Greedy Criterion in Orthogonal Greedy Learning

Lin Xu, Shaobo Lin, Jinshan Zeng, Xia Liu, Yi Fang, and Zongben Xu

***Abstract*—Orthogonal greedy learning (OGL) is a stepwise learning scheme that starts with selecting a new atom from a specified dictionary via the steepest gradient descent (SGD) and then builds the estimator through orthogonal projection. In this paper, we found that SGD is not the unique greedy criterion and introduced a new greedy criterion, called as "δ-greedy threshold" for learning. Based on this new greedy criterion, we derived a straightforward termination rule for OGL. Our theoretical study shows that the new learning scheme can achieve the existing (almost) optimal learning rate of OGL. Numerical experiments are also provided to support that this new scheme can achieve almost optimal generalization performance while requiring less computation than OGL.**

***Index Terms*—Generalization performance, greedy algorithms, greedy criterion, orthogonal greedy learning (OGL), supervised learning.**

I. Introduction

SUPERVISED learning focuses on synthesizing a function to approximate an underlying relationship between inputs and outputs based on finitely many input-output samples. Commonly, a system tackling supervised learning problems is called as a learning system. A standard learning system usually comprises a hypothesis space, an optimization strategy, and a learning algorithm. The hypothesis space is a family of parameterized functions providing a candidate set of estimators, the optimization strategy formulates an optimization problem to define the estimator based on samples, and the learning algorithm is an inference procedure that numerically solves the optimization problem.

Manuscript received January 22, 2017; revised January 30, 2017; accepted February 9, 2017. This work was supported in part by the Major State Basic Research Program under Grant 2013CB329404, and in part by the Natural Science Foundation of China under Grant 11131006 and Grant 91330204. This paper was recommended by Associate Editor S. Ventura.

L. Xu is with the Institute for Information and System Sciences, Xi'an Jiaotong University, Xi'an 710049, China, and also with the Department of Electrical and Computer Engineering, New York University Abu Dhabi, Abu Dhabi 129188, UAE (e-mail: xulinshadow@gmail.com; kylin@nyu.edu).

S. Lin is with the College of Mathematics and Information Science, Wenzhou University, Wenzhou 325035, China (e-mail: sblin1983@gmail.com).

J. Zeng is with the College of Computer Information Engineering, Jiangxi Normal University, Nanchang, Jiangxi 330022, China (e-mail: jinshanzeng@jxnu.edu.cn).

X. Liu is with the School of Sciences, Xi'an University of Technology, Xi'an 710049, China (e-mail: liuxia1232007@163.com).

Y. Fang is with the Department of Electrical and Computer Engineering, New York University Abu Dhabi, Abu Dhabi 129188, UAE (e-mail: yfang@nyu.edu).

Z. Xu is with the Institute for Information and System Sciences, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: zbxu@mail.xjtu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2017.2669259

Dictionary learning is a special learning system, whose hypothesis spaces are linear combinations of atoms in some given dictionaries. Here, the dictionary denotes a family of base learners [1]. For such type hypothesis spaces, many regularization schemes such as the bridge estimator [2], ridge estimator [3], and Lasso estimator [4] are commonly used optimization strategies. When the scale of dictionary is moderate (i.e., about hundreds of atoms), these optimization strategies can be effectively realized by various learning algorithms such as the regularized least squares (RLSs) algorithms [5], iterative thresholding algorithms [6], and iterative reweighted algorithms [7]. However, when faced with large input dictionary, a large portion of the aforementioned learning algorithms are time-consuming and even worse, they may cause the sluggishness of the corresponding learning systems.

Greedy learning or, more specifically, learning through greedy type algorithms provides a possible way to circumvent the drawbacks of the regularization methods [8]. Greedy algorithms are stepwise inference processes that start from a null model and solve heuristically the problem of making the locally optimal choice at each step with the hope of finding a global optimum. Within moderate number of iterations, greedy algorithms possess charming computational advantage compared with the regularization schemes [1]. This property triggers avid research activities of greedy algorithms in signal processing [9]–[11], inverse problems [12], [13], sparse approximation [14], [15], and machine learning [8], [16], [17].

Four most important elements of greedy learning we formulated are dictionary selection, greedy criterion, iterative strategy, and termination rule. This is essentially different from greedy approximation which focuses only on dictionary selection and iterative format issues [1]. Greedy learning concerns generalization performance more than approximation capability. In a nutshell, greedy learning can be regarded as a four-issue learning scheme.

1) *Dictionary Selection:* This issue devotes to inferring a dictionary from training data for a given learning task. As a classical topic of greedy approximation, there are a great deal of dictionaries available to greedy learning. Typical examples includes the radial basis functions (RBF) [18], wavelets [19], and decision trees [20].
2) *Greedy Criterion:* This issue regulates the criterion to choose a new atom from the dictionary in each greedy step. Besides the widely used steepest gradient descent (SGD) method [21], there are also many methods such as the weak greedy [22], thresholding greedy [1], and super greedy [23] to quantify the greedy criterion for approximation purpose. However, to the best of our knowledge, only the SGD criterion

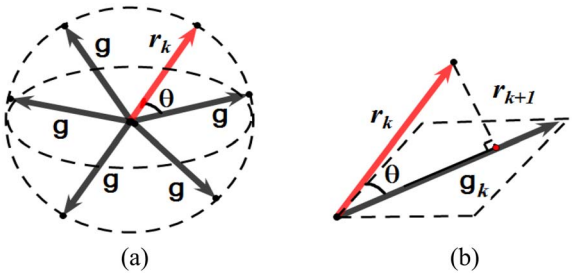


Fig. 1. Intuitive description of the greedy criterion. (a) Normalize the current residual r_k and atoms g to the unit ball. (b) Atom g_k possessing the smallest θ is regarded as the greediest one at each iteration.

is employed in greedy learning, since all the results in greedy approximation [1], [22], [23] imply that SGD is superior to other criteria.

3) *Iterative Format:* This issue focuses on how to define a new estimator based on the selected atoms. Similar to the dictionary selection, the iterative format issue is also a classical topic in greedy approximation. There are several types of iterative schemes [1]. Among these, three most commonly used iterative schemes are pure greedy [24], orthogonal greedy [21], and relaxed greedy formats [25]. Each iterative format possesses its own pros and cons [1]. For instance, compared with the orthogonal greedy format, pure greedy, and relaxed greedy formats have benefits in computation but suffer from either low convergence rate or small applicable scope.

4) *Termination Rule:* This issue depicts how to terminate the learning process. The termination rule is regarded as the main difference between greedy approximation and learning, which has been recently studied [8], [17], [26], [27]. For example, Barron *et al.* [8] proposed an l^0-based complexity regularization strategy as the termination rule, and Chen *et al.* [26] provided an l^1-based termination rule.

Orthogonal greedy learning (OGL) is a special greedy learning strategy. It selects a new atom based on SGD in each iteration and then constructs an estimator through orthogonal projecting to subspaces spanned by the selected atoms. It is well known that SGD needs to traverse the whole dictionary for selecting the most correlative atom, which leads to an insufferable computational burden when the scale of the dictionary is large. Moreover, OGL always searches the most correlative atom to realize the optimal approximation capability. As the samples are noised, the generalization performance of OGL is sensitive to the number of iterations. In other words, due to the SGD criterion, a slight turbulence of the number of atoms may lead to a great change of the generalization performance.

To overcome the above problems of OGL, a natural idea is to reregulate the criterion to choose a new atom by taking the greedy criterion issue into account. Fig. 1 is an intuitive description to quantify the greedy criterion, where r_k represents the residual at the kth iteration, g is an arbitrary atom from the dictionary and θ is the included angle between r_k and g. In Fig. 1(a), both r_k and g are normalized to the unit ball due to the greedy criterion focusing on the orientation rather than magnitude. The cosine of the angle θ (cosine similarity) is used to quantify the greedy criterion. As shown in Fig. 1(b), the atom g_k possessing the smallest θ is regarded to be the greediest one at each iteration in OGL.

Since the greedy criterion can be quantified by the cosine similarity, a preferable way to circumvent the aforementioned problems of OGL is to weaken the correlation by thresholding or regulating the cosine similarity. In particular, other than traversing the whole dictionary and then choosing the most correlative atom, we can select an arbitrary atom satisfying a predesigned thresholding condition. It should essentially reduce the complexity of OGL and make the learning process accelerated.

Different from other three issues, the greedy criterion issue, to the best of our knowledge, has not been noted for the learning purpose. The aim of this paper is to reveal the importance and necessity of studying the greedy criterion issue in OGL. The main contributions can be summarized as follows.

1) We argue that SGD is not the unique criterion for OGL. There are many other greedy criteria in greedy learning, which possess similar learning performance as SGD.
2) We use a new greedy criterion called δ-greedy threshold to quantify the correlation (or cosine similarity more precisely) in OGL. Although a similar criterion has already been used in greedy approximation [25], the innovation point of this paper is that we translate it into greedy learning to accelerate the learning process. Meanwhile, we also theoretically prove that, if the number of iterations is appropriately specified, then OGL with the δ-greedy threshold can reach the existing (almost) optimal learning rate of OGL [8].
3) Based on the δ-greedy threshold criterion, we can derive a straightforward termination rule for OGL and then provide a complete learning system called δ-thresholding orthogonal greedy learning (δ-TOGL). Different from the conventional termination rules that devote to searching the appropriate number of iterations based on the bias-variance balance principle [8], [27], this paper implies that this balance can also be attained through setting a suitable greedy threshold criterion. This phenomenon reveals the essential importance of the greedy criterion issue. We also present the theoretical justification of δ-TOGL.
4) Compared with other popular learning strategies such as the pure greedy learning (PGL) [1], [8], OGL, RLS [28], and fast iterative shrinkage-thresholding algorithm (FISTA) [29] through empirical studies, we provide a comprehensive analysis of δ-TOGL. The main advantage of δ-TOGL is that it can reduce the computational cost without sacrificing the generalization performance.

