Hindawi
Scientific Programming
Volume 2018, Article ID 4563040, 10 pages
https://doi.org/10.1155/2018/4563040

Research Article

Robust Matching Pursuit Extreme Learning Machines

Zejian Yuan,[1] Xin Wang,[1] Jiuwen Cao,[2] Haiquan Zhao,[3] and Badong Chen[1]

[1]*Institute of Artificial Intelligence and Robotics, Xian Jiaotong University, Xi'an 710049, China*
[2]*Institute of Information and Control, Hangzhou Dianzi University, Zhejiang 310018, China*
[3]*School of Electrical Engineering, Southwest Jiaotong University, Chengdu, China*

Correspondence should be addressed to Badong Chen; chenbd@mail.xjtu.edu.cn

Received 25 August 2017; Revised 23 November 2017; Accepted 7 December 2017; Published 1 February 2018

Academic Editor: Wenbing Zhao

Extreme learning machine (ELM) is a popular learning algorithm for single hidden layer feedforward networks (SLFNs). It was originally proposed with the inspiration from biological learning and has attracted massive attentions due to its adaptability to various tasks with a fast learning ability and efficient computation cost. As an effective sparse representation method, orthogonal matching pursuit (OMP) method can be embedded into ELM to overcome the singularity problem and improve the stability. Usually OMP recovers a sparse vector by minimizing a least squares (LS) loss, which is efficient for Gaussian distributed data, but may suffer performance deterioration in presence of non-Gaussian data. To address this problem, a robust matching pursuit method based on a novel kernel risk-sensitive loss (in short KRSLMP) is first proposed in this paper. The KRSLMP is then applied to ELM to solve the sparse output weight vector, and the new method named the KRSLMP-ELM is developed for SLFN learning. Experimental results on synthetic and real-world data sets confirm the effectiveness and superiority of the proposed method.

1. Introduction

Extreme learning machine [1] is a kind of single hidden layer feedforward network (SLFN) [2]. In the past decade, ELM became popular and attractive in the machine learning and pattern recognition communities for its fast adaptability and good generalization performance [3]. In general, ELM has the following advantages: (i) It not only has the ability of estimating the unknown mathematical model embedded in a mass of training samples but also possesses parallel schemes to be efficiently implemented in parallel for training and testing; (ii) it uses randomly generated input weights and hidden biases without tuning during the training phase, and therefore, the output weights can be analytically obtained by solving the standard least squares (LS) problem. Thus, extremely fast learning ability and efficient computation cost can be achieved, especially for big data applications. In view of these remarkable superiorities, ELM has been widely applied in many applications, such as face recognition [4], series compensated transmission line protection [5], time series analysis [6], and nonlinear model identification [7].

However, ELM still has several drawbacks. First, ELM encounters the problem of irrelevant variables when handling real-world data sets [8]. Second, choosing a proper hidden nodes number is an open problem for all ELM algorithms. An ELM network with too few hidden nodes may not be accurate for modeling the input data, whereas a network with too many hidden nodes tends to generate an overfitting model [9]. Moreover, when the number of hidden nodes is more than the input data, ELM might have the singularity problem [4]. Third, the original ELM learns the model with an L_2-norm based loss function, which is very vulnerable to noise. It is well known that the L_2-norm can magnify the bad effects of outliers associated with large deviations [10]. The presence of non-Gaussian noises or outliers in the training data may thus lead to an unreliable model with degraded performance.

To overcome the first and second limitations, several methods have been proposed in the regularization framework [9, 11–13]. Furthermore orthogonal matching pursuit (OMP) is a plain and efficient iterative algorithm which chooses an atom in the dictionary with the best correlation to the remaining elements at each iteration [14]. As such, OMP has been

embedded to ELM (OMP-ELM) to overcome the singularity problem and led to more stable solution than the original ELM [15]. Most of the existing methods learn the model with an L_2-norm based loss function, which may perform poorly in the presence of non-Gaussian noises (which exist in many real-world situations) or outliers [16–18]. To combat non-Gaussian noises or outliers and improve the generalization ability, the regularized correntropy criterion is used to replace the L_2-norm based loss function in original ELM model to develop the ELM-RCC [16]. In [19], ELM with L_1-norm based loss function (ORELM) was proposed to achieve robust performance.

The kernel risk-sensitive loss (KRSL) is a nonlinear similarity measure firstly proposed in [20], which can reach a more satisfying robust performance. The KRSL is based on the original structure of risk-sensitive loss and is defined in the reproducing kernel Hilbert space (RKHS) [21, 22]:

$$V(X,Y) = \frac{1}{\lambda}\mathbf{E}\left[\exp\left(\lambda\left(1-\kappa_\sigma(X-Y)\right)\right)\right], \quad (1)$$

where $\mathbf{E}[\cdot]$ denotes the mathematical expectation, $\kappa_\sigma(\cdot)$ is the Gaussian kernel with bandwidth σ, and λ is the risk-sensitive parameter. In this paper, we propose a KRSL based matching pursuit (KRSLMP) method. The KRSLMP is then embedded to ELM to construct a robust and sparse ELM model.

The rest of the paper is structured as follows. In Section 2, we sketch the related work, including similarity measures in kernel space, kernel risk-sensitive loss, ELM model, and orthogonal matching pursuit algorithm. In Section 3, we develop the KRSLMP-ELM. In Section 4, experiments on regression problem with synthetic and real-world data sets are conducted to verify the effectiveness of the proposed algorithm. The sensitivity of the KRSLMP-ELM to free parameters is also analyzed. Finally, conclusion is given in Section 5.

2. Preliminaries and Related Works

For convenience of presentation, the following notations used in this paper are introduced. Vectors and matrices are represented with boldface lowercase letters and boldface capital letters, respectively. For any vector $\mathbf{x}$, we use $x(i)$ to denote its ith entry. The notation $\mathbf{x}|_I$ denotes the subvector of $\mathbf{x} \in \mathbb{R}^n$ with entries indexed by the set $I \subset \Omega = \{1, 2, \ldots, n\}$. The complementary set of I is denoted as $I^c = \Omega - I$.

2.1. Similarity Measures in Kernel Space. Let X and Y be two random variables; the correntropy between X and Y is defined by [17, 23]

$$V(X,Y) = \mathbf{E}\left[\kappa_\sigma(X-Y)\right] = \int \kappa_\sigma(x-y)\, dF_{XY}(x,y), \quad (2)$$

where $F_{XY}(x, y)$ is the joint distribution function of (X, Y). The Gaussian kernel with bandwidth σ is given by

$$\kappa_\sigma(x-y) = \exp\left(-\frac{(x-y)^2}{2\sigma^2}\right). \quad (3)$$

Correntropy $V(X, Y)$ is a local correlation measure in the kernel space $\mathbb{H}$. According to Mercers theorem [24], it can be expressed in terms of the inner product as

$$V(X,Y) = \mathbf{E}\left[\langle \Phi(X), \Phi(Y)\rangle_{\mathbb{H}}\right]. \quad (4)$$

It applies a kernel trick that nonlinearly maps the original space to a higher dimensional feature space. It can be shown that correntropy is directly related to the probability of how similar two random variables are in a neighborhood of the joint space controlled by the kernel bandwidth σ [17, 25, 26].

