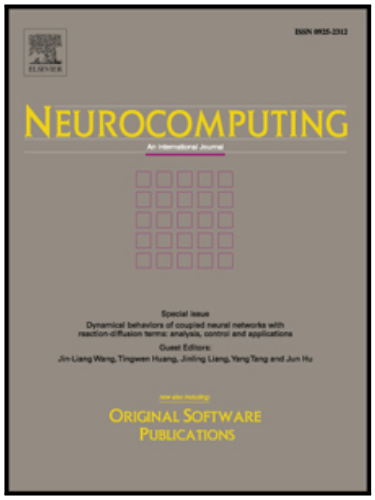
Communicated by Dr. Deng Cheng

Accepted Manuscript



Orthogonal extreme learning machine for image classification

Yong Peng, Wanzeng Kong, Bing Yang

PII: S0925-2312(17)30921-9
DOI: 10.1016/j.neucom.2017.05.058
Reference: NEUCOM 18476

To appear in: *Neurocomputing*

Received date: 7 August 2016
Revised date: 19 December 2016
Accepted date: 23 May 2017

Please cite this article as: Yong Peng, Wanzeng Kong, Bing Yang, Orthogonal extreme learning machine for image classification, *Neurocomputing* (2017), doi: 10.1016/j.neucom.2017.05.058

Orthogonal extreme learning machine for image classification

Yong Peng[a,b,c,*], Wanzeng Kong[a], Bing Yang[a]

[a] *MOE Key Laboratory of Complex Systems Modeling and Simulation, School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China*
[b] *Key Laboratory of Image and Video Understanding for Social Safety, Nanjing University of Science and Technology, Nanjing 210094, China*
[c] *Guangxi High School Key Laboratory of Complex System and Computational Intelligence, Nanning 530006, China*

Abstract

Extreme learning machine (ELM) is an emerging learning algorithm for the generalized single hidden layer feedforward neural networks in which the parameters of hidden units are randomly generated and thus the output weights can be analytically calculated. From the hidden to output layer, ELM essentially learns the output weight matrix based on the least squares regression formula which can be used for both classification/regression and dimensionality reduction. In this paper, we impose the orthogonal constraint on the output weight matrix and then formulate an orthogonal extreme learning machine (OELM) model, which produces orthogonal basis functions and can have more locality preserving power from ELM feature space to output layer than ELM. Since the locality preserving ability is potentially related to the discriminating power, the OELM is expect to have more discriminating power than ELM. Considering the case that the number of hidden units is usually greater than the number of classes, we propose an effective method to optimize the OELM objective by solving an orthogonal procrustes problem. Experiments by pairwisely comparing OELM with ELM on three widely used image data sets show the effectiveness of learning orthogonal mapping especially when given only limited training samples.

*Corresponding author
Email address: yongpeng@hdu.edu.cn (Yong Peng)

Keywords: Extreme learning machine, Orthogonal constraint, Orthogonal procrustes problem, Image classification

1. Introduction

Extreme learning machine [1] has been proved as an efficient and effective method to train single hidden layer feed neural networks (SLFNs), providing us an unified framework for both multi-class classification and regression tasks. The basic ELM model can be simply seen as a random feature mapping followed by the least squares regression formula. The main contribution of ELM to general SLFNs is that the parameters of hidden units, including the input weights between the input layer and hidden layer as well as the biases of hidden units, can be randomly generated, which leads to the analytical determination of the output weights between the hidden layer and output layer. Such improvement greatly alleviates the burden of weight tuning caused by the widely used back-propagation algorithms and thus guarantees the fast learning speed of ELM. As a variant of SLFNs, though the mathematical formula of ELM is simple, the universal approximation capacity [2, 3] can be also kept. Furthermore, the rationality of the randomly generated input weights and biases was analyzed by some recently published studies [4, 5]. ELM fills gaps among many types of SLFNs such as feedforward networks (e.g., sigmoid networks), RBF networks, SVM (considered as a special type of SLFNs), polynomial networks and proposes that it need not have different learning algorithms for different SLFNs if universal approximation and classification capabilities are considered [2, 6, 7]. Further, ELM theories and philosophy show that some earlier learning theories such as ridge regression theory, Bartlett's neural network generalization performance theory and SVM's maximal margin are actually consistent in machine learning [8, 9]. Inspired by deep learning but different from it, the hierarchical models using ELM as building block do not require intensive tuning in hidden layers and hidden units and also obtain amazing performance [10, 11]. Due to the success of ELM in diverse applications, ELM research has been a hotspot

in machine learning communities and many studies are conducted from many aspects such as theoretical investigation [4, 5], model improvements [12, 11] and applications [13, 14]. Some of recent progresses were briefly reviewed in [15, 16].

From the hidden layer to output layer, ELM essentially learns the output weight matrix based on the least squares regression formula. Therefore, many approaches were proposed to do discriminant analysis based on the least squares regression. The central task is to find a proper transformation matrix to minimize the sum-of-squares error function, which will be further used for dimensionality reduction or classification. Xiang *et al.* proposed a framework of discriminative least squares regression for multiclass classification whose idea is to utilize the ε-dragging to enlarge the distance of samples from different classes [17]. Similar work conducted by Zhang *et al.* aims to directly learn regression targets from data which can better evaluate the classification error than conventional predefined regression targets [18]. In most cases, ELM is viewed as classifier in which the hidden layer data representation (ELM feature space) is projected to output layer (label space), we expect to learn proper output weight matrix (transformation matrix) to make ELM more effective in classification. To this end, many efforts were made to impose different properties on the output weight matrix. Peng *et al.* proposed to enhance the label consistency property of ELM and formulated the graph regularized extreme learning machine which shows excellent performance in face recognition [19] and EEG-based emotion recognition [20]. Shi *et al.* introduced the elastic net regularization into ELM which can simultaneously bring the sparsity of output weight matrix and avoid the singularity problem [21]. Among different existing strategies, orthogonal constraint on transformation matrix has been widely employed in both subspace learning and least squares-based classification, which shows excellent performance in both situations. Cai *et al.* proposed the orthogonal locality preserving projection (OLPP) method which produces orthogonal basis functions and can have more locality preserving power than LPP [22]. Since it has been shown that the locality preserving power is directly related to the discriminating power, OLPP obtains better performance than LPP [23]. In [24], Nie

et al. showed that the orthogonal least squares discriminant analysis is better than the basic counterpart without orthogonal constraint. Similar work was conducted to do feature extraction based on orthogonal least squares regression [25]. Motivated by these studies, in this paper, we propose to learn orthogonal output weight matrix from hidden layer to output layer which is expect to the transformation matrix under the orthogonal constraint can preserve more structure information between these two layers and thus have more discriminating power for classification.