The rest of this paper is organized as follows. In Section II, we present a brief introduction of statistical learning theory and greedy learning. In Section III, we introduce the δ-greedy threshold criterion in OGL and provide its feasibility justification. In Section IV, based on the δ-greedy threshold criterion,

we propose a straightforward termination rule and the corresponding δ-TOGL system. The theoretical feasibility of the δ-TOGL system is also given in this section. In Section V, we present numerical simulation experiments to verify our arguments. In Section VI, δ-TOGL is further tested with real-world data. In Section VII, we close this paper with a brief conclusion.

II. Preliminaries

In this section, we present some preliminaries to serve as the basis for the following sections.

A. Statistical Learning Theory

Suppose that the samples $\mathbf{z} = (x_i, y_i)_{i=1}^m$ are drawn independently and identically from $Z \coloneqq X \times Y$ according to an unknown probability distribution ρ which admits the decomposition

$$\rho(x, y) = \rho_X(x)\rho(y|x). \tag{1}$$

Let $f : X \to Y$ be an approximation of the underlying relation between the input and output spaces. A commonly used measurement of the quality of f is the generalization error, defined by

$$\mathcal{E}(f) \coloneqq \int_Z (f(x) - y)^2 d\rho \tag{2}$$

which is minimized by the regression function [30]

$$f_\rho(x) \coloneqq \int_Y y d\rho(y|x). \tag{3}$$

Since the distribution ρ is unknown, the regression function f_ρ can not be computed directly. So the goal of learning is to find a best approximation of f_ρ.

Let $L^2_{\rho_X}$ be the Hilbert space of ρ_X square integrable functions on X, with norm $\|\cdot\|_\rho$. It is known that, for every $f \in L^2_{\rho_X}$, it holds that

$$\mathcal{E}(f) - \mathcal{E}(f_\rho) = \|f - f_\rho\|_\rho^2. \tag{4}$$

Without loss of generality, we assume $y \in [-M, M]$ almost surely. Thus, it is reasonable to truncate the estimator to $[-M, M]$. That is, if we define

$$\pi_M u \coloneqq \begin{cases} u, & \text{if } |u| \le M \\ M\text{sign}(u), & \text{otherwise} \end{cases} \tag{5}$$

as the truncation operator, where $\text{sign}(u)$ represents the sign function of u, then

$$\left\|\pi_M f_{\mathbf{z}} - f_\rho\right\|_\rho^2 \le \left\|f_{\mathbf{z}} - f_\rho\right\|_\rho^2. \tag{6}$$

B. Greedy Learning

Let H be a Hilbert space endowed with norm $\|\cdot\|_H$ and inner product $\langle\cdot,\cdot\rangle_H$. Let $\mathcal{D} = \{g\}_{g\in\mathcal{D}}$ be a given dictionary satisfying $\sup_{g\in D, x\in X} |g(x)| \le 1$. Denote $\mathcal{L}_1 = \{f : f = \sum_{g\in D} a_g g\}$ as a Banach space endowed with the norm

$$\|f\|_{\mathcal{L}_1} \coloneqq \inf_{\{a_g\}_{g\in\mathcal{D}}} \left\{ \sum_{g\in\mathcal{D}} |a_g| : f = \sum_{g\in\mathcal{D}} a_g g \right\}. \tag{7}$$

There exist several types of greedy algorithms [1]. The three most commonly used are the pure greedy algorithm (PGA) [24], orthogonal greedy algorithm (OGA) [21], and relaxed greedy algorithm [25]. These algorithms initialize with $f_0 \coloneqq 0$. The new approximation f_k ($k \ge 1$) is defined based on $r_{k-1} \coloneqq f - f_{k-1}$. In OGA, f_k is defined by

$$f_k = P_{V_{\mathbf{z},k}} f \tag{8}$$

where $P_{V_{\mathbf{z},k}}$ is the orthogonal projection onto the space $V_{\mathbf{z},k} = \text{span}\{g_1, \ldots, g_k\}$ and g_k is defined as

$$g_k = \arg\max_{g\in\mathcal{D}} |\langle r_{k-1}, g\rangle_H|. \tag{9}$$

Given $\mathbf{z} = (x_i, y_i)_{i=1}^m$, the empirical inner product and norm are defined by

$$\langle f, g\rangle_m \coloneqq \frac{1}{m}\sum_{i=1}^m f(x_i)g(x_i) \tag{10}$$

and

$$\|f\|_m^2 \coloneqq \frac{1}{m}\sum_{i=1}^m |f(x_i)|^2. \tag{11}$$

Setting $f_{\mathbf{z}}^0 = 0$, the four aforementioned issues are attended in OGL as follows.

1) *Dictionary Selection:* Select a suitable dictionary
$$\mathcal{D}_n \coloneqq \{g_1, \ldots, g_n\}$$
2) *Greedy Criterion:* Choose an atom satisfying the inequality
$$g_k = \arg\max_{g\in\mathcal{D}_n} |\langle r_{k-1}, g\rangle_m|. \tag{12}$$
3) *Iteration Format:* Compute the k-step estimator
$$f_{\mathbf{z}}^k = P_{V_{\mathbf{z},k}} f \tag{13}$$
where $P_{V_{\mathbf{z},k}}$ is the orthogonal projection onto $V_{\mathbf{z},k} = \text{span}\{g_1, \ldots, g_k\}$ in the metric of $\langle\cdot,\cdot\rangle_m$.
4) *Termination Rule:* Terminate the learning process when k satisfies a certain assumption.

III. Greedy Criterion in OGL

Given a real functional $V : H \to \mathbf{R}$, the Fréchet derivative of V at f, $V_f' : H \to \mathbf{R}$ is a linear functional such that for $h \in H$

$$\lim_{\|h\|_H \to 0} \frac{\left|V(f+h) - V(f) - V_f'(h)\right|}{\|h\|_H} = 0 \tag{14}$$

and the gradient of V as a map $\text{grad}V : H \to H$ is defined by

$$\langle \text{grad}V(f), h\rangle_H = V_f'(h), \text{ for all } h \in H. \tag{15}$$

The greedy criterion adopted in (12) is to find $g_k \in \mathcal{D}_n$ such that

$$\left\langle -\text{grad}(A_m)\left(f_{\mathbf{z}}^{k-1}\right), g_k \right\rangle = \sup_{g\in\mathcal{D}_n} \left\langle -\text{grad}(A_m)\left(f_{\mathbf{z}}^{k-1}\right), g \right\rangle \tag{16}$$

where $A_m(f) = \sum_{i=1}^m |f(x_i) - y_i|^2$. Therefore, the classical greedy criterion is based on the SGD of r_{k-1} with

respect to the dictionary $\mathcal{D}_n$. By normalizing the residual r_k, $k = 0, 1, 2, \ldots, n$, greedy criterion in (12) means to search g_k satisfying

$$g_k = \arg\max_{g\in\mathcal{D}_n} \frac{|\langle r_{k-1}, g\rangle_m|}{\|r_{k-1}\|_m}. \tag{17}$$

Geometrically, the current g_k minimizes the angle between $r_{k-1}/\|r_{k-1}\|_m$ and g, which is depicted in Fig. 1.

Recalling the definition of OGL, it is not difficult to verify that the angles satisfy

$$|\cos\theta_1| \le \cdots \le |\cos\theta_k| \le \cdots \le |\cos\theta_n| \tag{18}$$

or

$$\frac{|\langle r_0, g_1\rangle_m|}{\|r_0\|_m} \ge \cdots \ge \frac{|\langle r_{k-1}, g_k\rangle_m|}{\|r_{k-1}\|_m} \ge \cdots \ge \frac{|\langle r_{n-1}, g_n\rangle_m|}{\|r_{n-1}\|_m} \tag{19}$$

since $(|\langle r_{k-1}, g_k\rangle_m|/\|r_{k-1}\|_m) = |\cos\theta_k|$. If the algorithm stops at the kth iteration, then there exists a threshold $\delta \in [|\cos\theta_k|, |\cos\theta_{k+1}|]$ to quantify whether another atom should be added to construct the final estimator. To be detailed, if $|\cos\theta_k| \ge \delta$, then g_k is regarded as an "active atom" and can be selected to build the estimator, otherwise, g_k is a "dead atom" which should be discarded. Based on the above observations and motivated by the Chebshev greedy algorithm with thresholds [25], we are interested in selecting an arbitrary active atom, g_k, in $\mathcal{D}_n$, that is

$$\frac{|\langle r_{k-1}, g_k\rangle_m|}{\|r_{k-1}\|_m} > \delta. \tag{20}$$

If there is no g_k satisfying (20), then the algorithm terminates. We call the greedy criterion (20) as the δ-greedy threshold criterion. In practice, the number of active atoms is usually not unique. We can choose the first active atom satisfied with δ-greedy threshold criterion (20) at each greedy iteration to accelerate the algorithm. Once the active atom is selected, then the algorithm goes to the next greedy iteration and the active atom is redefined.