2.2. Kernel Risk-Sensitive Loss. Similarity measures in kernel space have the ability to extract higher-order statistics of data, which can significantly improve the learning performance in non-Gaussian environments [21]. The optimization problem can be determined by maximizing the correntropy criterion (MCC) or equivalently minimizing the correntropic loss (C-Loss) [27, 28] between the output estimation and the target response. However, highly nonconvex problem may happen in C-Loss performance surface which has steep slopes around the optimal solution but is extremely flat far from the solution. This may lead to slow convergence and poor performance. Choosing a large kernel bandwidth may overcome the above problem. But the robustness will decrease significantly when outliers occur with kernel bandwidth increasing [29]. To achieve a satisfying performance surface, the KRSL was proposed in [20].

The KRSL is defined by

$$\begin{aligned} L_\lambda(X,Y) &= \frac{1}{\lambda}\mathbf{E}\left[\exp\left(\lambda\left(1-\kappa_\sigma(X-Y)\right)\right)\right] \\ &= \frac{1}{\lambda}\int \exp\left(\lambda\left(1-\kappa_\sigma(x-y)\right)\right) dF_{XY}(x,y) \end{aligned} \quad (5)$$

which can also be expressed in a traditional risk-sensitive loss form as [30]

$$L_\lambda(X,Y) = \frac{1}{\lambda}\mathbf{E}\left[\exp\left(\lambda\left(\frac{1}{2}\left\|\Phi(X)-\Phi(Y)\right\|_{\mathbb{H}}^2\right)\right)\right], \quad (6)$$

where λ is the risk-sensitive parameter that controls the shape of performance surface.

In practice, the joint distribution function of X and Y is usually unknown and only a finite number of samples $\{(x_j, y_j)\}_{j=1}^M$ are available. The KRSL can thus be estimated by

$$\widehat{L}_\lambda(X,Y) = \frac{1}{M\lambda}\sum_{j=1}^{M}\exp\left(\lambda\left(1-\kappa_\sigma\left(x_j-y_j\right)\right)\right). \quad (7)$$

As one can see, (6) defines a distance between the vectors $\mathbf{X} = [x_1, x_2, \ldots, x_M]^T$ and $\mathbf{Y} = [y_1, y_2, \ldots, y_M]^T$.

2.3. Extreme Learning Machine. Extreme learning machine (ELM) was proposed by Huang et al. for training single hidden layer feedforward neural networks (SLFNs) [2, 31]. The input weights and biases are initialized randomly in ELM and remain unchanged during training. The network learning thus becomes optimizing the output weights, which can be

formulated as solving a linear equation. Let $\{(\mathbf{x}_j, y_j)\}_{j=1}^{M}$ be given by M training samples, where input $\mathbf{x}_j \in \mathbb{R}^n$ and corresponding desired output $y_j \in \mathbb{R}$; the relationship between $\mathbf{x}_j$ and y_j can be represented under the assumption of the model. The network model of ELM with L hidden neurons can be modeled and expressed as

$$\sum_{i=1}^{L} \beta_i f\left(\mathbf{a}_i \cdot \mathbf{x}_j + b_i\right) = y_j, \quad j = 1, 2, \ldots, M, \tag{8}$$

where L is hidden nodes number, β_i is the weight connecting the ith hidden node and output nodes, f is the activation function (in this work, f is a sigmoid function without explicit mention), $\mathbf{a}_i$ denotes the weight that connects the ith hidden node and input nodes, and b_i represents the randomly chosen bias of the ith hidden node. Equation (7) can be compactly written as a matrix notation

$$\mathbf{y} = \mathbf{H}\boldsymbol{\beta}, \tag{9}$$

where

$$\mathbf{H} = \begin{pmatrix} f\left(\mathbf{a}_1 \cdot \mathbf{x}_1 + b_1\right) & \cdots & f\left(\mathbf{a}_L \cdot \mathbf{x}_1 + b_L\right) \\ \vdots & \ddots & \vdots \\ f\left(\mathbf{a}_1 \cdot \mathbf{x}_M + b_1\right) & \cdots & f\left(\mathbf{a}_L \cdot \mathbf{x}_M + b_L\right) \end{pmatrix} \tag{10}$$

and $\boldsymbol{\beta}$ is the minimal norm least squares solution of (8). The parameter $\boldsymbol{\beta}$ can be obtained by

$$\boldsymbol{\beta} = \mathbf{H}^{\dagger}\mathbf{y}, \tag{11}$$

where $\mathbf{H}^{\dagger}$ is the Moore Penrose generalized inverse of the hidden layer output matrix $\mathbf{H}$.

2.4. Orthogonal Matching Pursuit. Matching pursuit method is one of the effective methods for sparse representation [14, 32, 33]. In general, a sparse representation problem can be formulated as

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \boldsymbol{\rho}, \tag{12}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$ $(m < n)$ denotes the measurement matrix, $\mathbf{x}$ is the sparse vector, and $\boldsymbol{\rho} \in \mathbb{R}^m$ represents the noise vector. The main purpose is to recover the sparse vector $\mathbf{x}$ from the observation $\mathbf{y}$ and the measurement matrix $\mathbf{A}$. The OMP uses the L_0-norm constrained least squares model

$$\begin{aligned} \min_{\mathbf{x}} \quad & \left\|\mathbf{y} - \mathbf{A}\mathbf{x}\right\|_2^2 \\ \text{s.t.} \quad & \|\mathbf{x}\|_0 \le K, \end{aligned} \tag{13}$$

where $\|\mathbf{x}\|_0$ counts the number of nonzero coordinates of $\mathbf{x}$.

In the following, we briefly describe the OMP method. First, we initialize the residual $\mathbf{r}_0 = \mathbf{y}$, the index set $\Lambda_0 = \emptyset$, and the iteration $t = 1$. At each iteration, OMP algorithm selects a column of the measurement matrix $\mathbf{A}$ which is most correlated to the residual as

$$\alpha_t = \arg\max_{i=1,\ldots,n} \left|\langle \mathbf{r}_{t-1}, \varphi_i \rangle\right|, \tag{14}$$

where $\mathbf{r}_{t-1}$ denotes the residual in $t-1$th iteration and φ_i is the ith column of $\mathbf{A}$. Then collect α_t to index set Λ

$$\Lambda_t = \Lambda_{t-1} \cup \{\alpha_t\}. \tag{15}$$

We can solve an LS problem to obtain a new estimation $\mathbf{x}_t$ supported in Λ_t:

$$\mathbf{x}_t = \arg\min_{\mathbf{x} \in \mathbb{R}^n, \operatorname{supp}(\mathbf{x}) \subset \Lambda_t} \left\|\mathbf{y} - \mathbf{A}\mathbf{x}\right\|_2, \tag{16}$$

where $\operatorname{supp}(\mathbf{x})$ denotes the support set of $\mathbf{x}$. If the stopping criterion is satisfied, we output $\mathbf{x}_t$ as the estimate of $\mathbf{x}$.