The remainder of this paper is organized as follows. Section 2 gives a brief description of the basic ELM model. The model formulation, optimization method, convergence as well as computational complexity of the proposed OELM are detailed in Section 3. Experimental studies are conducted in Section 4 to show the effectiveness of OELM. Section 5 concludes the whole paper.

2. Extreme learning machine

Suppose we have n labeled training samples $\{(\mathbf{x}_i, \mathbf{y}_i), i = 1, \ldots, n\}$ where each sample $\mathbf{x}_i \in \mathbb{R}^{d\times 1}$ and its corresponding label vector $\mathbf{y}^i \in \mathbb{R}^{1\times c}$ (c is the number of classes). If $\mathbf{x}_i$ is labeled as class p, then the p-th element of $\mathbf{y}^i$ is 1 and the other elements of $\mathbf{y}^i$ are 0. Consider a SLFN with input weight matrix $\mathbf{A} \in \mathbb{R}^{d\times k}$, hidden bias vector $\mathbf{b} \in \mathbb{R}^{k\times 1}$ and output weight matrix $\mathbf{W} \in \mathbb{R}^{k\times m}$, where k is the number of hidden units. For an input vector $\mathbf{x}$, the output of this SLFN can be represented as

$$\mathbf{f}(\mathbf{x}) = \sum\nolimits_{i=1}^{k} G_i(\mathbf{x}, \mathbf{a}_i, b_i) \cdot \mathbf{w}^i, \quad \mathbf{a}_i \in \mathbf{R}^{d\times 1},\ \mathbf{w}^i \in \mathbb{R}^{1\times m}, \tag{1}$$

with each $h_i(\mathbf{x}) = G(\mathbf{x}, \mathbf{a}_i, b_i)$, $i = 1, \ldots, k$ is the output of the i-th hidden unit. $G(\cdot)$ denotes the activation function, $\mathbf{a}_i$ is the input weight vector connecting the input layer to the i-th hidden unit, b_i is the bias weight of the i-th hidden unit, and $\mathbf{w}_i$ is the weight vector connecting the i-th hidden unit to the output units. $G(\cdot)$ may have various forms such as the sigmoid function

$$G(\mathbf{x}, \mathbf{a}_i, b_i) = \frac{1}{1 + \exp(-(\mathbf{a}_i^T\mathbf{x} + b_i))}. \tag{2}$$

and the Hard-limit function

$$G(\mathbf{x}_i, \mathbf{a}_i, b_i,) = \begin{cases} 1, & \text{if } \mathbf{a}_i^T \cdot \mathbf{x}_i - b_i \geq 0 \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

In this way, the hidden representation of all training samples can be written in a compact matrix as

$$\mathbf{H} = \begin{bmatrix} G(\mathbf{x}_1, \mathbf{a}_1, b_1) & \cdots & G(\mathbf{x}_1, \mathbf{a}_k, b_k) \\ \vdots & \ddots & \vdots \\ G(\mathbf{x}_n, \mathbf{a}_1, b_1) & \cdots & G(\mathbf{x}_n, \mathbf{a}_k, b_k) \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{h}(\mathbf{x}_1) \\ \vdots \\ \mathbf{h}(\mathbf{x}_n) \end{bmatrix}. \tag{4}$$

Using the label matrix $\mathbf{Y} = [\mathbf{y}^1; \cdots; \mathbf{y}^l]$ as the regression target, it is obvious that we have $m = c$.

ELM theory aims to reach the smallest training error as well as the norm of the output weight matrix

$$\min_{\mathbf{W}} \ \|\mathbf{HW} - \mathbf{Y}\|^2 + \lambda \|\mathbf{W}\|^2, \tag{5}$$

where $\lambda > 0$ is a regularization parameter. Setting the derivative of objective (5) w.r.t. $\mathbf{W}$ to zero, we obtain the analytical solution to ELM as

$$\mathbf{W}^* = (\mathbf{H}^T\mathbf{H} + \lambda \mathbf{I})^{-1}\mathbf{H}^T\mathbf{Y}. \tag{6}$$

3. Orthogonal ELM

3.1. Model formulation and optimization

By introducing the orthogonal constraint, we formulate the objective of orthogonal ELM as

$$\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \|\mathbf{HW} - \mathbf{Y}\|^2, \tag{7}$$

where $\mathbf{H} \in \mathbb{R}^{n\times k}$, $\mathbf{W} \in \mathbb{R}^{k\times c}$ and $\mathbf{Y} \in \mathbb{R}^{n\times c}$. Under the orthogonal constraint, the data will be projected to an orthogonal subspace where the data metric structure can be preserved. Some properties of OELM will be discussed in section 3.3 in detail. This section focuses on its model formulation as well as the optimization method.

Since $k > c$, objective (7) is an unbalanced orthogonal procrustes problem which is difficult to obtain the solution directly due to the orthogonal constraint. In this paper, we propose an iterative method to optimize objective (7) based on the following theorem.