Through such a greedy-criterion, we can develop a new OGL scheme, called TOGL. The two corresponding elements of TOGL can be reformulated as follows:

1) *Greedy Definition:* Let g_k be an arbitrary (or the first) atom from $\mathcal{D}_n$ satisfying δ-greedy threshold criterion (20).
2) *Termination Rule:* Terminate the learning process either there is no atom satisfying δ-greedy threshold criterion (20) or k satisfies a certain assumption.

Compare with the greedy criterion in OGL and TOGL, we find that the classical greedy criterion (12) in OGL always selects the greediest atom at each greedy iteration. While, δ-greedy threshold criterion (20) in TOGL slows down the speed of gradient descent and therefore may conduct a more flexible model selection strategy. According to the bias and variance balance principle [31], the bias decreases while the variance increases as a new atom is selected to build the estimator. If a lower-correlation atom is added, then the bias decreases slower and the variance also increases slower. Then, the balance can be achieved in TOGL within a more gradual flavor than OGL. Moreover, δ-greedy threshold criterion (20) can also provides a natural termination rule that if no atom, g, in $\mathcal{D}_n$ satisfy δ-greedy threshold criterion (20) as

$$\max_{g\in\mathcal{D}_n} |\langle r_k, g\rangle_m| \le \delta\|r_k\|_m \tag{21}$$

then the algorithm terminates.

Now we present a theoretical assessment of TOGL. At first, we give a few notations and concepts, which will be used in the rest part of this paper. For $r > 0$, the space $\mathcal{L}^r_{1,\mathcal{D}_n}$ is defined to be the set of all functions f such that, there exists a $h \in \text{span}\{\mathcal{D}_n\}$ satisfying

$$\|h\|_{\mathcal{L}_1(\mathcal{D}_n)} \le \mathcal{B}, \text{ and } \|f - h\| \le \mathcal{B}n^{-r} \tag{22}$$

where $\|\cdot\|$ denotes the uniform norm for the continuous function space $C(X)$. The infimum of all $\mathcal{B}$ satisfying (22) defines a norm (for f) on $\mathcal{L}^r_{1,\mathcal{D}_n}$. Equation (22) defines an interpolation space and is a natural assumption for the regression function in greedy learning [8]. This assumption has already been adopted to analyze the generalization performance of greedy learning [8], [17], [27]. Theorem 1 illustrates the performance of TOGL and consequently, reveals the feasibility of the greedy criterion in δ-greedy threshold criterion (20). The proof of Theorem 1 is put in the Appendix.

Theorem 1: Let $0 < t < 1$, $0 < \delta \le 1/2$, and $f^{k,\delta}_{\mathbf{z}}$ be the estimator deduced by TOGL. If $f_\rho \in \mathcal{L}^r_{1,\mathcal{D}_n}$, then there exits a $k^* \in \mathbf{N}$ such that

$$\mathcal{E}\left(\pi_M f^{k^*,\delta}_{\mathbf{z}}\right) - \mathcal{E}(f_\rho) \le C\mathcal{B}^2\left(\left(m\delta^2\right)^{-1}\log m \log\frac{1}{\delta}\log\frac{2}{t} + \delta^2 + n^{-2r}\right)$$

holds with probability at least $1 - t$, where C is a positive constant depending only on d and M.

If $\delta = \mathcal{O}(m^{-1/4})$, and the size of dictionary, n, is selected to be large enough, i.e., $n \ge \mathcal{O}(m^{(1/4r)})$, then Theorem 1shows that the generalization error of $\pi_M f^{k^*,\delta}_{\mathbf{z}}$ is asymptotic to $\mathcal{O}(m^{-1/2}(\log m)^2)$. Up to a logarithmic factor, this bound is the same as that of Barron *et al.* [8], which is the best known bound in existing literature of OGL. This implies that weakening the correlation in OGL is a feasible way to avoid traversing the dictionary. It should also be pointed out that different from OGL [8], there are two parameters, k and δ, in TOGL. The termination rule in TOGL concerning k is necessary and is used to avoid certain extreme cases in practice. In fact, only using the termination rule (21) may drive the algorithm to select all atoms from $\mathcal{D}_n$. As Fig. 2 shows, if the target function f is almost orthogonal to the space spanned by the dictionary and the atoms in the dictionary are almost linear dependent, then the selected δ should be too small to distinguish which is the active atom. Consequently, the corresponding learning scheme selects all atoms of the dictionary, and therefore, degrades the generalization performance of OGL.

Therefore, Theorem 1 only presents a theoretical verification that introducing the δ-greedy threshold to measure the correlation does not essentially degrade the generalization performance of OGL. However, taking practical aspects into account, simultaneously tunning two main parameters in TOGL should be a tough task.

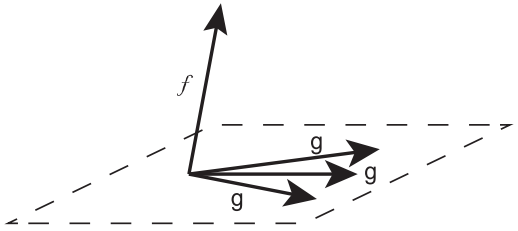


Fig. 2. Necessity of termination rule concerning k in TOGL.

IV. δ-Thresholding Orthogonal Greedy Learning

In the previous section, we developed a new greedy learning scheme called as TOGL and theoretically verified its feasibility. However, there are two main parameters (i.e., the value of threshold δ and iteration k) should be simultaneously fine-tuned. It puts more pressure on parameter selection, which may dampen the spirits of practitioners. Given this, we further propose a termination rule only based on the value of threshold δ. Notice that, the value $\|r_{k-1}\|_m/\|y(\cdot)\|_m$ becomes smaller and smaller along the selection of more and more active atoms, where $y(\cdot)$ is a function satisfying $y(x_i) = y_i, i = 1, \ldots, m$. Then, an advisable termination rule is to use δ to quantify $\|r_{k-1}\|_m/\|y(\cdot)\|_m$. Therefore, we append another termination rule as

$$\|r_{k-1}\|_m \leq \delta \|y(\cdot)\|_m \tag{23}$$

to replace the previous termination rule concerning k in TOGL. Based on it, a new termination rule can be obtained.

1) *Termination Rule:* Terminate the learning process if either (23) holds or there is no atom satisfying δ-greedy threshold criterion (20). That is

$$\max_{g\in\mathcal{D}_n} |\langle r_k, g\rangle_m| \leq \delta \|r_k\|_m \text{ or } \|r_k\|_m \leq \delta \|f\|_m. \tag{24}$$

Then we present a new learning system named δ-TOGL as Algorithm 1.

The implementation of OGL requires traversing the whole dictionary, which has a complexity of $\mathcal{O}(mn)$. Inverting a $k\times k$ matrix in orthogonal projection has a complexity of $\mathcal{O}(k^3)$. Thus, the kth iteration of OGL has a complexity of $\mathcal{O}(mn + k^3)$. In step 2 of δ-TOGL, g_k is an arbitrary atom from $\mathcal{D}_n$ satisfying the δ-greedy threshold condition. It motivates us to select a random atom from $\mathcal{D}_n$ satisfying δ-greedy threshold criterion (20). Suppose we use $\hat{n}$ to denote the number of atoms δ-TOGL traversed, generally $\hat{n} \ll n$. Thus, the complexity of δ-TOGL is smaller than $\mathcal{O}(mn + k^3)$. In fact, it usually requires a complexity of $\mathcal{O}(m + k^3)$, and gets a complexity of $\mathcal{O}(mn + k^3)$ only for the worst case [here the worst case means all atoms in dictionary satisfied (20)]. Furthermore, there are an additional termination rule (24) in δ-TOGL compared with the conventional OGL. The benefit is a smaller number of iterations in δ-TOGL generally, except the value of threshold δ is a really small positive number tending to 0. Thus, δ-TOGL can essentially reduce the complexity of OGL, especially when n is large. The memory requirements of OGL and δ-TOGL are the same as $\mathcal{O}(mn)$ for inner product operation.

Algorithm 1 δ-TOGL

Inputs: Training data $\mathbf{z} = (x_i, y_i)_{i=1}^m$.
Outputs: Function estimator $f_{\mathbf{z}}^{\delta}$.
Step 1 (Initialization):
Given dictionary $\mathcal{D}_n$ and a proper greedy threshold δ.
Set initial estimator $f_0 = 0$ and iteration $k \coloneqq 0$.
Step 2 (δ-greedy threshold):
Select g_k be an arbitrary atom from $\mathcal{D}_n$ satisfying

$$\frac{|\langle r_{k-1}, g_k\rangle_m|}{\|r_{k-1}\|_m} > \delta.$$

Step 3 (Orthogonal projection):
Let $V_{\mathbf{z},k} = \text{Span}\{g_1, \ldots, g_k\}$. Compute $f_{\mathbf{z}}^{\delta}$ as:

$$f_{\mathbf{z}}^{\delta} = P_{\mathbf{z},V_{\mathbf{z},k}}(y).$$

The residual: $r_k \coloneqq y - f_{\mathbf{z}}^{\delta}$, where $P_{\mathbf{z},V_{\mathbf{z},k}}$ is the orthogonal projection onto space $V_{\mathbf{z},k}$ in the criterion of $\langle\cdot,\cdot\rangle_m$.
Step 4 (Termination rule):
If termination rule is satisfied as:

$$\max_{g\in\mathcal{D}_n} |\langle r_k, g\rangle_m| \leq \delta \|r_k\|_m \text{ or } \|r_k\|_m \leq \delta \|f\|_m,$$

then algorithm terminates and outputs the final estimator $f_{\mathbf{z}}^{\delta}$. Otherwise, return to Step 2 and $k \coloneqq k + 1$.

