band feature.

5. Conclusion and future work

In this paper, we have presented a novel ELM variant for unsupervised learning, called unsupervised discriminative



(a) Clustering accuracy of the first time EEG data.

(b) Clustering NMI of the first time EEG data.

(c) Clustering accuracy of the second time EEG data.

(d) Clustering NMI of the second time EEG data.

(e) Clustering accuracy of the third time EEG data.

(f) Clustering NMI of the third time EEG data.

Figure 13: Clustering performance on all frequency bands features of emotional EEG data.

Extreme Learning Machine. UDELM takes both local geometrical structure and the global discriminative information of data into learning. As a result, UDELM can learn more effective data representation than the ordinary unsupervised ELM, which only preserves the local structure of data. Moreover, UDELM objective can be efficiently solved via generalized eigen-value decomposition. Experiments on extensive image and EEG data sets show that UDELM can obtain promising results on data clustering.

The non-negative representation has shown superior performance to mixed-sign representation in clustering [41, 48], for our future work, we will consider the non-negativity of ELM output for clustering.

Acknowledgement

This work was partially supported by the National Basic Research Program of China (No.2013CB329401), the National Natural Science Foundation of China (No.61272248), and the Science and Technology Commission of Shanghai Municipality (No.13511500200).

References

[1] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (1) (2006) 489–501.

[2] G.-B. Huang, X. Ding, H. Zhou, Optimization method based extreme learning machine for classification, Neurocomputing 74 (1) (2010) 155–163.

[3] G.-B. Huang, L. Chen, C.-K. Siew, Universal approximation using incremental constructive feedforward networks with random hidden nodes, IEEE Transactions on Neural Networks 17 (4) (2006) 879–892.

[4] R. Zhang, Y. Lan, G.-B. Huang, Z.-B. Xu, Universal approximation of extreme learning machine with adaptive growth of hidden nodes, IEEE Transactions on Neural Networks and Learning Systems 23 (2) (2012) 365–371.

[5] G.-B. Huang, H. Zhou, X. Ding, R. Zhang, Extreme learning machine for regression and multiclass classification, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 42 (2) (2012) 513–529.

[6] W. Zong, G.-B. Huang, Face recognition based on extreme learning machine, Neurocomputing 74 (16) (2011) 2541–2551.

[7] R. Minhas, A. A. Mohammed, Q. J. Wu, Incremental learning in human action recognition based on snippets, IEEE Transactions on Circuits and Systems for Video Technology 22 (11) (2012) 1529–1541.

[8] A. Iosifidis, A. Tefas, I. Pitas, Minimum class variance extreme learning machine for human action recognition, IEEE Transactions on Circuits and Systems for Video Technology 23 (11) (2013) 1968–1979.

[9] H. Yu, Y. Chen, J. Liu, G.-B. Huang, An adaptive and iterative online sequential ELM-based multi-degree-of-freedom gesture recognition system, IEEE Intelligent Systems 28 (6) (2013) 55–59.

[10] Y. Xu, Z. Y. Dong, J. H. Zhao, P. Zhang, K. P. Wong, A reliable intelligent system for real-time dynamic security assessment of power systems, IEEE Transactions on Power Systems 27 (3) (2012) 1253–1263.

[11] L.-C. Shi, B.-L. Lu, EEG-based vigilance estimation using extreme learning machines, Neurocomputing 102 (2013) 135–143.

[12] S. Samet, A. Miri, Privacy-preserving back-propagation and extreme learning machine algorithms, Data & Knowledge Engineering 79 (2012) 40–61.

[13] S. Suresh, R. Venkatesh Babu, H. Kim, No-reference image quality assessment using modified extreme learning machine classifier, Applied Soft Computing 9 (2) (2009) 541–552.

[14] S. Decherchi, P. Gastaldo, R. Zunino, E. Cambria, J. Redi, Circular-ELM for the reduced-reference assessment of perceived image quality, Neurocomputing 102 (2013) 78–89.

[15] M. Pal, A. E. Maxwell, T. A. Warner, Kernel-based extreme learning machine for remote-sensing image classification, Remote Sensing Letters 4 (9) (2013) 853–862.

[16] G. Feng, G.-B. Huang, Q. Lin, R. Gay, Error minimized extreme learning machine with growth of hidden nodes and incremental learning, IEEE Transactions on Neural Networks 20 (8) (2009) 1352–1357.

[17] H.-J. Rong, G.-B. Huang, N. Sundararajan, P. Saratchandran, Online sequential fuzzy extreme learning machine for function approximation and classification problems, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 39 (4) (2009) 1067–1072.

[18] Y. Miche, A. Sorjamaa, P. Bas, O. Simula, C. Jutten, A. Lendasse, OP-ELM: optimally pruned extreme learning machine, IEEE Transactions on Neural Networks 21 (1) (2010) 158–162.

[19] Y. Miche, M. Van Heeswijk, P. Bas, O. Simula, A. Lendasse, TROP-ELM: a double-regularized ELM using LARS and Tikhonov regularization, Neurocomputing 74 (16) (2011) 2413–2421.