Then one can update the residual

$$\mathbf{r}_t = \mathbf{y} - \mathbf{A}\mathbf{x}_t. \tag{17}$$

From (8) and (11), we can find that ELM has a similar network model for sparse representation problem. Thus, one can take advantage of the OMP algorithm for selecting the best hidden nodes of the ELM network. The OMP estimates the sparse vector by using the L_2-norm based criterion, which performs well with the Gaussian error distribution. However, the presence of non-Gaussian noise may give rise to performance degradation.

3. Kernel Risk-Sensitive Loss Based Matching Pursuit Extreme Learning Machine

To address the aforementioned issue, we propose a robust kernel risk-sensitive loss based orthogonal matching pursuit extreme learning machine algorithm (KRSLMP-ELM) in this section. In the KRSLMP-ELM, we initialize the residual $\mathbf{r}_0$ as $\mathbf{y}$ and the initial index set as $\Lambda_0 = \emptyset$. Then, similar to OMP, a column of H most correlated with the residual is selected and the index set is augmented at each iteration. Then we obtain a new estimation $\boldsymbol{\beta}_t$ by solving the following KRSL minimization problem:

$$\boldsymbol{\beta}_t = \arg\min_{\boldsymbol{\beta} \in \mathbb{R}^L, \operatorname{supp}(\boldsymbol{\beta}) \subset \Lambda_t} \phi_{\sigma,\lambda}\left(\mathbf{y} - \mathbf{H}\boldsymbol{\beta}\right) + C\left\|\boldsymbol{\beta}\right\|_2^2. \tag{18}$$

We utilize the half-quadratic (HQ) theory [34] to construct the optimization algorithm. Considering that the measurements may include both large and small noise, we can use HQ optimization to estimate the importance of different samples. The samples severely corrupted will be assigned small weight values in learning procedure to decrease the impact of large noise. Thus, the performance of KRSLMP-ELM can be significantly further improved.

According to the convex optimization theory [35], the dual function for $\phi_{\sigma,\lambda}(x) = (1/\lambda)\exp(\lambda(1 - \exp(-x^2/2\sigma^2)))(0 < \lambda < 1)$ is convex and defined as

$$\psi(s) = \sup_{t \in \mathbb{R}} \left\{-sx^2 + \phi_{\sigma,\lambda}(x)\right\} \tag{19}$$

and then

$$\phi_{\sigma,\lambda}(x) = \inf_{s \in \mathbb{R}} \left\{sx^2 + \psi(s)\right\}, \tag{20}$$

where the infimum is reached at $s = \phi_{\sigma,\lambda}(x)$. We point out here that when the parameter $\lambda > 1$, the KRSLMP-ELM can also work well in our simulations. Substituting (18) for (20), the KRSLMP-ELM objective function can be reformulated as

$$\min_{\substack{\boldsymbol{\beta}\in\mathbb{R}^L,\mathbf{w}\in\mathbb{R}_+^M \\ \mathrm{supp}(\boldsymbol{\beta})\subset\Lambda_k}} J(\boldsymbol{\beta},\mathbf{w}) = \left\| \sqrt{\mathrm{diag}(\mathbf{w})}\,(\mathbf{y}-\mathbf{H}\boldsymbol{\beta}) \right\|_2^2 + C\|\boldsymbol{\beta}\|_2^2 + \sum_{i=1}^{M}\psi(w(i)), \tag{21}$$

where diag($\mathbf{w}$) represents a diagonal matrix with its primary diagonal element $w(i)$ and C is the regularization parameter. Inspired by the HQ theory, (21) can be solved by the following alternate technique:

$$w^{(t+1)}(i) = \frac{1}{\lambda}\exp\left(\lambda\left(1-\kappa_\sigma\left(y(i)-\left(\mathbf{H}\boldsymbol{\beta}^{(t)}\right)(i)\right)\right)\right),$$

$$\boldsymbol{\beta}^{(t+1)} = \underset{\boldsymbol{\beta}\in\mathbb{R}^L,\mathrm{supp}(\boldsymbol{\beta})\subset\Lambda_k}{\arg\min} \left\| \sqrt{\mathrm{diag}\left(\mathbf{w}^{(t+1)}\right)}\,(\mathbf{y}-\mathbf{H}\boldsymbol{\beta}) \right\|_2^2 + C\|\boldsymbol{\beta}\|_2^2, \tag{22}$$

where t denotes the iteration number. In the proposed algorithm, the bandwidth is adaptively chosen during the iteration. In order to make the scheme robust to outliers, we calculate the value of σ as follows.

Denote the training error as $e(i) = \|y(i) - (\mathbf{H}\boldsymbol{\beta})(i)\|_2^2$, $i = 1, 2, \ldots, M$. We can then reorder the error in an ascending order, and we get the reordered as e_σ. Let $k = \lfloor \tau M \rfloor$, where scalar $\tau \in (0, 1]$ and $\lfloor \tau M \rfloor$ outputs the largest integer smaller than τM. We can select $e_\sigma(k)$ as the bandwidth in accordance with the proportion of outlier. Discussions on the detailed experimental results by choosing different bandwidths are given in the experiment section. A solution for the optimization problem in (21) can be derived as follows:

$$\boldsymbol{\beta}^{(t+1)}\big|_{\Lambda_t} = \left(\mathbf{H}^T\mathrm{diag}\left(\mathbf{w}^{(t+1)}\right)\mathbf{H} + \frac{1}{C}\mathbf{I}\right)^{-1}\mathbf{H}^T\mathrm{diag}\left(\mathbf{w}^{(t+1)}\right)\mathbf{y}, \tag{23}$$

where $\boldsymbol{\beta}^{(t+1)}|_{\Lambda_t^c} = 0$ and $\mathbf{I}$ denotes the identity matrix.

Since the importance degree of the measurements is employed to adaptively update the output weight vector in the KRSLMP-ELM, we update the residual

$$\mathbf{r}_t = \sqrt{\mathrm{diag}\left(\mathbf{w}^{(t)}\right)}\left(\mathbf{y}-\mathbf{H}\boldsymbol{\beta}^{(t)}\right). \tag{24}$$

It is noted that the sparsity level K has to be assigned in advance in the KRSLMP-ELM. The sparsity K directly determines the number of the active hidden nodes used in ELM due to the fact that more hidden nodes than necessary are generated. To obtain the best sparsity level K, namely, the best number of hidden nodes used in ELM, we utilize the root mean square error (RMSE) as the criterion

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{M}(y_i-\widehat{y}_i)^2}{M}}, \tag{25}$$

where y_i denotes the target response and $\widehat{y}_i$ the corresponding output estimated by the KRSLMP-ELM.