The following theorem shows that if the value of threshold δ is appropriately tuned, then the δ-TOGL estimator $f_{\mathbf{z}}^{\delta}$ can also realize the (almost) optimal generalization performance of OGL and TOGL. Please see the Appendix for the proof of Theorem 2.

Theorem 2: Let $0 < t < 1$, $0 < \delta \leq 1/2$, and $f_{\mathbf{z}}^{\delta}$ be defined in Algorithm 1. If $f_\rho \in \mathcal{L}^r_{1,\mathcal{D}_n}$, then the inequality

$$\mathcal{E}\left(\pi_M f_{\mathbf{z}}^{\delta}\right) - \mathcal{E}\left(f_\rho\right) \leq C\mathcal{B}^2\left(\left(m\delta^2\right)^{-1} \log m \log\frac{1}{\delta}\log\frac{2}{t} + \delta^2 + n^{-2r}\right)$$

holds with probability at least $1 - t$, where C is a positive constant depending only on d and M.

If $n \geq \mathcal{O}(m^{(1/4r)})$ and $\delta = \mathcal{O}(m^{-1/4})$, then the learning rate in Theorem 2 asymptotically equals to $\mathcal{O}(m^{-1/2}(\log m)^2)$, which is the same as that of Theorem 1. Therefore, Theorem 2 implies that using (23) only concerning δ to fully replace the termination rule concerning k is theoretically feasible.

The most important trait of Theorem 2 is that it provides a totally different way to circumvent the overfitting phenomenon of OGL. As we know that the termination rule is crucial for OGL, but designing an effective one is a tricky problem. Almost all the previous studies [8], [26], [27] concerning on the termination rule in OGL attempted to control the number of iterations directly. Since the generalization performance of OGL is sensitive to the iterations, the results are sometimes unsatisfactory. The termination rule (23) employed in this paper is based on the study of the "greedy-criterion" issue of greedy learning. Theorem 2 shows that, besides controlling the number of iterations directly, setting a greedy threshold to redefine the greedy criterion can also conduct an effective termination rule. Theorem 2 implies that this new termination

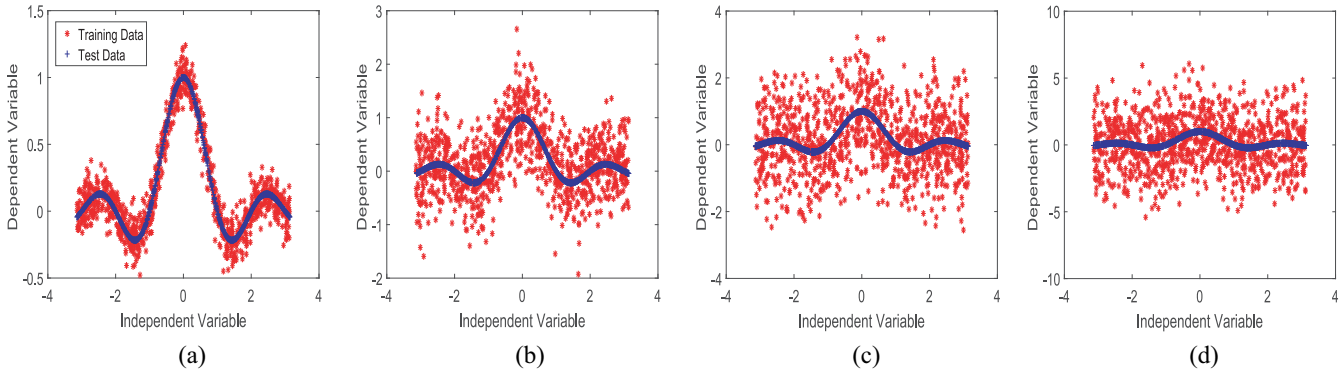


Fig. 3. Simulated training and test samples. The learning task becomes more difficult with respect to increasing of the noise. (a) Noise level $\sigma_1 = 0.1$. (b) $\sigma_2 = 0.5$. (c) $\sigma_3 = 1$. (d) $\sigma_4 = 2$.

rule theoretically works as well as others. Furthermore, since the new criterion slows down the changes of bias and variance, the generalization performance of δ-TOGL is more stable to δ than that of OGL to k.

V. Simulation Verifications

In this section, a series of simulations are carried out to verify our theoretical assertions. First, we introduce the simulation settings, including the data sets, dictionary, greedy criteria, and experimental environment. Second, we analyze the greedy criteria in OGL. Third, we study δ-greedy threshold criterion in δ-TOLG Finally, we compare δ-TOGL with other widely used dictionary-based learning methods and verify its feasibility.

A. Simulation Settings

Throughout the simulations, let $\mathbf{z} = \{(x_i, y_i)\}_{i=1}^{m_1}$ be the training samples with independent variable $\{x_i\}_{i=1}^{m_1}$ being drawn independently and identically according to the uniform distribution on $[-\pi, \pi]$ and the corresponding dependent variable $y_i = f_\rho(x_i) + \mathcal{N}(0, \sigma^2)$, where

$$f_\rho(x) = \frac{\sin x}{x}, \quad x \in [-\pi, \pi]. \tag{25}$$

In each simulation, we use the RBF [18] to build up the dictionary

$$\left\{ e^{-\|x-t_i\|^2/\eta^2} : i = 1, \ldots, n \right\} \tag{26}$$

where $\{t_i\}_{i=1}^n$ are drawn according to the uniform distribution in $[-\pi, \pi]$. The learning performance of different learning schemes are then tested by using the root mean squared error (RMSE) criterion

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{m_2} \left(\hat{y}_i - y_i^{(t)}\right)^2}{m_2}} \tag{27}$$

where $\hat{y}_i$ is the resultant estimator and $y_i^{(t)} = f_\rho(x_i^{(t)})$ is taken from the test set $\mathbf{z}_{test} = \{(x_i^{(t)}, y_i^{(t)})\}_{i=1}^{m_2}$. Since the aim of each simulation is to compare δ-TOGL with other learning methods under the same dictionary, we just set $m_1 = 1000$, $m_2 = 1000$, $n = 300$, and $\eta = 1$ throughout the simulations unless otherwise stated. In order to make the simulated learning task more "real," four levels of noise $\sigma_1 = 0.1$, $\sigma_2 = 0.5$, $\sigma_3 = 1$, and $\sigma_4 = 2$ has been added to all the training samples while test data remain noise-free. Fig. 3 shows the the simulated training and test samples, we can find that the learning problem becomes more difficult with respect to increasing of the noise.

We use four different criteria to select the new atom in each greedy iteration

$$g_k := \arg\max_{g \in \mathcal{D}_n} \left| \langle r_{k-1}, g \rangle_m \right|$$

$$g_k := \arg \text{second} \max_{g \in \mathcal{D}_n} \left| \langle r_{k-1}, g \rangle_m \right|$$

$$g_k := \arg \text{third} \max_{g \in \mathcal{D}_n} \left| \langle r_{k-1}, g \rangle_m \right|$$

and

$$g_k \text{ randomly selected from } \mathcal{D}_n.$$

Here, arg second max and arg third max mean the values of $|\langle r_{k-1}, g\rangle_m|$ reach the second and third largest values, respectively. Randomly selected means to randomly select g_k from the dictionary. We use four abbreviations OGL1, OGL2, OGL3, and OGLR to represent the corresponding greedy criteria in OGL, respectively. Accordingly, δ-TOGL1, δ-TOGL2, δ-TOGL3, and δ-TOGLR are used to denote the corresponding greedy criteria in δ-TOGL. Noticing that, selecting g_k in δ-TOGL also should satisfy δ-greedy threshold criterion $(|\langle r_{k-1}, g_k\rangle_m| / \|r_{k-1}\|_m) > \delta$.

All numerical studies are implemented in MATLAB R2015a on a Windows personal computer with dual-core i7-3770 (3.40 GHz) CPUs and 16GB of RAM. All the statistics are averaged based on 50 independent trials.

B. Greedy Criteria in OGL

In this part, we analyze the role of the greedy criterion in OGL by comparing the performance of OGL1, OGL2, OGL3, and OGLR.

Fig. 4 shows the performance of OGL with four different greedy criteria. Since all the values of the optimal iteration k (i.e., k^*_{OGL}) are small (less than 15), so we only plot the figures with $k \in [0; 15]$ to present more details around the optimal value. First, we can see that the performance of OGL is very sensitive to iteration and its performance will be sharply deteriorated when the number of iterations becomes larger. Afterward,

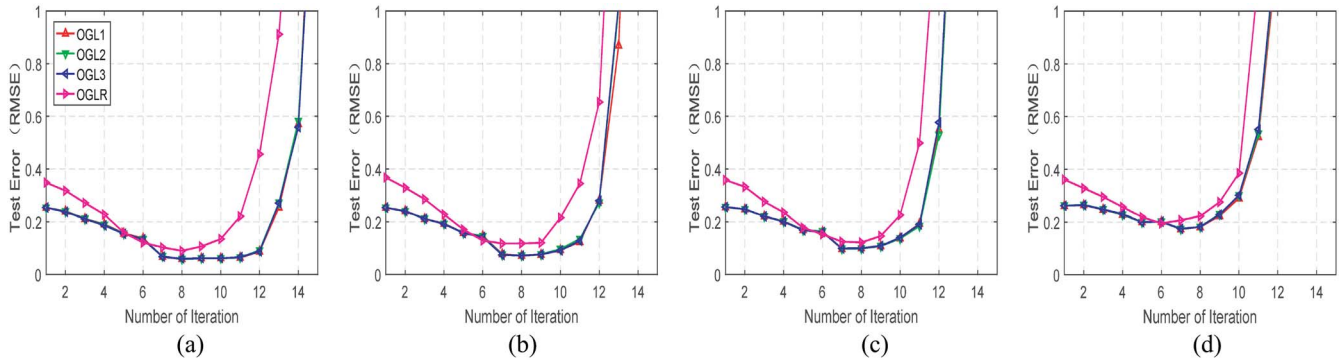


Fig. 4. Generalization performance of OGL with four different greedy criteria. (a) Noise level $\sigma_1 = 0.1$. (b) $\sigma_2 = 0.5$. (c) $\sigma_3 = 1$. (d) $\sigma_4 = 2$.