Theorem 1. *The optimal solution to unbalanced orthogonal procrustes (UOP) problem shown in (7) is equivalent to the optimal solution to the following problem*

$$\min_{\widetilde{\mathbf{W}}^T\widetilde{\mathbf{W}}=\mathbf{I},\mathbf{Y}_1} \|\mathbf{H}\widetilde{\mathbf{W}} - [\mathbf{Y}, \mathbf{Y}_1]\|^2, \tag{8}$$

where $\widetilde{\mathbf{W}} = [\mathbf{W}, \mathbf{W}_1] \in \mathbb{R}^{k\times k}$, $\mathbf{W}_1 \in \mathbb{R}^{k\times(k-c)}$ *and* $\mathbf{Y}_1 \in \mathbb{R}^{n\times(k-c)}$.

Proof. From the objective (8), we know that the optimal solution to $\mathbf{Y}_1$ is $\mathbf{H}\mathbf{W}_1$. Substituting $\mathbf{Y}_1 = \mathbf{H}\mathbf{W}_1$ into (8), we arrive at

$$\min_{\widetilde{\mathbf{W}}^T\widetilde{\mathbf{W}}=\mathbf{I}} \|\mathbf{H}\widetilde{\mathbf{W}} - [\mathbf{Y}, \mathbf{H}\mathbf{W}_1]\|^2, \tag{9}$$

which is equivalent to

$$\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \|\mathbf{H}\mathbf{W} - \mathbf{Y}\|^2. \tag{10}$$

This completes the proof. □

According to Theorem 1, we can simply solve the problem (8) instead of directly solving the UOP problem (7). However, minimizing (8) jointly with repsect to $\widetilde{\mathbf{W}}$ and $\mathbf{Y}_1$ is difficult and thus we alternately update one variable while fixing the other.

When fixing $\mathbf{Y}_1$, $\mathbf{H} \in \mathbb{R}^{n\times k}$ and $[\mathbf{Y}, \mathbf{Y}_1] \in \mathbb{R}^{n\times k}$, problem (8) will be simplified as the following balanced orthogonal procrustes problem

$$\min_{\widetilde{\mathbf{W}}^T\widetilde{\mathbf{W}}=\mathbf{I}} \|\mathbf{H}\widetilde{\mathbf{W}} - [\mathbf{Y}, \mathbf{Y}_1]\|^2, \tag{11}$$

which can be solved by singular value decomposition (SVD). Suppose that the SVD of $\mathbf{H}^T[\mathbf{Y}, \mathbf{Y}_1]$ is

$$\mathbf{H}^T[\mathbf{Y}, \mathbf{Y}_1] = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \tag{12}$$

where $\mathbf{U} \in \mathbb{R}^{k\times k}$, $\mathbf{\Sigma} \in \mathbb{R}^{k\times k}$, $\mathbf{V} \in \mathbb{R}^{k\times k}$ and the solution to problem (11) is

$$\widetilde{\mathbf{W}} = \mathbf{U}\mathbf{V}^T. \tag{13}$$

The detailed derivation of above optimization method can be found in Appendix. Once $\widetilde{\mathbf{W}}$ is obtained, the output weight matrix $\mathbf{W}$ can be formed by the first c columns of matrix $\widetilde{\mathbf{W}}$.

When fixing $\widetilde{\mathbf{W}}$, problem (8) can be simplified to the following problem

$$\min_{\mathbf{Y}_1} \|\mathbf{H}[\mathbf{W}, \mathbf{W}_1] - [\mathbf{Y}, \mathbf{Y}_1]\|^2. \tag{14}$$

Obviously, the solution to (14) is

$$\mathbf{Y}_1 = \mathbf{H}\mathbf{W}_1, \tag{15}$$

where $\mathbf{W}_1$ is the last $k - c$ columns of matrix $\widetilde{\mathbf{W}}$ in (13).

Based on the above analysis, we summarize the optimization to the objective function (8) of variant orthogonal extreme learning machine in Algorithm 1.

Algorithm 1 Optimization to objective (8)

Input: Training data $\mathbf{X} \in \mathbb{R}^{d\times n}$ and label $\mathbf{Y} \in \mathbb{R}^{n\times c}$;

Output: Output weight matrix $\mathbf{W} \in \mathbb{R}^{k\times c}$.

1: Generate the input weight matrix $\mathbf{A}$ and hidden bias vector $\mathbf{b}$;
2: Calculate the hidden representation $\mathbf{H}$ based on (2);
3: Centralize $\mathbf{H}$;
4: Initialize $\mathbf{Y}_1 = \mathbf{0}^{n\times(k-c)}$;
5: **while** not converge **do**
6: Update $\widetilde{\mathbf{W}}$ based on (13) and $\mathbf{W} = \widetilde{\mathbf{W}}(:, 1:c)$, $\mathbf{W}_1 = \widetilde{\mathbf{W}}(:, c+1:k)$;
7: Update $\mathbf{Y}_1$ based on (15);
8: **end while**

3.2. Convergence and computational complexity

Below we give the analysis on the convergence of Algorithm 1 and its computational complexity.