For different sparsity level K, the corresponding RMSE is first calculated. Then the best $\boldsymbol{\beta}$ coefficients associated with the minimum RMSE value are selected.

The iteration is repeated until achieving the stopping criterion. The KRSLMP-ELM is summarized in Algorithm 1.

4. Experimental Results

To validate the effectiveness of the proposed KRSLMP-ELM algorithm, experiments on two synthetic data sets and seven benchmark data sets are conducted in this section. The performance of the new method is compared to five state-of-the-art algorithms, namely, ELM, RELM, ELM-RCC, OMP-ELM, and ORELM. Sigmoid function $f(x) = 1/(1 + e^{-x})$ is used as the activation function for all methods.

4.1. Synthetic Data Sets. In this subsection, experiments on two synthetic regression data sets for nonlinear function approximation problem are carried out. Descriptions of the two data sets are as follows.

Sinc. The synthetic data set is generated by $y_i = c \cdot \mathrm{Sinc}(x_i) + \rho_i$, where $c = 8$ and

$$\mathrm{Sinc}(x) = \begin{cases} \dfrac{\sin(x)}{x} & x \neq 0 \\ 1 & x = 0 \end{cases} \tag{26}$$

and ρ_i contains two mutually independent noises that are inner noise B_i and outliers noise O_i. Specifically, ρ_i is defined as $\rho_i = (1 - g_i)B_i + g_iO_i$, where g_i is binary distributed with the probability masses $\Pr\{g_i = 1\} = p$ and $\Pr\{g_i = 0\} = 1 - p (0 \leq p \leq 1)$. B_i and O_i are independent of g_i. In this experiment, p is set at 0.1. The outlier O_i is generated by using a zero-mean Gaussian distributed noise with standard deviation 4.0. For the inner noise B_i, two different noises are tested, which are (a) uniform distribution over $[-1.0, 1.0]$ and (b) Sine wave noise $\sin(\alpha)$, with α uniformly distributed over $[0, 2\pi]$. We uniformly generate the input data x_i from $[-10.0, 10.0]$, where 200 data points are used for training and another 200 clean data points which are not contaminated by any noise are used for testing.

Func. This synthetic data set is generated by

$$\mathbf{y}(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1\exp\left\{-\left(\mathbf{x}_1^2 + \mathbf{x}_2^2\right)\right\} + \boldsymbol{\rho}, \tag{27}$$

where $\boldsymbol{\rho}$ is a zero-mean Gaussian distributed noise vector with standard deviation 0.4. The input data vectors $\mathbf{x}_1$ and $\mathbf{x}_2$ are uniformly generated from $[-2.0, 2.0]$. Similar to the previous experiments, 200 data samples are used for training and another 200 data samples without noise are used for testing.

Input: samples $\{\mathbf{x}_i, y_i\}_{i=1}^{M}$
Output: weight vector $\boldsymbol{\beta}$

Parameters setting: number of hidden nodes $\widetilde{L}$, regularization parameter C and sparsity level K.

Initialization: randomly initialize ELM parameters: input weights $\mathbf{a}_i$ and biases b_i $(i = 1, \ldots, L)$ in measurement matrix $\mathbf{H}$.

Set the index set $\Lambda_0 = \emptyset$, the residual $\mathbf{r}_0 = \mathbf{y}$, the iteration counter $t = 0$ and $\mathrm{diag}(\mathbf{w}^0) = \mathbf{I}$.

(1) **for** $t = 1, 2, \ldots, K$ **do**

(2) $t = t + 1$

(3) Find a column of $\mathbf{H}$ most correlated with the residual

$$\alpha_t = \arg\max_{j=1,2,\ldots,L} \left| \left\langle \mathbf{r}_{t-1}, \sqrt{\mathrm{diag}\left(\mathbf{w}^{(t-1)}\right)} \cdot h_j \right\rangle \right|$$

(4) Augment the index set

$$\Lambda_t = \Lambda_{t-1} \cup \{\alpha_t\}$$

(5) Solve the KRSLMP minimization problem by the following iterations

$$w_i^{(t+1)} = \frac{1}{\lambda} \exp\left(\lambda\left(1 - \kappa_\sigma\left(y_i - \left(\mathbf{H}\boldsymbol{\beta}^{(t)}\right)_i\right)\right)\right)$$

$$\boldsymbol{\beta}^{(t+1)} = \arg\min_{\boldsymbol{\beta} \in \mathbb{R}^L, \mathrm{supp}(\boldsymbol{\beta}) \subset \Lambda_k} \left\| \sqrt{\mathrm{diag}(\mathbf{w}^{(t+1)})}(\mathbf{y} - \mathbf{H}\boldsymbol{\beta}) \right\|_2^2 + C \left\| \boldsymbol{\beta} \right\|_2^2$$

The solution is denoted as $(\mathbf{w}^{(t)}, \boldsymbol{\beta}^{(t)})$

(6) Update residual $r_t = \sqrt{\mathrm{diag}(\mathbf{w}^{(t)})}(\mathbf{y} - \mathbf{H}\boldsymbol{\beta}^{(t)})$

(7) **end for**

Algorithm 1: KRSLMP-ELM.

Table 1: Parameter settings of four algorithms in function fitting.

	ELM	RELM		OMP-ELM		ORELM		ELM-RCC			KRSLMP-ELM			
	L	L	C	L	K	L	C	L	σ	C	L	K	C	λ
Sinc-Uniform	10	80	10^{-4}	200	10	80	10^{-4}	100	1	10^{-4}	200	50	2×10^{-5}	0.01
Sinc-Sine wave	10	40	2×10^{-5}	200	10	90	10^{-4}	50	1.1	2×10^{-5}	200	50	10^{-5}	0.05
Func	35	35	10^{-6}	200	20	100	10^{-9}	70	3	10^{-4}	200	70	10^{-5}	0.001

Parameters used in the six methods for experiments of the two synthetic data sets are summarized in Table 1, where L, C, K, and λ represent the number of hidden layer nodes, regularization parameter, sparsity level, and risk-sensitive parameter in KRSLMP-ELM. We set $\tau = 0.9$ in Sinc synthetic data set experiment and $\tau = 1$ in Func synthetic data set experiment. For the convenient distinguishment of the proposed method with other methods in Sinc function approximation problem, only the estimation results of the original ELM, ORELM, ELM-RCC, and KRSLMP-ELM are illustrated in Figure 1. In Figure 2, we plot the squared training errors obtained by the KRSLMP-ELM, ELM-RCC, ORELM, and the original ELM, respectively. As shown in these figures, the KRSLMP-ELM wins the best approximation performance. The testing RMSEs of six algorithms are presented in Table 2. It is indicated that the KRSLMP-ELM is more robust than the other five methods.