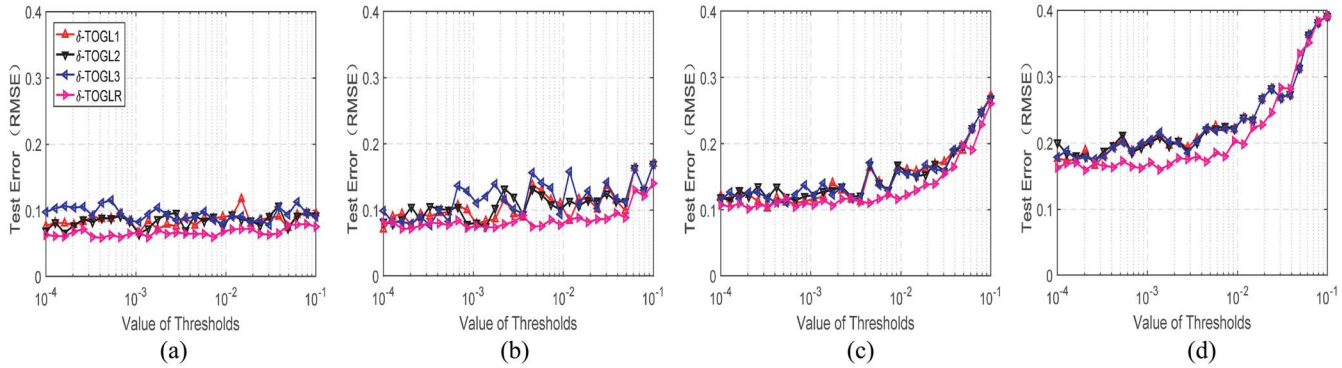


Fig. 5. Generalization performance of δ-TOGL with four different greedy criteria. (a) Noise level $\sigma_1 = 0.1$. (b) $\sigma_2 = 0.5$. (c) $\sigma_3 = 1$. (d) $\sigma_4 = 2$.

TABLE I
COMPARISONS OF OGL WITH FOUR GREEDY CRITERIA

Method	TestRMSE	k^*_{OGL}	Method	TestRMSE	k^*_{OGL}
$\sigma = 0.1$			$\sigma = 0.5$		
OGL1	0.0545 (0.0219)	8.9 (1.3)	OGL1	0.0671 (0.0223)	8.8 (1.4)
OGL2	0.0539 (0.0223)	9.1 (1.3)	OGL2	0.0658 (0.0215)	8.7 (1.5)
OGL3	0.0548 (0.0224)	9.0 (1.4)	OGL3	0.0639 (0.0188)	8.9 (1.4)
OGLR	0.0649 (0.0252)	9.5 (2.0)	OGLR	0.0817 (0.0332)	8.3 (1.6)
Method	TestRMSE	k^*_{OGL}	Method	TestRMSE	k^*_{OGL}
$\sigma = 1$			$\sigma = 2$		
OGL1	0.0935 (0.0210)	7.9 (1.1)	OGL1	0.1651 (0.0284)	6.9 (1.2)
OGL2	0.0932 (0.0214)	7.9 (0.9)	OGL2	0.1652 (0.0281)	6.9 (1.2)
OGL3	0.0933 (0.0210)	7.9 (0.9)	OGL3	0.1652 (0.0281)	7.0 (1.2)
OGLR	0.1029 (0.0268)	7.8 (1.3)	OGLR	0.1761 (0.0241)	6.7 (1.5)

we also find that OGL1, OGL2, and OGL3 have similar performance, while OGLR performs worse. This phenomenon shows that SGD (or OGL1) is not the unique greedy criterion for learning, meanwhile, random selecting atom in OGL is not a wise choice. It implies the necessity to study the greedy criterion issue for learning purpose. Detailed comparisons are listed in the Table I. Here TestRMSE denotes the optimal generalization performance (in RMSE), where the parameter k^*_{OGL} is selected according to the test data (or TestRMSE) directly. The standard deviation of optimal TestRMSE and parameter are also listed in the corresponding brackets.

C. δ-Greedy Threshold Criterion in δ-TOGL

Now, we begin to examine the performance of δ-TOGL. From OGL to δ-TOGL, the main parameter changes from the number of iteration k to the value of greedy threshold δ. Similar to Fig. 4, we also consider the relationship between the performance and parameter of δ-TOGL in Fig. 5. We plot the range of δ in $[10^{-4}, 10^{-1}]$. Notice that the plot range of test error (RMSE) in Fig. 5 (i.e., [0, 0.4]) is much smaller than that in Fig. 4 (i.e., [0, 1]) for distinguishing different performance curves.

From the figures, we can see that different from previous Fig. 4, now the performance of δ-TOGLR is better and more robust than δ-TOGL1, δ-TOGL2, and δ-TOGL3 in various noise settings. The main reason is that δ-TOGLR random selecting a new atom satisfied with the δ-greedy threshold criterion (20). This constrained randomness can suppress noise interference to some extent, and thus achieve better and robust performance.

Detailed comparisons are also listed in Table II. Here TestRMSE denotes the optimal generalization performance (in RMSE), where the parameter δ^* is selected according to the test data (or TestRMSE) directly. The standard deviation of optimal TestRMSE and parameter are listed in the corresponding brackets. The numbers in bold represent the best result compared with others in the same experimental settings. We also record the number of iteration $k^*_{\delta\text{-TOGL}}$ corresponding to δ^* for comparisons, although δ-TOGL has no part in adjusting this parameter. From the result, we can clearly find that the performance of δ-TOGLR is better

TABLE II
COMPARISONS OF δ-TOGL WITH FOUR GREEDY CRITERIA

Method	δ^*	TestRMSE	$k^*_{\delta\text{-TOGL}}$
$\sigma = 0.1$			
δ-TOGL1	0.0139 (0.0255)	0.0467 (0.0178)	11.4 (4.5)
δ-TOGL2	0.0128 (0.0192)	0.0470 (0.0207)	11.6 (4.6)
δ-TOGL3	0.0145 (0.0224)	0.0479 (0.0212)	10.7 (3.5)
δ-TOGLR	**0.0105 (0.0237)**	**0.0396 (0.0208)**	**9.8 (1.2)**
$\sigma = 0.5$			
δ-TOGL1	0.0054 (0.0122)	0.0569 (0.0141)	8.9 (2.8)
δ-TOGL2	0.0074 (0.0125)	0.0587 (0.0139)	10.2 (4.3)
δ-TOGL3	0.0112 (0.0177)	0.0597 (0.0150)	8.9 (2.8)
δ-TOGLR	**0.0052 (0.0121)**	**0.0562 (0.0147)**	**8.2 (0.8)**
$\sigma = 1$			
δ-TOGL1	0.0069 (0.0095)	0.0814 (0.0194)	10.0 (5.3)
δ-TOGL2	0.0068 (0.0077)	0.0832 (0.0180)	8.7 (4.0)
δ-TOGL3	0.0068 (0.0072)	0.0824 (0.0187)	8.5 (4.3)
δ-TOGLR	**0.0052 (0.0065)**	**0.0810 (0.0185)**	**7.9 (1.5)**
$\sigma = 2$			
δ-TOGL1	0.0025 (0.0039)	0.1355 (0.0431)	11.0 (6.2)
δ-TOGL2	0.0039 (0.0060)	0.1329 (0.0443)	8.9 (4.9)
δ-TOGL3	0.0019 (0.0019)	0.1334 (0.0451)	10.5 (5.7)
δ-TOGLR	**0.0037 (0.0017)**	**0.1301 (0.0349)**	**7.2 (1.2)**

and more robust in parameter compared with other δ-TOGL variants.

D. Compare With Other Learning Schemes

We than compare δ-TOGLR with other dictionary-based learning schemes such as the PGL [20], OGL [8], ridge regression [3], and Lasso [4]. We use $\mathcal{L}_2$ regularized least-square (RLS) solution for ridge regression and FISTA algorithm for Lasso [29].

Firstly, we introduce the parameter settings of the corresponding learning schemes. For OGL, the maximum number of iterations equals to the size of dictionary. And for the greedy threshold parameters δ-TOGLR, we also use 20 equally spaced values of δ in logarithmic space within $[10^{-4}, 10^{-1}]$. Due to the convergence rate of PGL is more slower than OGL [8], [21], the maximum number of iterations of PGL is set as 10 000 for better generalization performance. The regularization parameter λ in RLS and FISTA is also chosen from a 50 points set whose elements are uniformly localized in $[10^{-4}, 1]$. All the parameters, i.e., the number of iterations k in PGL or OGL, the regularization parameter λ in RLS or FISTA, and the greedy threshold δ in δ-TOGLR are all selected according to test dataset (or test RMSE) directly, since we mainly focus on the impact of the theoretically optimal parameter rather than validation techniques.