Theorem 2. *The updating rules in Algorithm 1 will monotonically decrease the objective (8) in each iteration.*

Proof. In t-th iteration, we fix $\mathbf{Y}_1^{(t)}$ and update $\widetilde{\mathbf{W}}^t$ based on (13)

$$\widetilde{\mathbf{W}}^{(t+1)} = \arg\min_{\widetilde{\mathbf{W}}^T\widetilde{\mathbf{W}}=\mathbf{I}} \|\mathbf{H}\widetilde{\mathbf{W}} - [\mathbf{Y}, \mathbf{Y}_1^{(t)}]\|^2, \tag{16}$$

which indicates that

$$\|\mathbf{H}\widetilde{\mathbf{W}}^{(t+1)} - [\mathbf{Y}, \mathbf{Y}_1^{(t)}]\|^2 \leq \|\mathbf{H}\widetilde{\mathbf{W}}^{(t)} - [\mathbf{Y}, \mathbf{Y}_1^{(t)}]\|^2. \tag{17}$$

Then we fix $\widetilde{\mathbf{W}}^{(t+1)}$ and update $\mathbf{Y}_1^{(t)}$ based on (15) as

$$\|\mathbf{H}\widetilde{\mathbf{W}}^{(t+1)} - [\mathbf{Y}, \mathbf{Y}_1^{(t+1)}]\|^2 \leq \|\mathbf{H}\widetilde{\mathbf{W}}^{(t+1)} - [\mathbf{Y}, \mathbf{Y}_1^{(t)}]\|^2. \tag{18}$$

By combining (17) and (18), we arrive at

$$\|\mathbf{H}\widetilde{\mathbf{W}}^{(t+1)} - [\mathbf{Y}, \mathbf{Y}_1^{(t+1)}]\|^2 \leq \|\mathbf{H}\widetilde{\mathbf{W}}^{(t)} - [\mathbf{Y}, \mathbf{Y}_1^{(t)}]\|^2. \tag{19}$$

According to (19), we know that Algorithm 1 will monotonically decrease the objective of OELM shown in (8) in each iteration. □

We can see that the main complexity of ELM in obtaining $\mathbf{W}$ is to compute the inverse of a $k \times k$ matrix, that is $\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}$. As in most cases, the number of hidden units k can be much smaller than the number of training samples n: $k \ll n$, and thus the computational cost reduces dramatically in comparison with LS-SVM and PSVM which needs to compute the inverse of a $n \times n$ matrix [8]. Therefore, ELM has $O(k^3)$ complexity. Below we analyze the complexity of the optimization method to proposed OELM. From Algorithm 1, we know that the main computational complexity is consumed by the loop. In each iteration, we need to respectively update $\mathbf{W}$ and $\mathbf{Y}_1$. When updating $\mathbf{W}$, we need to do SVD decomposition on a $k \times k$ matrix $\mathbf{H}^T[\mathbf{Y}, \mathbf{Y}_1]$ which has $O(k^3)$ complexity. When updating $\mathbf{Y}_1$, we need to a multiplication of a $n \times k$ matrix $\mathbf{H}$ and a $k \times (k-c)$ matrix $\mathbf{W}_1$, which has $O(nk(k-c))$ complexity. In real applications, the number of classes c is usually greatly smaller than the number of hidden neurons k; therefore, such complexity approximates to $O(nk^2)$. Suppose that algorithm 1 needs t iterations to converge and then the total computational complexity is $O(t(k^3 + nk^2))$.

3.3. Disscussion

This section discusses 1) the connection between OELM and adaptive least squares method and 2) the metric structure preserving ability from hidden layer to output layer by orthogonal constraint.

First we give a viewpoint of OELM to provide further insight into its effectiveness, that is, the orthogonal constraint least squares method can be viewed as another adaptive orthogonal least squares method. Suppose that the hidden layer representation of data in ELM feature space $\{\mathbf{h}_1, \cdots, \mathbf{h}_n\}$ are centered, the least squares method is to solve the following problem

$$\min_{\mathbf{W}} \|\mathbf{HW} - \mathbf{Y}\|^2. \tag{20}$$

It can be proved that the optimal solution $\mathbf{W}$ to problem (20) is also the solution to the traditional LDA when $\mathbf{Y}$ is the class indicator matrix. When we constrain the projection $\mathbf{W}$ to be orthogonal ($\mathbf{W}^T\mathbf{W} = \mathbf{I}$), note that the scale of the regression values of $\mathbf{Y}$ is fixed while the scale of the data $\mathbf{H}$ can be arbitrary, the regression error could be large. Here we consider an adaptive orthogonal least squares method, which is to solve the following problem

$$\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I},\mathbf{R}} \|\mathbf{HW} - \mathbf{YR}\|^2, \tag{21}$$

where $\mathbf{R}$ is a squared matrix, which is optimized to reduce the scale between $\mathbf{H}$ and $\mathbf{Y}$. By setting the derivative w.r.t. $\mathbf{R}$ to zero, we have $\mathbf{R} = (\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{HW}$. Substituting it to problem (21), we have

$$\begin{aligned} &\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \|\mathbf{HW} - \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{HW}\|^2 \\ =&\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \mathrm{Tr}(\mathbf{W}^T(\mathbf{H}^T\mathbf{H} - \mathbf{H}^T\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{H})\mathbf{W}), \end{aligned} \tag{22}$$

where $\mathbf{Y}$ is the class indicator matrix, then $\mathbf{H}^T\mathbf{H} - \mathbf{H}^T\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{H} = \mathbf{S}_t - \mathbf{S}_b = \mathbf{S}_w$. Therefore, the orthogonal constraint least squares method can be also viewed as an adaptive orthogonal least squares method to solve (21).

Below we give analysis on the metric structure preserving ability of OELM. Once the output weight matrix $\mathbf{W}$ is computed, the Euclidean distance between

two data points in the reduced space (denoted as $\hat{\mathbf{y}}_i, \hat{\mathbf{y}}_j$) can be computed as follows

$$\begin{aligned} \mathrm{dist}(\hat{\mathbf{y}}_i, \hat{\mathbf{y}}_j) &= \|\hat{\mathbf{y}}_i - \hat{\mathbf{y}}_j\| = \|\mathbf{W}^T\mathbf{h}_i^T - \mathbf{W}^T\mathbf{h}_j^T\| = \|\mathbf{W}^T(\mathbf{h}_i^T - \mathbf{h}_j^T)\| \\ &= \sqrt{(\mathbf{h}_i - \mathbf{h}_j)\mathbf{W}\mathbf{W}^T(\mathbf{h}_i - \mathbf{h}_j)^T}. \end{aligned} \tag{23}$$

Since $\mathbf{W}^T\mathbf{W} = \mathbf{I}$, we have $\mathbf{W}^T = \mathbf{W}^{-1}$ and $(\mathbf{W}^T)^T\mathbf{W}^T = \mathbf{W}\mathbf{W}^T = \mathbf{W}\mathbf{W}^{-1} = \mathbf{I}$ and then $\mathrm{dist}(\hat{\mathbf{y}}_i, \hat{\mathbf{y}}_j) = \|\mathbf{h}_i - \mathbf{h}_j\|$. This means the orthogonal matrix $\mathbf{W}$ can naturally preserve the metric structure from the ELM feature space to reduced subspace space. Since the locality preserving power is potentially related to the discriminating power, the OELM is expected to have more discriminating power than ELM [22, 23].