[20] J. Cao, Z. Lin, G.-B. Huang, N. Liu, Voting based extreme learning machine, Information Sciences 185 (1) (2012) 66–77.

[21] Y. Peng, S. Wang, X. Long, B.-L. Lu, Discriminative graph regularized extreme learning machine and its application to face recognition, Neurocomputing 149 (2015) 340–353.

[22] G. Huang, S. Song, J. N. Gupta, C. Wu, Semi-supervised and unsupervised extreme learning machines, IEEE Transactions on Cybernetics 44 (12) (2014) 2405–2417.

[23] Q. He, X. Jin, C. Du, F. Zhuang, Z. Shi, Clustering in extreme learning machine feature space, Neurocomputing 128 (2014) 88–95.

[24] A. Akusok, D. Veganzones, K.-M. Björk, E. Séverin, P. Du Jardin, A. Lendasse, Y. Miche, ELM clustering — application to bankruptcy prediction —, in: International work conference on TIme SEries, 711–723, 2014.

[25] N. Guan, D. Tao, Z. Luo, B. Yuan, Manifold regularized discriminative nonnegative matrix factorization with fast gradient descent, IEEE Transactions on Image Processing 20 (7) (2011) 2030–2048.

[26] Y. Peng, J.-Y. Zhu, W.-L. Zheng, B.-L. Lu, EEG-based emotion recognition with manifold regularized extreme learning machine, in: Proceedings of Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 974–977, 2014.

[27] X. Shu, Y. Gao, H. Lu, Efficient linear discriminant analysis with locality preserving for face recognition, Pattern Recognition 45 (5) (2012) 1892–1898.

[28] Y. Yang, D. Xu, F. Nie, S. Yan, Y. Zhuang, Image clustering using local discriminant models and global integration, IEEE Transactions on Image Processing 19 (10) (2010) 2761–2773.

[29] P. Li, J. Bu, Y. Yang, R. Ji, C. Chen, D. Cai, Discriminative orthogonal nonnegative matrix factorization with flexibility for data representation, Expert Systems with Applications 41 (4) (2014) 1283–1293.

[30] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 888–905.

[31] A. E. Hoerl, R. W. Kennard, Ridge regression: biased estimation for nonorthogonal problems, Technometrics 42 (1) (2000) 80–86.

[32] X. He, P. Niyogi, Locality preserving projections, in: Proceedings of Advances in Neural Information Processing Systems, 153–160, 2004.

[33] T. Zhou, D. Tao, X. Wu, Manifold elastic net: a unified framework for sparse dimension reduction, Data Mining and Knowledge Discovery 22 (3) (2011) 340–371.

[34] S. T. Roweis, L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.

[35] J. B. Tenenbaum, V. De Silva, J. C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.

[36] M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, in: Proceedings of Advances in Neural Processing Systems, 585–591, 2001.

[37] Z. Zhang, H. Zha, Principal manifolds and nonlinear dimensionality reduction via tangent space alignment, SIAM Journal on Scientific Computing 26 (1) (2004) 313–338.

[38] M. Belkin, P. Niyogi, V. Sindhwani, Manifold regularization: A geometric framework for learning from labeled and unlabeled examples, Journal of Machine Learning Research 7 (2006) 2399–2434.

[39] D. Cai, X. He, J. Han, T. S. Huang, Graph regularized nonnegative matrix factorization for data representation, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (8) (2011) 1548–1560.

[40] D. Cai, X. He, J. Han, Locally consistent concept factorization for document clustering, IEEE Transactions on Knowledge and Data Engineering 23 (6) (2011) 902–913.

[41] Y. Yang, H. T. Shen, F. Nie, R. Ji, X. Zhou, Nonnegative spectral clustering with discriminative regularization, in: Proceedigns of the Twenty-Fifth AAAI Conference on Artificial Intelligence, 555–560, 2011.

[42] J. Ye, Z. Zhao, M. Wu, Discriminative k-means for clustering, in: Proceedings of Advances in Neural Information Processing Systems, 1649–1656, 2007.

[43] K. Fukunaga, Introduction to statistical pattern recognition, Academic press, 1990.

[44] M. D. Plummer, L. Lovász, Matching theory, Elsevier, 1986.

[45] A. Strehl, J. Ghosh, Cluster ensembles—a knowledge reuse framework for combining multiple partitions, Journal of Machine Learning Research 3 (2003) 583–617.

[46] M. Li, B.-L. Lu, Emotion classification based on gamma-band EEG, in: Proceedings of Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 1223–1226, 2009.

[47] R.-N. Duan, J.-Y. Zhu, B.-L. Lu, Differential entropy feature for EEG-based emotion classification, in: Proceedings of International IEEE/EMBS Conference on Neural Engineering, 81–84, 2013.

[48] Z. Li, Y. Yang, J. Liu, X. Zhou, H. Lu, Unsupervised feature selection using nonnegative spectral analysis., in: Proceedings of Twenty-Sixth AAAI Conference on Artificial Intelligence, 1026–1032, 2012.