Further, we perform another experiment to compare the performance of KRSLMP-ELM to that of the original ELM with different outliers. We consider the Sinc function approximation problem and set the inner noise as a zero-mean Gaussian distributed noise with standard deviation 0.1, and the outliers noise is zero-mean Gaussian with standard deviation ranging between 0.1 and 10. We run 100 trials for different outliers noises and show the RMSE results in Figure 3. One can see that the original ELM's performance degrades severely when the outliers get enhanced while the KRSLMP-ELM's performance is much less influenced by outliers.

4.2. Benchmark Data Sets. In this subsection, seven benchmark regression data sets from UCI machine learning repository [36] are tested to support the superiority of the proposed method. Specifications of the data sets are shown detailedly in Table 3. It should be pointed out that the training and testing data samples are randomly chosen in each data set and all the features are normalized into $[0, 1]$. The parameters of each method are all chosen by the fivefold cross-validation and are given in Table 4. For all algorithms, 100 independent trials are conducted and the average results are reported. The training and testing RMSEs and their standard deviation of all algorithms are listed in Table 5. As highlighted in boldface, the ELM-KRSLMP achieves the best performance in most regression data sets.

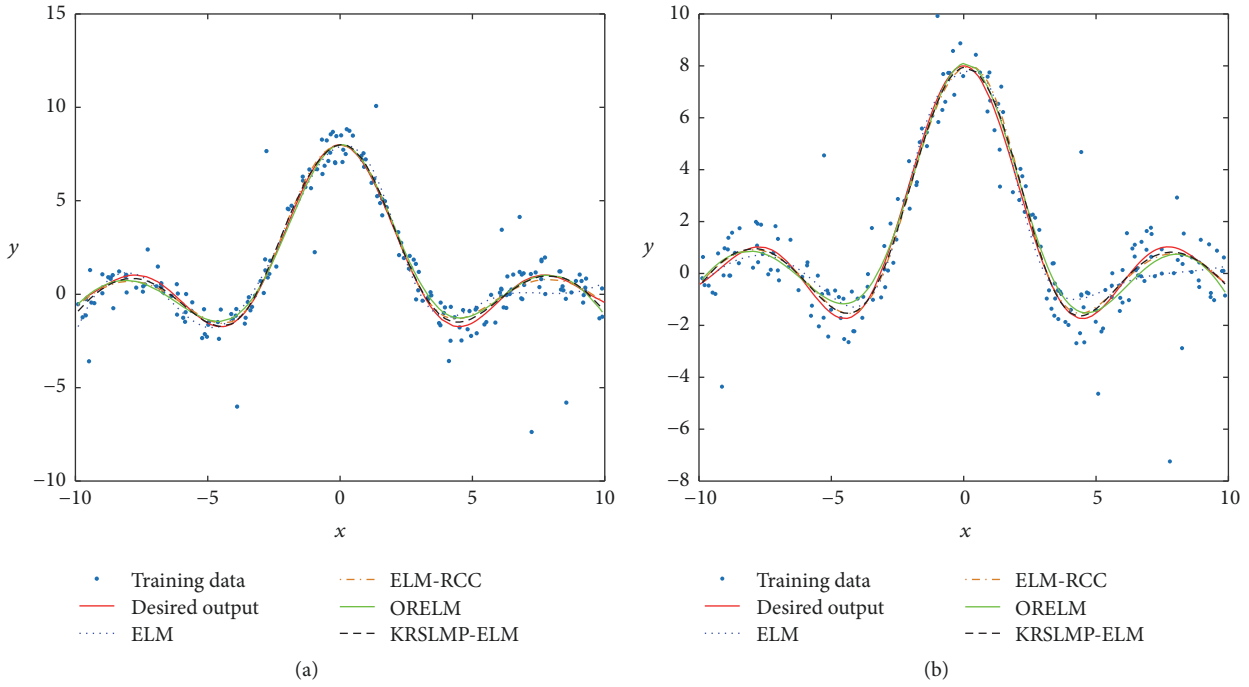

Figure 1: Sinc function regression results with different inner noises: (a) Uniform; (b) Sine wave.