The compared results are listed in Table III, where the standard errors of testRMSE are also reported (numbers in parentheses). The sparsity means the number of atoms the corresponding algorithm employed and running time (in s) implies the whole cost (training and test cost) the algorithm paid. From Table III, we first observe that the sparsities of greedy-type strategies are obviously far smaller than regularization-based methods, while they enjoy better performance. It empirically verifies that greedy-type algorithms are more suitable for redundant dictionary learning, which is also empirically consistent with the work of Barron *et al.* [8]. Furthermore, we also find that, although the performance of such three greedy-type algorithms (PGL, OGL, and δ-TOGLR) are similar, δ-TOGLR has a big advantage in running time and sparsity.

TABLE III
COMPARING δ-TOGLR WITH OTHER SCHEMES

Method	Parameter	TestRMSE	Sparsity	Running time
Regression function $sinc$, dictionary $\mathcal{D}_n, n = 300$, noise level $\sigma = 0.1$				
PGL	$k = 81$	0.0434 (0.0172)	81.0	31.3 (1.2)
OGL	$k = 10$	0.0452 (0.0241)	10.0	13.2 (0.4)
δ-TOGLR	$\delta = 0.0113$	0.0421 (0.0121)	9.2	3.5 (0.1)
$\mathcal{L}_2$(RLS)	$\lambda = 0.0012$	0.0412 (0.0198)	300.0	0.6 (0.1)
$\mathcal{L}_1$(FISTA)	$\lambda = 0.0007$	0.0418 (0.0192)	292.5	61.8 (2.1)
Regression function $sinc$, dictionary $\mathcal{D}_n, n = 1000$, noise level $\sigma = 0.1$				
PGL	$k = 217$	0.0422 (0.0142)	217.0	126.7 (3.1)
OGL	$k = 9$	0.0415 (0.0151)	9.0	71.1 (1.1)
δ-TOGLR	$\delta = 0.0201$	0.0384 (0.0182)	10.1	5.1 (0.1)
$\mathcal{L}_2$(RLS)	$\lambda = 0.0012$	0.0489 (0.0103)	1000.0	6.1 (0.1)
$\mathcal{L}_1$(FISTA)	$\lambda = 0.0006$	0.0481 (0.0179)	821.2	112.7 (2.3)
Regression function $sinc$, dictionary $\mathcal{D}_n, n = 2000$, noise level $\sigma = 0.1$				
PGL	$k = 221$	0.0367 (0.0136)	221.0	236.2 (6.2)
OGL	$k = 9$	0.0351 (0.0143)	9.0	374.7 (5.9)
δ-TOGLR	$\delta = 0.0112$	0.0326 (0.0129)	13.5	6.7 (0.5)
$\mathcal{L}_2$(RLS)	$\lambda = 0.0051$	0.0423 (0.0128)	2000.0	38.2 (1.4)
$\mathcal{L}_1$(FISTA)	$\lambda = 0.0012$	0.0412 (0.0181)	1151.2	121.3 (3.2)

VI. REAL DATA EXPERIMENTS

We have verified that δ-TOGL is feasible in simulations. Especially, δ-TOGLR possesses both good generalization performance and the lowest computation complexity. Now, we begin to verify the performance (also in RMSE) and running time (in s) of δ-TOGLR and further compare it with other dictionary-based learning methods including PGL, OGL, RLS, and FISTA on five real data sets.

The first dataset is the Prostate cancer dataset [32]. The data set consists of the medical records of 97 patients who have received a radical prostatectomy. The predictors are eight clinical measures and one response variable. The second dataset is the Diabetes data set [33]. This data set contains 442 diabetes patients that are measured on ten independent variables and one response variable. The third one is the Boston Housing data set created from a housing values survey in suburbs of Boston by Harrison and Rubinfeld [34]. The Boston Housing dataset contains 506 instances which include 13 attributions and one response variable. The fourth one is the concrete compressive strength (CCS) dataset [35], which contains 1030

TABLE IV
COMPARATIVE RESULTS OF PERFORMANCE AND RUNNING TIME ON FIVE REAL DATA SETS

Method \ Dataset	Prostate	Diabetes	Housing	CCS	Abalone
Dictionary size	$n = 50$	$n = 220$	$n = 255$	$n = 520$	$n = 2100$
Performance in RMSE (average and standard deviation)					
δ-TOGLR	0.4208 (0.0112)	55.1226 (1.0347)	4.0450 (0.4256)	7.1279 (0.3294)	2.2460 (0.0915)
PGL	0.4280 (0.0081)	56.3125 (2.0542)	4.0716 (0.2309)	11.2803 (0.0341)	2.5880 (0.0106)
OGL	0.5170 (0.0119)	54.6518 (2.8700)	3.9447 (0.1139)	6.0128 (0.1203)	2.1725 (0.0088)
RLS	0.4415 (0.0951)	57.3886 (1.5854)	3.9554 (0.3236)	9.8512 (0.2693)	2.2559 (0.0514)
FISTA	0.6435 (0.0151)	61.7636 (2.5811)	5.1845 (0.1859)	12.8127 (0.3019)	3.4161 (0.0774)
Running time in seconds (average and standard deviation)					
δ-TOGLR	0.58 (0.01)	1.11 (0.02)	0.89 (0.01)	0.82 (0.01)	4.22 (0.04)
PGL	41.93 (1.12)	49.06 (1.09)	52.04 (1.12)	79.93 (1.26)	193.97 (1.87)
OGL	0.16 (0.01)	1.11 (0.03)	1.42 (0.03)	7.46 (0.01)	787.2 (1.22)
RLS	0.15 (0.01)	0.27 (0.01)	0.33 (0.01)	1.20 (0.02)	42.59 (0.24)
FISTA	0.52 (0.06)	1.11 (0.32)	1.40 (0.03)	9.04 (0.12)	257.8 (1.62)

instances including eight quantitative independent variables and one dependent variable. The fifth one is the Abalone dataset [36] collected for predicting the age of abalone from physical measurements. The data set contains 4177 instances which were measured on eight independent variables and one response variable.

We randomly divide all the real data sets into two disjoint equal parts. The first half serves as the training set and the second half serves as the test set. We use the Z-score standardization method [37] to normalize the data sets, in order to avoid the error caused by considerable magnitude difference among data dimensions. For each real data experiment, Gaussian RBF is also used to build up the dictionary

$$\left\{e^{-\|x-t_i\|^2/\eta^2} : i = 1, \ldots, n\right\} \tag{28}$$

where $\{t_i\}_{i=1}^n$ are drawn as the training samples themselves, thus the size of dictionary equals to the training samples. We set the standard deviation of RBF as $\eta = (d_{\max}/\sqrt{2n})$, where $d_{\max}$ is maximum distance among all centers $\{t_i\}_{i=1}^n$, in order to avoid the RBF is too sharp or flat.

Table IV documents the performance and running time of the corresponding algorithms on five real data sets. We find that, for the small-scale dictionary, i.e., for the Prostate data set, although δ-TOGLR can achieve good performance, its running time is more than OGL and RLS. This is attributed to additional parameter-selection cost in δ-TOGLR. In fact, for each candidate threshold parameter δ, a different iteration of the algorithm is needed run from scratch, which seems to cancel the major computational advantage of δ-TOGLR in small size dictionary learning. However, we also notice that, when the size of dictionary increased (i.e., diabetes, housing, and CCS), δ-TOGLR begin to gradually surpass the other methods in computation with maintaining good performance. Especially in Abalone data set, δ-TOGLR dominates other methods with a large margin in computation and still possesses good performance.

VII. CONCLUSION

In this paper, we study the greedy criteria in OGL. The main contributions can be concluded in four aspects.

Firstly, we propose that the SGD is not the unique greedy criterion to select atoms from dictionary in OGL, which paves a new way for exploring greedy criterion in greedy learning. To the best of our knowledge, this may be the first work concerning the greedy criterion issue in the field of supervised learning. Secondly, motivated by a series of previous researches of Temlyakov [1], [22], [23], [25] in greedy approximation, we eventually use the δ-greedy threshold criterion to quantify the correlation for the learning purpose. Our theoretical result shows that OGL with such a greedy criterion yields a learning rate as $m^{-1/2}(\log m)^2$, which is almost the same as that of the classical SGD-based OGL [8]. Thirdly, based on the δ-greedy threshold criterion, we derive a terminal rule for the corresponding OGL and thus provide a complete new learning scheme called as δ-TOGL. We also present the theoretical demonstration that δ-TOGL can reach the existing (almost) optimal learning rate [8] just as the iteration-based termination rule dose. Finally, we analyze the generalization performance of δ-TOGL and compare it with other popular dictionary-based learning methods through plenty of numerical experiments. The empirical results verify that the δ-TOGL is a promising learning scheme, which reduces the computational cost without sacrificing the generalization performance.

Future work is required to enable such a trend. Among the many possible research directions we mention three: 1) a study of the heuristic strategy for a suitable threshold value in δ-TOGL; 2) faster implementation of the algorithm (i.e., parallel processing for atoms in dictionary and matrix factorization for inverting a huge matrix in orthogonal projection step); and

3) handing the scalability problem of δ-TOGL, when tuning to work with large-scale dictionary.