4. Experimental studies

4.1. Experimental settings and datasets

In this section, we conduct pairwise comparison between OELM and ELM (with ℓ_2 regularization) to show the effectiveness of the orthogonal constraint on output weight matrix. The activation function for ELM is the 'sigmoid' function, and the number of hidden neurons is set as three times of the input dimension of data. The regularization parameter for ELM is searched from candidates $2.^{\{-10,\cdots,10\}}$.

The properties of the three images data sets used in our experiments are descried as follows

- UMIST. This face data set contains 20 subjects and totally 575 face images. All images are cropped and resized to 28×23 pixels per image.
- ORL. It contains 10 different images of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions and facial details. All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position. The size of each image is 32×32 pixels.

- COIL20. It contains 20 objects. The images of each objects were taken 5 degrees apart as the object is rotated on a turntable and each object has 72 images. The size of each image is 32×32 pixels, with 256 grey levels per pixel.

For UMIST, we randomly select 3, 5, 10 and 15 samples from each class as training samples while the rest samples are used for testing. This process was repeated for 20 times, and the indices for partitioning training and testing samples are kept the same for both ELM and OELM. For ORL, the number of training samples are respectively 2, 3, 4 and 5. For COIL20, the number of training samples are respectively 2, 4, 6 and 8.

4.2. Results and analysis

The experimental results of the pairwise comparison between ELM and OELM on UMIST, ORL and COIL20 are respectively shown in Tables 1, 2 and 3. From the results, we can find that OELM can obtain higher accuracies than ELM in most cases especially when given limited training samples. This means that if there is no sufficient data to learn the transformation matrix, orthogonality is a reasonable constraint to map the data from ELM feature space to label space. When gradually given more training samples, the desirable property of transformation matrix can be learned, and thus the accuracy gap between ELM and OELM decreases.

To illustrate the statistical difference between OELM and ELM, we did the paired student' t-test on these data sets. Here the hypothesis is "the classification accuracy obtained by OELM is greater than that obtained by ELM". Each test run on two accuracy sequences, which are obtained from the 20 splits by OELM and ELM. The results of statistical tests are reported in the bottom line of Tables 1, 2 and 3. In each entity, "1" means the hypothesis is correct (true) with probability 0.95, and "0" means that the hypothesis is wrong (false) with probability 0.95. For example, on UMIST data set (see Table 1), the decision "76.30>73.32" is correct with probability 0.95. In summary, we see decision

Table 1: Pairwise comparison between ELM and OELM on UMIST.

#	3 Train		5 Train		10 Train		15 Train	
	ELM	OELM	ELM	OELM	ELM	OELM	ELM	OELM
1	75.53	**76.89**	87.37	**89.89**	93.87	**95.20**	**98.18**	**98.18**
2	71.07	**74.56**	84.00	**85.05**	94.67	**95.73**	97.09	**97.45**
3	71.65	**73.01**	82.32	**86.11**	96.80	**97.33**	98.18	**98.55**
4	70.87	**73.98**	84.63	**88.00**	96.80	**97.33**	98.18	**98.91**
5	70.49	**74.76**	83.37	**85.89**	**94.67**	**94.67**	94.18	**94.55**
6	73.01	**76.89**	90.74	91.79	90.67	**93.07**	**98.18**	97.45
7	70.29	**74.17**	84.84	**86.11**	**96.00**	95.73	**98.18**	**98.18**
8	74.37	**78.45**	83.79	**86.53**	96.53	**96.80**	96.73	**97.45**
9	71.65	**76.12**	84.63	**87.58**	93.60	**94.67**	**98.91**	97.82
10	68.16	**71.46**	85.89	**86.74**	91.47	**92.53**	**97.09**	96.73
11	74.17	**76.12**	87.58	**89.26**	95.20	**96.00**	**96.73**	97.45
12	74.17	**79.03**	91.37	**92.84**	94.40	**95.20**	98.18	**98.55**
13	77.67	**80.39**	88.42	**89.26**	93.87	**94.67**	98.55	**98.91**
14	67.77	**70.87**	85.68	**88.42**	**94.13**	93.60	97.45	**98.18**
15	73.01	**75.15**	86.74	**87.16**	93.07	**94.93**	**97.82**	97.09
16	79.22	**80.58**	89.26	**90.53**	95.20	**96.00**	**98.91**	**98.91**
17	74.37	**78.06**	87.37	**89.47**	94.93	**95.73**	97.09	**97.45**
18	72.62	**74.95**	87.58	**90.11**	93.07	**93.33**	**97.45**	**97.45**
19	80.19	**82.14**	82.74	**85.26**	96.27	**96.53**	**98.91**	98.18
20	76.12	**78.45**	84.84	**86.95**	93.60	**94.13**	**96.73**	96.00
MEAN	73.32	**76.30**	86.16	**88.15**	94.44	**95.16**	97.63	**97.67**
±STD	±3.32	±3.01	±2.55	±2.19	±1.65	±1.36	±1.10	±1.06
t-test	1		1		1		0	

Table 2: Pairwise comparison between ELM and OELM on ORL.