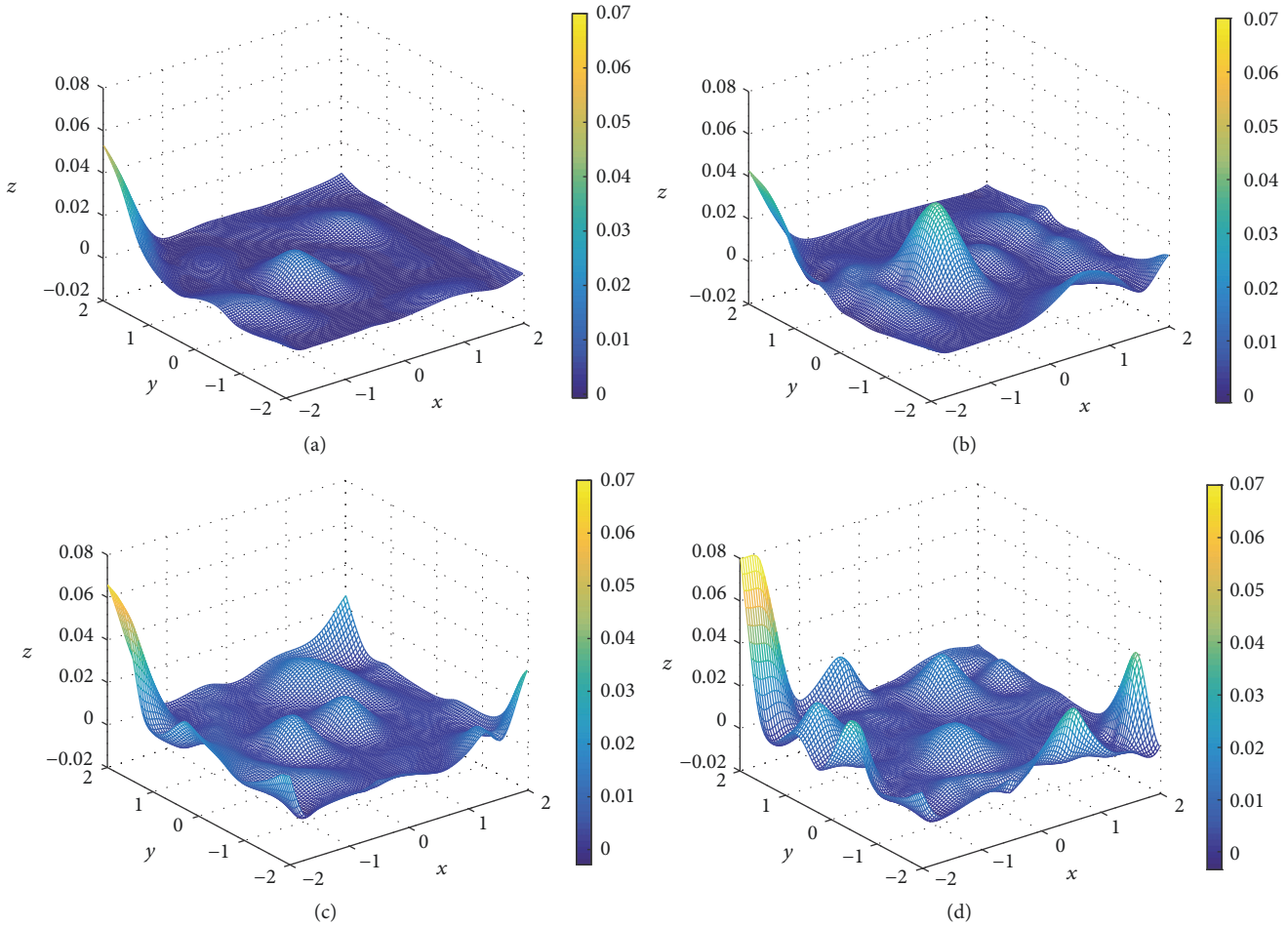

Figure 2: Squared training errors of Func function regression: (a) KRSLMP-ELM; (b) ELM-RCC; (C) ORELM; (D) original ELM.

Table 2: Testing RMSEs of six methods.

	ELM	RELM	OMP-ELM	ORELM	ELM-RCC	KRSLMP-ELM
Sinc-Uniform	0.4737	0.2871	0.3038	0.2369	0.1948	**0.1485**
Sinc-Sine wave	0.4406	0.2935	0.2711	0.2798	0.2155	**0.1562**
Func	0.0652	0.0638	0.0637	0.0591	0.0582	**0.0434**

Table 3: Specification of the data sets.

Data sets	Features	Observations	
		Training	Testing
Servo	5	83	83
Auto MPG	7	192	200
Body fat	14	126	126
Concrete	9	515	515
Housing	14	253	253
Yacht	6	154	154
Airfoil	5	751	751

Table 4: Parameter settings of six methods.

	ELM	RELM		OMP-ELM		ORELM		ELM-RCC			KRSLMP-ELM			
	L	L	C	L	K	L	C	L	σ	C	L	K	C	λ
Servo	25	90	10^{-5}	100	20	120	10^{-6}	65	0.8	10^{-4}	200	40	5×10^{-5}	1.5
Auto MPG	20	40	10^{-4}	50	15	100	10^{-4}	100	0.3	10^{-2}	200	80	10^{-3}	0.8
Body fat	20	100	10^{-2}	50	15	100	10^{-3}	160	0.1	10^{-1}	100	25	10^{-2}	1.1
Concrete	120	185	2×10^{-4}	200	80	200	10^{-7}	200	0.6	5×10^{-6}	500	140	10^{-5}	0.6
Housing	40	180	2×10^{-4}	200	30	200	10^{-5}	200	0.8	10^{-3}	500	150	10^{-3}	0.5
Yacht	90	185	2×10^{-5}	200	60	200	10^{-9}	195	0.4	10^{-7}	500	145	10^{-8}	0.3
Airfoil	130	200	2×10^{-4}	200	85	180	10^{-9}	150	0.4	10^{-7}	500	140	10^{-8}	1.0

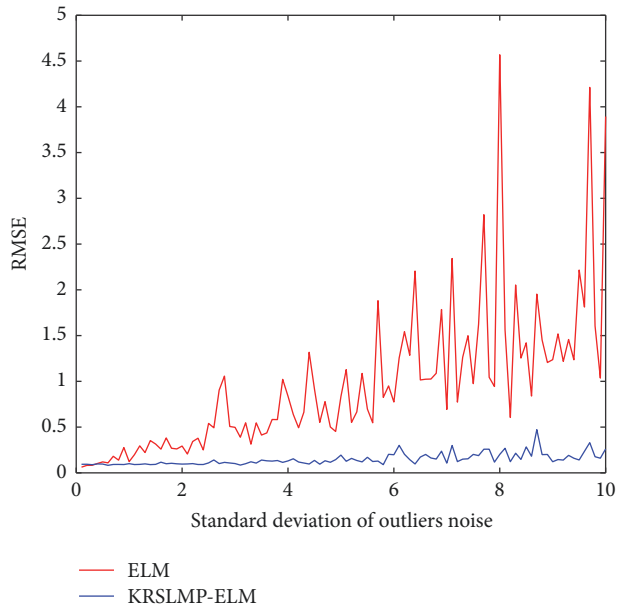

Figure 3: Sinc function regression results with different outliers noises.