APPENDIX
PROOF OF THEOREM 2

Since Theorem 1 can be derived from Theorem 2 directly, we only prove Theorem 2. The methodology of proof is somewhat standard in learning theory. In fact, we use the error decomposition strategy [17] to divide the generalization error into approximation error, sample error and hypothesis error. The main difficulty of the proof is to bound the hypothesis error. The main tool to bound it is borrowed from [25].

In order to give an error decomposition strategy for $\mathcal{E}(f_{\mathbf{z}}^k)-\mathcal{E}(f_\rho)$, we need to construct a function $f_k^* \in \mathrm{span}(D_n)$ as follows. Since $f_\rho \in \mathcal{L}^r_{1,\mathcal{D}_n}$, there exists a $h_\rho := \sum_{i=1}^n a_i g_i \in \mathrm{Span}(\mathcal{D}_n)$ such that

$$\|h_\rho\|_{\mathcal{L}_{1,\mathcal{D}_n}} \le \mathcal{B}, \text{ and } \|f_\rho - h_\rho\| \le \mathcal{B}n^{-r}. \tag{29}$$

Define

$$f_0^* = 0,\ f_k^* = \left(1-\frac{1}{k}\right)f_{k-1}^* + \frac{\sum_{i=1}^n |a_i|\|g_i\|_\rho}{k} g_k^* \tag{30}$$

where

$$g_k^* := \arg\max_{g\in\mathcal{D}_n'} \left\langle h_\rho - \left(1-\frac{1}{k}\right)f_{k-1}^*, g\right\rangle_\rho \tag{31}$$

and

$$\mathcal{D}_n' := \left\{g_i(x)/\|g_i\|_\rho\right\}_{i=1}^n \bigcup \left\{-g_i(x)/\|g_i\|_\rho\right\}_{i=1}^n \tag{32}$$

with $g_i \in \mathcal{D}_n$.

Let $f_{\mathbf{z}}^\delta$ and f_k^* be defined as in Algorithm 1 and (30), respectively, then we have

$$\begin{aligned}\mathcal{E}\left(\pi_M f_{\mathbf{z}}^\delta\right) - \mathcal{E}\left(f_\rho\right) \le{}& \mathcal{E}\left(f_k^*\right) - \mathcal{E}\left(f_\rho\right) + \mathcal{E}_{\mathbf{z}}\left(\pi_M f_{\mathbf{z}}^\delta\right) - \mathcal{E}_{\mathbf{z}}\left(f_k^*\right)\\ &+ \mathcal{E}_{\mathbf{z}}\left(f_k^*\right) - \mathcal{E}\left(f_k^*\right) + \mathcal{E}\left(\pi_M f_{\mathbf{z}}^\delta\right) - \mathcal{E}_{\mathbf{z}}\left(\pi_M f_{\mathbf{z}}^\delta\right)\end{aligned}$$

where $\mathcal{E}_{\mathbf{z}}(f) = (1/m)\sum_{i=1}^m (y_i - f(x_i))^2$.

Upon making the short hand notations

$$\mathcal{D}(k) := \mathcal{E}\left(f_k^*\right) - \mathcal{E}\left(f_\rho\right) \tag{33}$$

$$\mathcal{S}(\mathbf{z},k,\delta) := \mathcal{E}_{\mathbf{z}}\left(f_k^*\right) - \mathcal{E}\left(f_k^*\right) + \mathcal{E}\left(\pi_M f_{\mathbf{z}}^\delta\right) - \mathcal{E}_{\mathbf{z}}\left(\pi_M f_{\mathbf{z}}^\delta\right) \tag{34}$$

and

$$\mathcal{P}(\mathbf{z},k,\delta) := \mathcal{E}_{\mathbf{z}}\left(\pi_M f_{\mathbf{z}}^\delta\right) - \mathcal{E}_{\mathbf{z}}(f_k^*) \tag{35}$$

respectively for the approximation error, the sample error and the hypothesis error, we have

$$\mathcal{E}\left(\pi_M f_{\mathbf{z}}^\delta\right) - \mathcal{E}\left(f_\rho\right) = \mathcal{D}(k) + \mathcal{S}(\mathbf{z},k,\delta) + \mathcal{P}(\mathbf{z},k,\delta). \tag{36}$$

At first, we give an upper bound estimate for $\mathcal{D}(k)$, which can be found in [17, Proposition 1].

Lemma 1: Let f_k^* be defined in (30). If $f_\rho \in \mathcal{L}^r_{1,\mathcal{D}_n}$, then

$$\mathcal{D}(k) \le \mathcal{B}^2\left(k^{-1/2} + n^{-r}\right)^2. \tag{37}$$

To bound the sample and hypothesis errors, we need the following Lemma 2.

Lemma 2: Let $y(x)$ satisfy $y(x_i) = y_i$, and $f_{\mathbf{z}}^\delta$ be defined in Algorithm 1. Then, there are at most

$$C\delta^{-2}\log\frac{1}{\delta} \tag{38}$$

atoms selected to build up the estimator $f_{\mathbf{z}}^\delta$. Furthermore, for any $h \in \mathrm{Span}\{D_n\}$, we have

$$\left\|y - f_{\mathbf{z}}^\delta\right\|_m^2 \le 2\|y-h\|_m^2 + 2\delta^2\|h\|_{\mathcal{L}_1(\mathcal{D}_n)}. \tag{39}$$

Proof: Equation (38) can be found in [25, Th. 4.1]. Now we turn to prove (39). Our termination rule guarantees that either $\max_{g\in\mathcal{D}_n}|\langle r_k, g\rangle_m| \le \delta\|r_k\|_m$ or $\|r_k\| \le \delta\|y\|_m$. In the latter case the required bound follows from:

$$\begin{aligned}\|y\|_m &\le \|y-h\|_m + \|h\|_m \le \delta\left(\|y-h\|_m + \|h\|_m\right)\\ &\le \delta\left(\|f-h\|_m + \|h\|_{\mathcal{L}_1(\mathcal{D}_n)}\right).\end{aligned}$$

Thus, we assume $\max_{g\in\mathcal{D}_n}|\langle r_k, g\rangle_m| \le \delta\|r_k\|_m$ holds. By using

$$\langle y - f_k, f_k\rangle_m = 0 \tag{40}$$

we have

$$\begin{aligned}\|r_k\|_m^2 &= \langle r_k, r_k\rangle_m = \langle r_k, y-h\rangle_m + \langle r_k, h\rangle_m\\ &\le \|y-h\|_m\|r_k\|_m + \langle r_k, h\rangle_m\\ &\le \|y-h\|_m\|r_k\|_m + \|h\|_{\mathcal{L}_1(\mathcal{D}_n)}\max_{g\in\mathcal{D}_n}\langle r_k, g\rangle_m\\ &\le \|y-h\|_m\|r_k\|_m + \|h\|_{\mathcal{L}_1(\mathcal{D}_n)}\delta\|r_k\|_m.\end{aligned}$$

This finishes the proof. ∎

Based on Lemma 2 and the fact $\|f_k^*\|_{\mathcal{L}_1(\mathcal{D}_n)} \le \mathcal{B}$ [17, Lemma 1], we obtain

$$\mathcal{P}(\mathbf{z},k,\delta) \le 2\mathcal{E}_{\mathbf{z}}\left(\pi_M f_{\mathbf{z}}^\delta\right) - \mathcal{E}_{\mathbf{z}}\left(f_k^*\right) \le 2\mathcal{B}\delta^2. \tag{41}$$

Now, we turn to bound the sample error $\mathcal{S}(\mathbf{z},k)$. Upon using the short hand notations

$$S_1(\mathbf{z},k) := \left\{\mathcal{E}_{\mathbf{z}}\left(f_k^*\right) - \mathcal{E}_{\mathbf{z}}\left(f_\rho\right)\right\} - \left\{\mathcal{E}\left(f_k^*\right) - \mathcal{E}\left(f_\rho\right)\right\} \tag{42}$$

and

$$S_2(\mathbf{z},\delta) := \left\{\mathcal{E}\left(\pi_M f_{\mathbf{z}}^\delta\right) - \mathcal{E}\left(f_\rho\right)\right\} - \left\{\mathcal{E}_{\mathbf{z}}\left(\pi_M f_{\mathbf{z}}^\delta\right) - \mathcal{E}_{\mathbf{z}}\left(f_\rho\right)\right\} \tag{43}$$

we write

$$\mathcal{S}(\mathbf{z},k) = \mathcal{S}_1(\mathbf{z},k) + \mathcal{S}_2(\mathbf{z},\delta). \tag{44}$$

It can be found in Lin *et al.* [17, Proposition 2] that for any $0 < t < 1$, with confidence $1 - (t/2)$

$$\mathcal{S}_1(\mathbf{z},k) \le \frac{7\left(3M + \mathcal{B}\log\frac{2}{t}\right)}{3m} + \frac{1}{2}\mathcal{D}(k). \tag{45}$$

Using [27, eqs. (A.10)] with k replaced by $C\delta^{-2}\log(1/\delta)$, we have

$$\mathcal{S}_2(\mathbf{z},\delta) \le \frac{1}{2}\mathcal{E}\left(\pi_M f_{\mathbf{z}}^\delta\right) - \mathcal{E}\left(f_\rho\right) + \log\frac{2}{t}\frac{C\delta^{-2}\log\frac{1}{\delta}\log m}{m} \tag{46}$$

holds with confidence at least $1 - t/2$. Therefore, (36), (37), (41), (45), (46) and (44) yield that

$$\mathcal{E}\left(\pi_M f_{\mathbf{z}}^{\delta}\right) - \mathcal{E}\left(f_\rho\right) \leq C\mathcal{B}^2\left(\left(m\delta^2\right)^{-1}\log m \log\frac{1}{\delta}\log\frac{2}{t} + \delta^2 + n^{-2r}\right)$$

holds with confidence at least $1-t$. This finishes the proof of Theorem 2.