#	2 Train		3 Train		4 Train		5 Train	
	ELM	OELM	ELM	OELM	ELM	OELM	ELM	OELM
1	83.44	**86.25**	88.57	**90.00**	**95.42**	95.00	96.00	**96.50**
2	84.38	**87.50**	89.29	**90.36**	91.67	**92.92**	**97.00**	96.50
3	81.88	**85.00**	83.57	**86.79**	94.58	**95.83**	96.50	**97.50**
4	82.81	**84.69**	90.36	**91.43**	94.17	**94.58**	**92.00**	**92.00**
5	76.88	**78.44**	88.57	**91.43**	90.83	**91.67**	**96.00**	94.00
6	80.94	**82.50**	87.14	**89.64**	94.58	**95.00**	94.50	**95.50**
7	81.88	**83.75**	88.21	**90.00**	**91.67**	91.25	94.00	**95.50**
8	82.50	**84.38**	89.29	**90.71**	92.08	**92.92**	97.00	**97.50**
9	79.38	**80.00**	91.07	**92.86**	**93.33**	92.08	**94.50**	93.50
10	87.50	**90.00**	**91.79**	**91.79**	**93.75**	93.33	94.00	**97.00**
11	81.88	**85.31**	87.86	**90.71**	90.00	**91.25**	**97.50**	**97.50**
12	83.75	**86.25**	86.07	**87.86**	**93.33**	**93.33**	**97.50**	97.00
13	81.25	**81.56**	88.57	**90.36**	94.17	**97.08**	**96.00**	93.50
14	79.69	**82.19**	87.50	**88.93**	91.25	**92.50**	**96.00**	95.50
15	81.56	**83.75**	89.29	**90.71**	90.00	**92.08**	**98.00**	97.50
16	81.56	**82.50**	89.29	**91.43**	92.50	**93.33**	95.50	**96.00**
17	84.06	**85.00**	87.86	**89.64**	**94.17**	93.75	95.00	**96.50**
18	79.69	**82.81**	**89.29**	88.21	95.00	**96.25**	**94.00**	93.50
19	79.06	**80.63**	89.29	**90.00**	**90.00**	89.58	**93.00**	**93.00**
20	83.13	**85.63**	87.14	**88.93**	**94.58**	94.17	**95.00**	**95.00**
MEAN	81.86	**83.91**	88.50	**90.09**	92.85	**93.40**	95.45	**95.53**
±STD	±2.31	±2.70	±1.78	±1.45	±1.79	±1.87	±1.60	±1.73
t-test	1		1		1		0	

Table 3: Pairwise comparison between ELM and OELM on COIL20.

#	2 Train		4 Train		6 Train		8 Train	
	ELM	OELM	ELM	OELM	ELM	OELM	ELM	OELM
1	74.29	**76.79**	84.63	**86.10**	87.73	**88.64**	89.45	**91.80**
2	70.64	**72.07**	77.94	**80.00**	87.27	**89.09**	90.39	**92.34**
3	71.43	**73.86**	83.16	**84.41**	87.35	**89.32**	90.78	**91.02**
4	69.21	**70.14**	81.62	**84.26**	84.77	**86.89**	92.66	**92.97**
5	68.86	**69.93**	83.68	**85.44**	85.30	**88.79**	86.56	**88.67**
6	71.86	**73.64**	82.13	**84.49**	86.74	**89.02**	91.09	**92.11**
7	73.43	**77.14**	81.76	**83.16**	87.80	**89.47**	**89.84**	**89.84**
8	**72.79**	72.71	79.85	**83.38**	84.39	**88.11**	93.05	**94.30**
9	69.50	**71.07**	**82.94**	82.79	86.97	**89.92**	90.63	**93.59**
10	70.79	**72.79**	84.26	**87.21**	88.18	**90.68**	89.77	**91.64**
11	73.79	**75.21**	80.74	**83.90**	85.30	**88.26**	92.58	**93.28**
12	68.14	**69.86**	82.87	**83.68**	85.98	**87.50**	88.13	**92.27**
13	67.29	**69.57**	**83.90**	83.75	88.48	**89.02**	92.81	**95.63**
14	71.36	**74.64**	**83.01**	82.79	89.77	**92.20**	90.39	**91.80**
15	71.93	**74.29**	78.53	**83.75**	86.21	**87.95**	90.16	**91.02**
16	73.21	**76.00**	82.13	**84.63**	84.85	**86.52**	90.31	**91.17**
17	69.79	**72.71**	83.01	**85.51**	86.14	**89.92**	91.95	**92.34**
18	67.43	**69.00**	80.88	**83.01**	83.64	**86.89**	90.00	**91.48**
19	70.36	**71.07**	83.38	**84.26**	88.56	**90.68**	89.14	**91.88**
20	71.57	**72.64**	83.38	**83.75**	87.80	**88.71**	91.33	**91.56**
MEAN	70.88	**72.76**	82.19	**84.01**	86.66	**88.88**	90.55	**92.04**
±STD	±2.07	±2.45	±1.82	±1.47	±1.61	±1.41	±1.61	±1.50
t-test	1		1		1		1	

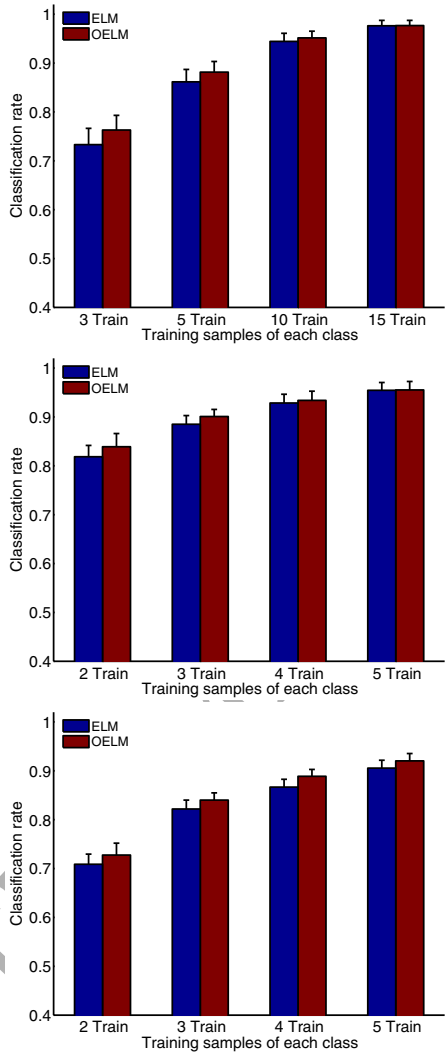


Figure 1: Average performance of ELM and OELM across 20 experiments.

that "OELM achieves higher classification accuracy" is correct in most cases of the three data sets.