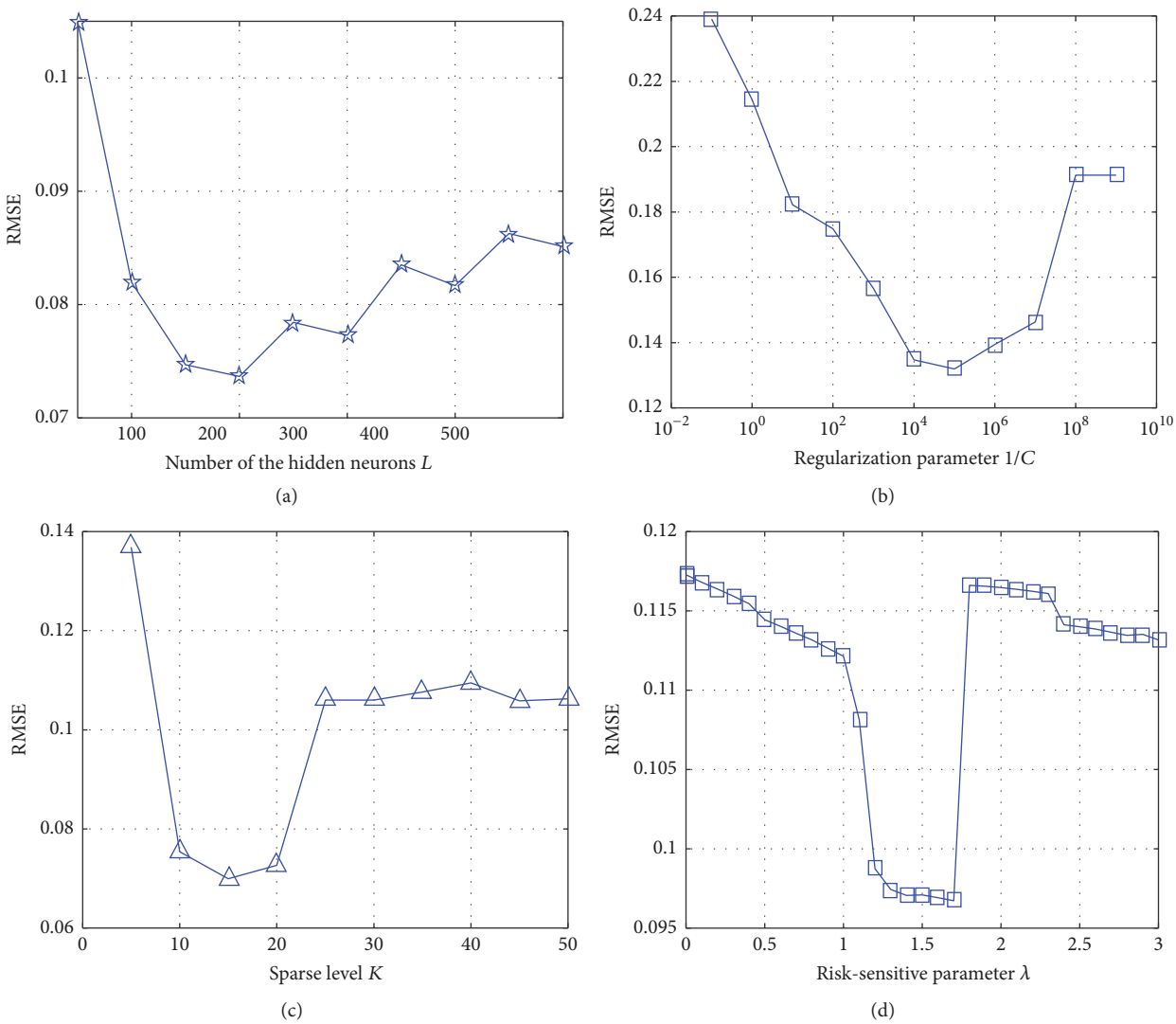

Figure 4: Regression results with different parameters: (a) L; (b) $1/C$; (c) K; (d) λ.

4.3. Sensitivity of Parameters. We analyze the sensitivity of the parameters L, K, C, and λ of KRSLMP-ELM in this subsection. For illustration, we use the regression results obtained by the Servo data set as an example. For each parameter, its sensitivity is tested by fixing the remaining parameters as the ones used in Table 4. Then, the testing RMSEs are recorded as criteria for performance comparison. The results of the regression performance are demonstrated in Figure 4.

5. Conclusion

In this paper, a robust matching pursuit based ELM algorithm, called the kernel risk-sensitive loss based matching pursuit extreme learning machine (KRSLMP-ELM), has been developed. Kernel risk-sensitive loss (KRSL) is a nonlinear similarity measure defined in kernel space, and it can achieve better performance than the conventional MSE criterion when dealing with non-Gaussian and nonlinear problems. Incorporating the KRSL into the existing orthogonal matching pursuit algorithm, we developed an improved KRSLMP-ELM algorithm, which is more robust than the OMP-ELM method. Comparisons with several existing state-of-the-art algorithms have also been provided to validate the superiority of the proposed KRSLMP-ELM algorithm.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was partially supported by the National Natural Science Foundation-Shenzhen Joint Research Program (no.

Table 5: Training and testing RMSEs for different data sets.

Data sets	ELM		RELM		OMP-ELM		ORELM		ELM-RCC		KRSLMP-ELM	
	Training RMSE	Testing RMSE	Training RMSE	Testing RMSE	Training RMSE	Testing RMSE	Training RMSE	Testing RMSE	Training RMSE	Testing RMSE	Training RMSE	Testing RMSE
Servo	0.0745 ± 0.0133	0.1189 ± 0.0228	0.0583 ± 0.0105	0.1044 ± 0.0189	0.0727 ± 0.0123	0.1134 ± 0.0181	0.0849 ± 0.0174	0.1036 ± 0.0211	0.0749 ± 0.0117	0.1034 ± 0.0171	0.0841 ± 0.0115	**0.1017 ± 0.0176**
Auto MPG	0.0689 ± 0.0049	0.0785 ± 0.0053	0.0627 ± 0.0044	0.0765 ± 0.0043	0.0676 ± 0.0044	0.0777 ± 0.0045	0.0705 ± 0.0047	0.0764 ± 0.0052	0.0683 ± 0.0042	0.0757 ± 0.0045	0.0632 ± 0.0043	**0.0749 ± 0.0045**
Body fat	0.0237 ± 0.0073	0.0340 ± 0.0060	0.0196 ± 0.0076	0.0278 ± 0.0062	0.0218 ± 0.0076	0.0313 ± 0.0059	0.0253 ± 0.0118	**0.0233 ± 0.0119**	0.0240 ± 0.0093	0.0236 ± 0.0087	0.0252 ± 0.0097	0.0239 ± 0.0095
Concrete	0.0615 ± 0.0025	0.1001 ± 0.0125	0.0732 ± 0.0022	0.0925 ± 0.0041	0.0654 ± 0.0026	0.0981 ± 0.0101	0.0668 ± 0.0026	0.0929 ± 0.0069	0.0557 ± 0.0018	0.0882 ± 0.0077	0.0542 ± 0.0018	**0.0864 ± 0.0067**
Housing	0.0736 ± 0.0053	0.0990 ± 0.0094	0.0443 ± 0.0041	0.0897 ± 0.0136	0.0644 ± 0.0048	0.0935 ± 0.0108	0.0576 ± 0.0061	0.0899 ± 0.0160	0.0453 ± 0.0039	0.0874 ± 0.0129	0.0503 ± 0.0040	**0.0849 ± 0.0114**
Yacht	0.0041 ± 0.0004	0.0583 ± 0.1374	0.0300 ± 0.0002	0.0483 ± 0.0064	0.0154 ± 0.0014	0.0437 ± 0.0134	0.0189 ± 0.0013	0.0373 ± 0.0073	0.0126 ± 0.0008	0.0320 ± 0.0063	0.0060 ± 0.0005	**0.0250 ± 0.0099**
Airfoil	0.0664 ± 0.0020	0.0967 ± 0.0130	0.0925 ± 0.0021	0.0991 ± 0.0023	0.0731 ± 0.0024	0.0968 ± 0.0105	0.0806 ± 0.0023	0.0951 ± 0.0052	0.0740 ± 0.0021	0.0907 ± 0.0061	0.0635 ± 0.0021	**0.0882 ± 0.0068**

U1613219) and National Natural Science Foundation of China (no. 91648208 and no. 61372152).

References

[1] G. Huang, "An insight into extreme learning machines: random neurons, random features and kernels," *Cognitive Computation*, 2014.

[2] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.

[3] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, vol. 2, pp. 985–990, July 2004.

[4] W. Zong and G.-B. Huang, "Face recognition based on extreme learning machine," *Neurocomputing*, vol. 74, no. 16, pp. 2541–2551, 2011.