ACKNOWLEDGMENT

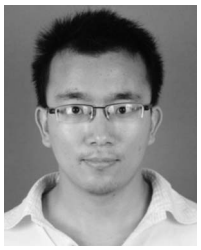
Three anonymous reviewers have carefully read this paper and have provided numerous constructive suggestions. As a result, the overall quality of this paper has been noticeably enhanced, to which the authors feel much indebted and are grateful.

REFERENCES

[1] V. N. Temlyakov, "Greedy approximation," *Acta Numerica*, vol. 17, pp. 235–409, May 2008.
[2] A. Armagan, "Variational ridge regression," in *Proc. Int. Conf. Artif. Intell. Stat.*, 2009, pp. 17–24.
[3] G. H. Golub, M. T. Heath, and G. Wahba, "Generalized cross-validation as a method for choosing a good ridge parameter," *Technometrics*, vol. 21, no. 2, pp. 215–223, Apr. 1979.
[4] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc. B*, vol. 58, no. 1, pp. 267–288, Jan. 1996.
[5] Q. Wu, Y. Ying, and D.-X. Zhou, "Learning rates of least-square regularized regression," *Found. Comput. Math.*, vol. 6, no. 2, pp. 171–192, Apr. 2006.
[6] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Math.*, vol. 57, no. 11, pp. 1413–1457, Nov. 2004.
[7] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, "Iteratively reweighted least squares minimization for sparse recovery," *Commun. Pure Appl. Math.*, vol. 63, no. 1, pp. 1–38, Jan. 2010.
[8] A. R. Barron, A. Cohen, W. Dahmen, and R. A. DeVore, "Approximation and learning by greedy algorithms," *Ann. Stat.*, vol. 36, no. 1, pp. 64–94, Feb. 2008.
[9] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2230–2249, May 2009.
[10] S. Kunis and H. Rauhut, "Random sampling of sparse trigonometric polynomials, II. Orthogonal matching pursuit versus basis pursuit," *Found. Comput. Math.*, vol. 8, no. 6, pp. 737–763, Dec. 2008.
[11] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.
[12] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 58, no. 2, pp. 1094–1121, Feb. 2012.
[13] J. A. Tropp and S. J. Wright, "Computational methods for sparse solution of linear inverse problems," *Proc. IEEE*, vol. 98, no. 6, pp. 948–958, Jun. 2010.
[14] D. L. Donoho, M. Elad, and V. N. Temlyakov, "On Lebesgue-type inequalities for greedy approximation," *J. Approx. Theory*, vol. 147, no. 2, pp. 185–195, Aug. 2007.
[15] V. N. Temlyakov and P. Zheltov, "On performance of greedy algorithms," *J. Approx. Theory*, vol. 163, no. 9, pp. 1134–1145, 2011.
[16] H. Chen, Y. Zhou, Y. T. Tang, L. Li, and Z. Pan, "Convergence rate of the semi-supervised greedy algorithm," *Neural Netw.*, vol. 44, pp. 44–50, Aug. 2013.
[17] S. B. Lin, Y. H. Rong, X. P. Sun, and Z. B. Xu, "Learning capability of relaxed greedy algorithms," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 10, pp. 1598–1608, Oct. 2013.
[18] S. Chen, C. F. N. Cowan, and P. M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 2, no. 2, pp. 302–309, Mar. 1991.
[19] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE Signal Process. Mag.*, vol. 8, no. 4, pp. 14–38, Oct. 1991.
[20] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001.
[21] R. A. DeVore and V. N. Temlyakov, "Some remarks on greedy algorithms," *Adv. Comput. Math.*, vol. 5, no. 1, pp. 173–187, Dec. 1996.
[22] V. Temlyakov, "Weak greedy algorithms," *Adv. Comput. Math.*, vol. 12, nos. 2–3, pp. 213–227, Feb. 2000.
[23] E. Liu and V. N. Temlyakov, "The orthogonal super greedy algorithm and applications in compressed sensing," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 2040–2047, Apr. 2012.
[24] E. D. Livshits, "Rate of convergence of pure greedy algorithms," *Math. Notes*, vol. 76, nos. 3–4, pp. 497–510, Sep. 2004.
[25] V. N. Temlyakov, "Relaxation in greedy approximation," *Constr. Approx.*, vol. 28, no. 1, pp. 1–25, Jun. 2008.
[26] H. Chen, L. Li, and Z. Pan, "Learning rates of multi-kernel regression by orthogonal greedy algorithm," *J. Stat. Plan. Inference*, vol. 143, no. 2, pp. 276–282, Aug. 2012.
[27] C. Xu, S. Lin, J. Fang, and R. Li, "Prediction-based termination rule for greedy learning with massive data," *Statistica Sinica*, vol. 26, no. 2, pp. 841–860, Jan. 2016.
[28] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, Feb. 1970.
[29] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, Mar. 2009.
[30] T. Poggio and C. R. Shelton, "On the mathematical foundations of learning," *Amer. Math. Soc.*, vol. 39, no. 1, pp. 1–49, Oct. 2001.
[31] F. Cucker and D.-X. Zhou, *Learning Theory: An Approximation Theory Viewpoint*, vol. 24. Cambridge, U.K.: Cambridge Univ. Press, Mar. 2007.
[32] T. A. Stamey *et al.*, "Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate. II. Radical prostatectomy treated patients," *J. Urol.*, vol. 141, no. 5, pp. 1076–1083, May 1989.
[33] B. Efron, T. Hastie, I. Johnstone, and R. Tibsirani, "Least angle regression," *Ann. Stat.*, vol. 32, no. 2, pp. 407–451, Apr. 2004.
[34] D. Harrison and D. L. Rubinfeld, "Hedonic prices and the demand for clean air," *J. Environ. Econ. Manag.*, vol. 5, no. 1, pp. 81–102, Mar. 1978.
[35] I.-C. Ye, "Modeling of strength of high-performance concrete using artificial neural networks," *Cement Concrete Res.*, vol. 28, no. 12, pp. 1797–1808, Dec. 1998.
[36] W. J. Nash, T. L. Sellers, S. R. Talbot, A. J. Cawthorn, and W. B. Ford, "The population biology of abalone (Haliotis species) in Tasmania. I. Blacklip abalone (H. Rubra) from the north coast and islands of bass strait," Div. Sea Fisheries, Dept. Primary Ind., Hobart, TAS, Australia, Tech. Rep. 48, Jan. 1994.
[37] A. Jain, K. Nandakumar, and A. Ross, "Score normalization in multimodal biometric systems," *Pattern Recognit.*, vol. 38, no. 12, pp. 2270–2285, Dec. 2005.

Lin Xu received the Ph.D. degree from the Institute for Information and System Sciences, School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, China.

He is currently a Post-Doctoral Associate with the Multimedia and Visual Computing Laboratory, New York University Abu Dhabi, Abu Dhabi, UAE. His current research interests include neural networks, learning algorithms, and applications in computer vision.



Shaobo Lin received the Ph.D. degree in applied mathematics from Xi'an Jiaotong University, Xi'an, China.

He is currently with Wenzhou University, Wenzhou, China. His current research interests include the neural networks and learning theory.

Jinshan Zeng received the Ph.D. degree in applied mathematics from Xi'an Jiaotong University, Xi'an, China.

He is currently an Assistant Professor with the College of Computer Information Engineering, Jiangxi Normal University, Nanchang, China.

Xia Liu received the Ph.D. degree in applied mathematics from Xi'an Jiaotong University, Xi'an, China.

She is currently an Assistant Professor with the School of Sciences, Xi'an University of Technology, Xi'an.

Yi Fang received the Ph.D. degree in computer graphics and vision from Purdue University, West Lafayette, IN, USA.

Upon one year industry experience, as a research intern with Siemens, Princeton, NJ, USA, and a Senior Research Scientist in Riverain Technologies, Dayton, OH, USA, and a half-year academic experience as a Senior Staff Scientist with the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN, USA. He joined New York University Abu Dhabi, Abu Dhabi, UAE, as an Assistant Professor of Electrical and Computer Engineering. He is currently researching on the development of state-of-the-art techniques in large-scale visual computing, deep visual learning, deep cross-domain, and cross-modality model and their applications in engineering, social science, medicine, and biology.

Zongben Xu received the Ph.D. degree in mathematics from Xi'an Jiaotong University, Xi'an, China, in 1987.

He currently serves as a Chief Scientist of "The Basic Theory and Key Technology of Intellisense for Unstructured Environment," the National Basic Research Program of China (973 Project), and the Director of the Institute for Information and System Sciences, Xi'an Jiaotong University. His current research interests include applied mathematics, intelligent information processing, and data science and technology.

Dr. Xu was a recipient of the National Natural Science Award of China in 2007 and the CSIAM Su Buchin Applied Mathematics Prize in 2008. He delivered a 45-min sectional talk at International Congress of Mathematicians in 2010. He is a member of the Chinese Academy of Sciences.