A more intuitive view on comparison of average performance of ELM and OELM is given in Figure 1.

To evaluate the convergence property of the optimization method to OELM,

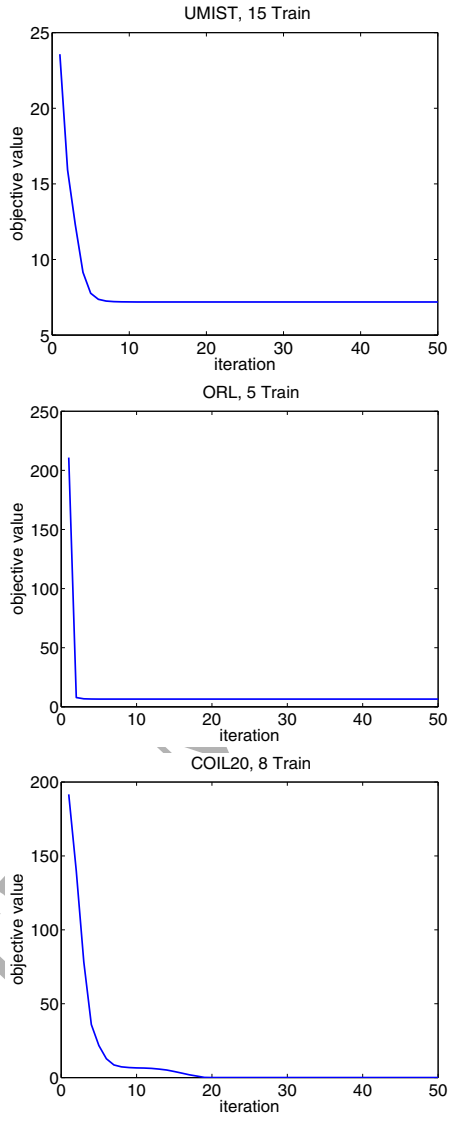


Figure 2: Convergence property of the optimization method to OELM.

we show the objective values of OELM in Figure 2 on these three data sets. We can easily see that OELM usually converge in a few iteration. Therefore, in all experiments above, we set such number as 10.

5. Conclusion

In this paper we proposed a new ELM model, termed OELM, in which the output weight matrix is enforced to be orthogonal. The main contributions of this work lie in three aspects: 1) formulating the objective of OELM and analyzing its effectiveness from the perspective of discriminative analysis; 2) presenting an effective iterative procedure to optimize the OELM objective by solving a balanced orthogonal procrustes problem via singular value decomposition; 3) demonstrating the effectiveness of OELM by conducting extensive experiments on three image data sets. As our future work, we will investigate OELM as a feature extraction model other than a classifier.

Appendix

The updating rule for $\widetilde{\mathbf{W}}$ in (13) can be reached based on the following derivation.

$$\begin{aligned} &\|\mathbf{H}\widetilde{\mathbf{W}} - [\mathbf{Y}, \mathbf{Y}_1]\|^2 \\ =&\mathrm{Tr}((\mathbf{H}\widetilde{\mathbf{W}} - [\mathbf{Y}, \mathbf{Y}_1])^T(\mathbf{H}\widetilde{\mathbf{W}} - [\mathbf{Y}, \mathbf{Y}_1])) \\ =&\mathrm{Tr}(\widetilde{\mathbf{W}}^T\mathbf{H}^T\mathbf{H}\widetilde{\mathbf{W}} - 2\widetilde{\mathbf{W}}^T\mathbf{H}^T[\mathbf{Y}, \mathbf{Y}_1] + [\mathbf{Y}, \mathbf{Y}_1]^T[\mathbf{Y}, \mathbf{Y}_1]) \end{aligned} \tag{24}$$

Thus, minimizing $\|\mathbf{H}\widetilde{\mathbf{W}} - [\mathbf{Y}, \mathbf{Y}_1]\|^2$ equals to maximizing

$$\begin{aligned} &\mathrm{Tr}(\widetilde{\mathbf{W}}^T\mathbf{H}^T[\mathbf{Y}, \mathbf{Y}_1]) \\ =&\mathrm{Tr}(\widetilde{\mathbf{W}}^T\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T) \quad ([\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}] = \mathrm{SVD}(\mathbf{H}^T[\mathbf{Y}, \mathbf{Y}_1])) \\ =&\mathrm{Tr}(\mathbf{\Sigma}\mathbf{V}^T\widetilde{\mathbf{W}}^T\mathbf{U}) \quad \mathbf{Z} = \mathbf{V}^T\widetilde{\mathbf{W}}^T\mathbf{U} \\ =&\mathrm{Tr}(\mathbf{\Sigma}\mathbf{Z}) = \sum_i z_{ii}\sigma_{ii} \leq \sum_i \sigma_{ii} \end{aligned} \tag{25}$$

The last inequality holds because $\mathbf{Z}$ is also an orthogonal matrix, and $\sum_j z_{ij}^2 = 1$, $z_{ij} \leq 1$. The objective can achieve the maximum if $\mathbf{Z} = \mathbf{I}$, i.e., $\widetilde{\mathbf{W}} = \mathbf{U}\mathbf{V}^T$.