[5] V. Malathi, N. S. Marimuthu, S. Baskar, and K. Ramar, "Application of extreme learning machine for series compensated transmission line protection," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 5, pp. 880–887, 2011.

[6] R. Singh and S. Balasundaram, "Application of extreme learning machine method for time series analysis," in *Proceedings of the Rampal Singh and S Balasundaram. Application of extreme learning machine method for time series analysis. International Journal of Intelligent Technology*, vol. 2, pp. 256–262, 2007.

[7] J. Deng, K. Li, and G. W. Irwin, "Fast automatic two-stage nonlinear model identification based on the extreme learning machine," *Neurocomputing*, vol. 74, no. 16, pp. 2422–2429, 2011.

[8] Y. Miche, M. van Heeswijk, P. Bas, O. Simula, and A. Lendasse, "TROP-ELM: a double-regularized ELM using LARS and Tikhonov regularization," *Neurocomputing*, vol. 74, no. 16, pp. 2413–2421, 2011.

[9] W. Y. Deng, Q. H. Zheng, and L. Chen, "Regularized extreme learning machine," in *Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining (CIDM '09)*, pp. 389–395, April 2009.

[10] H. Wang, "Block principal component analysis with L1-norm for image analysis," *Pattern Recognition Letters*, vol. 33, no. 5, pp. 537–542, 2012.

[11] J. M. Martínez-Martínez, P. Escandell-Montero, E. Soria-Olivas, J. D. Martín-Guerrero, R. Magdalena-Benedito, and J. Gómez-Sanchis, "Regularized extreme learning machine for regression problems," *Neurocomputing*, vol. 74, no. 17, pp. 3716–3721, 2011.

[12] L.-C. Shi and B.-L. Lu, "EEG-based vigilance estimation using extreme learning machines," *Neurocomputing*, vol. 102, pp. 135–143, 2013.

[13] Y. Miche, P. Bas, C. Jutten, O. Simula, and A. Lendasse, "A methodology for building regression models using extreme learning machine: OP-ELM," in *Proceedings of the 16th European Symposium on Artificial Neural Networks—Advances in Computational Intelligence and Learning (ESANN '08)*, pp. 247–252, April 2008.

[14] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[15] O. F. Alcin, A. Sengur, J. Qian, and M. C. Ince, "OMP-ELM: Orthogonal matching pursuit-based extreme learning machine

for regression," *Journal of Intelligent Systems*, vol. 24, no. 1, pp. 135–143, 2015.

[16] H.-J. Xing and X.-M. Wang, "Training extreme learning machine via regularized correntropy criterion," *Neural Computing and Applications*, vol. 23, no. 7-8, pp. 1977–1986, 2013.

[17] W. Liu, P. P. Pokharel, and J. C. Principe, "Correntropy: properties and applications in non-Gaussian signal processing," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5286–5298, 2007.

[18] Y. Feng, X. Huang, L. Shi, Y. Yang, and J. A. Suykens, "Learning with the maximum correntropy criterion induced losses for regression," *Journal of Machine Learning Research (JMLR)*, vol. 16, pp. 993–1034, 2015.

[19] K. Zhang and M. Luo, "Outlier-robust extreme learning machine for regression problems," *Neurocomputing*, vol. 151, no. 3, pp. 1519–1527, 2015.

[20] B. Chen, L. Xing, B. Xu, H. Zhao, N. Zheng, and J. C. Principe, "Kernel risk-sensitive loss: definition, properties and application to robust adaptive filtering," *IEEE Transactions on Signal Processing*, vol. 65, no. 11, pp. 2888–2901, 2017.

[21] J. C. Principe, *Information Theoretic Learning: Renyi's Entropy and Kernel Perspectives*, Information Science and Statistics, Springer, New York, NY, USA, 2010.

[22] X. Luo, J. Deng, J. Liu, W. Wang, X. Ban, and J. Wang, "A quantized kernel least mean square scheme with entropy-guided learning for intelligent data analysis," *China Communications*, vol. 14, no. 7, pp. 127–136, 2017.

[23] B. Chen and J. C. Príncipe, "Maximum correntropy estimation is a smoothed MAP estimation," *IEEE Signal Processing Letters*, vol. 19, no. 8, pp. 491–494, 2012.

[24] B. Schölkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT press, Cambridge, Massachusetts, USA, 2002.

[25] W. Ma, J. Duan, B. Chen, G. Gui, and W. Man, "Recursive generalized maximum correntropy criterion algorithm with sparse penalty constraints for system identification," *Asian Journal of Control*, vol. 19, no. 3, pp. 1164–1172, 2017.

[26] I. Santamaría, P. P. Pokharel, and J. C. Principe, "Generalized correlation function: definition, properties, and application to blind equalization," *IEEE Transactions on Signal Processing*, vol. 54, no. 6 I, pp. 2187–2197, 2006.

[27] B. Chen, Y. Zhu, J. Hu, and J. Principe, "System Parameter Identification: Information Criteria and Algorithms," *System Parameter Identification: Information Criteria and Algorithms*, pp. 1–249, 2013.

[28] B. Chen, J. Wang, H. Zhao, N. Zheng, and J. C. Principe, "Convergence of a fixed-point algorithm under maximum correntropy criterion," *IEEE Signal Processing Letters*, vol. 22, no. 10, pp. 1723–1727, 2015.

[29] X. Luo, J. Deng, W. Wang, J.-H. Wang, and W. Zhao, "A quantized kernel learning algorithm using a minimum kernel risk-sensitive loss criterion and bilateral gradient technique," *Entropy*, vol. 19, no. 7, article no. 365, 2017.

[30] R. K. Boel, M. R. James, and I. R. Petersen, "Robustness and risk-sensitive filtering," *Institute of Electrical and Electronics Engineers Transactions on Automatic Control*, vol. 47, no. 3, pp. 451–461, 2002.

[31] X. Luo, Y. Xu, W. Wang et al., "Towards enhancing stacked extreme learning machine with sparse autoencoder by correntropy," *Journal of The Franklin Institute*, 2017.

[32] M. A. Davenport and M. B. Wakin, "Analysis of orthogonal matching pursuit using the restricted isometry property," *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, vol. 56, no. 9, pp. 4395–4401, 2010.

[33] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Proceedings of the 27th Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 40–44, Pacific Grove, Calif, USA, November 1993.

[34] M. Nikolova and M. K. Ng, "Analysis of half-quadratic minimization methods for signal and image recovery," *SIAM Journal on Scientific Computing*, vol. 27, no. 3, pp. 937–966, 2005.

[35] R. T. Rockafellar, *Convex Analysis*, Princeton Mathematical Series, No. 28, Princeton University Press, Princeton, NJ, USA, 1970.

[36] A. Frank, *Uci machine learning repository*, 2010, http://archive.ics.uci.edu/ml.