Acknowledgments

This work was partially supported by Natural Science Foundation of China (61602140,61671193,61402143), Science and Technology Program of Zhejiang

Province (2017C33049), Natural Science Foundation of Zhejiang Province (LQ14F020012), Jiangsu Key Laboratory of Image and Video Understanding for Social Safety, Nanjing University of Science and Technology (30916014107) and Guangxi High School Key Laboratory of Complex System and Computational Intelligence.

References

[1] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: a new learning scheme of feedforward neural networks, in: IEEE International Joint Conference on Neural Networks, Vol. 2, 2004, pp. 985–990.

[2] 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.

[3] 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.

[4] X. Liu, S. Lin, J. Fang, Z. Xu, Is extreme learning machine feasible? A theoretical assessment (Part I), IEEE Transactions on Neural Networks and Learning Systems 26 (1) (2015) 7–20.

[5] S. Lin, X. Liu, J. Fang, Z. Xu, Is extreme learning machine feasible? A theoretical assessment (Part II), IEEE Transactions on Neural Networks and Learning Systems 26 (1) (2015) 21–34.

[6] G.-B. Huang, L. Chen, Convex incremental extreme learning machine, Neurocomputing 70 (16) (2007) 3056–3062.

[7] G.-B. Huang, L. Chen, Enhanced random search based incremental extreme learning machine, Neurocomputing 71 (16) (2008) 3460–3468.

[8] 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.

[9] G.-B. Huang, An insight into extreme learning machines: random neurons, random features and kernels, Cognitive Computation 6 (3) (2014) 376–390.

[10] L. L. C. Kasun, H. Zhou, G.-b. Huang, C. M. Vong, Representational learning with elms for big data, IEEE Intelligent Systems 28 (6) (2013) 31–34.

[11] J. Tang, C. Deng, G.-B. Huang, Extreme learning machine for multilayer perceptron, IEEE transactions on neural networks and learning systems 27 (4) (2016) 809–821.

[12] 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.

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

[14] 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.

[15] G. Huang, G.-B. Huang, S. Song, K. You, Trends in extreme learning machines: a review, Neural Networks 61 (2015) 32–48.

[16] S. Ding, H. Zhao, Y. Zhang, X. Xu, R. Nie, Extreme learning machine: algorithm, theory and applications, Artificial Intelligence Review 44 (1) (2015) 103–115.

[17] S. Xiang, F. Nie, G. Meng, C. Pan, C. Zhang, Discriminative least squares regression for multiclass classification and feature selection, IEEE Transactions on Neural Networks and Learning Systems 23 (11) (2012) 1738–1754.

[18] X.-Y. Zhang, L. Wang, S. Xiang, C.-L. Liu, Retargeted least squares regression algorithm, IEEE Transactions on Neural Networks and Learning Systems 26 (9) (2015) 2206–2213.

[19] 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.

[20] Y. Peng, B.-L. Lu, Discriminative manifold extreme learning machine and applications to image and EEG signal classification, Neurocomputing 174 (2016) 265–277.

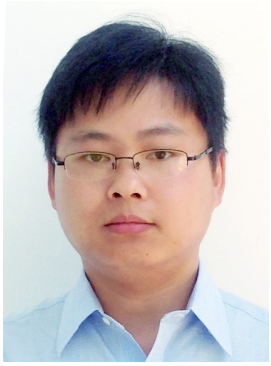
[21] L.-C. Shi, B.-L. Lu, EEG-based vigilance estimation using extreme learning machines, Neurocomputing 102 (2013) 135–143.

[22] D. Cai, X. He, J. Han, H.-J. Zhang, Orthogonal laplacianfaces for face recognition, IEEE Transactions on Image Processing 15 (11) (2006) 3608–3614.

[23] X. He, S. Yan, Y. Hu, P. Niyogi, H.-J. Zhang, Face recognition using laplacianfaces, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (3) (2005) 328–340.

[24] F. Nie, S. Xiang, Y. Liu, C. Hou, C. Zhang, Orthogonal vs. uncorrelated least squares discriminant analysis for feature extraction, Pattern Recognition Letters 33 (5) (2012) 485–491.

[25] H. Zhao, Z. Wang, F. Nie, Orthogonal least squares regression for feature extraction, Neurocomputing 216 (2016) 200–207.



Yong Peng Received the B.S. degree from Hefei New Star Research Institute of Applied Technology, the M.S. degree from Graduate University of Chinese Academy of Sciences, and the PhD degree from Shanghai Jiao Tong University, all in computer science, in 2006, 2010 and 2015, respectively. From September 2012 to August 2014, he was a visiting PhD student in the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor. He joined in School of Computer Science and Technology, Hangzhou Dianzi University as an Assistant Professor in June 2015 where he is currently a Research Associate Professor. He was awarded by the President Scholarship, Chinese Academy of Sciences in 2009 and National Scholarship for Graduate Students, Ministry of Education in 2012. His research interests are machine learning, pattern recognition and brain-computer interface.

Wanzeng Kong Received both bachelor degree and Ph.D degree from Electrical Engineering Department, Zhejiang University, Hangzhou, China, in 2003 and 2008 respectively. He is currently a professor and vice dean of college of computer science, Hangzhou Dianzi University, Hangzhou, China. From Nov.

2012 to Nov. 2013, Dr. Kong is a visiting research associate in department of biomedical engineering, University of Minnesota, Twin Cities, USA. His research interests include cognitive computing, pattern recognition and BCI-based electronic system. Dr. Kong now is also a member of IEEE, ACM and CCF.

Bing Yang Received her Ph.D. degree in Computer Science from Zhejiang University in 2013 and then joined in School of Computer Science and Technology, Hangzhou Dianzi University where she is now serving as an associate professor. Her main research interests computer vision and machine learning.