Groupwise Retargeted Least-Squares Regression

Lingfeng Wang and Chunhong Pan

*Abstract***—In this brief, we propose a new groupwise retargeted least squares regression (GReLSR) model for multicategory classification. The main motivation behind GReLSR is to utilize an additional regularization to restrict the translation values of ReLSR, so that they should be similar within same class. By analyzing the regression targets of ReLSR, we propose a new formulation of ReLSR, where the translation values are expressed explicitly. On the basis of the new formulation, discriminative least-squares regression can be regarded as a special case of ReLSR with zero translation values. Moreover, a groupwise constraint is added to ReLSR to form the new GReLSR model. Extensive experiments on various machine leaning data sets illustrate that our method outperforms the current state-of-the-art approaches.**

***Index Terms*— Groupwise, least-squares regression (LSR), multicategory classification, retargeted least-squares regression (ReLSR).**

I. Introduction

Least-squares regression (LSR) has been widely applied in many machine learning tasks, such as manifold learning, discriminative learning, semisupervised learning [1], feature selection [2], artificial neural networks (ANNs) training [3], and so on. In the past decades, researchers have paid more attention to LSR, and the proposed many variants, including “kernel” ridge regression [4], weighted LSR [5], LASSO regression [6], a least-squares support vector machine (SVM) [7], discriminative least-squares regression (DLSR) [8], and retargeted LSR (ReLSR) [9]. These variants have also had a profound influence on machine learning.

We have n training samples $\{\mathbf{x}_i, y_i\}_{i=1}^n$, where $\mathbf{x}_i \in \mathbb{R}^{d\times 1}$ is a data point and $y_i \in \{1, 2, \ldots, c\}$ is the corresponding label of $\mathbf{x}_i$ (c is the number of classes). Letting $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]^\top \in \mathbb{R}^{n\times d}$ be the data matrix, the purpose of LSR and its variants is to learn a regression matrix $\mathbf{W} \in \mathbb{R}^{d\times c}$ and an offset vector $\mathbf{b} \in \mathbb{R}^{c\times 1}$ such that a well-defined target matrix $\mathbf{T} \in \mathbb{R}^{n\times c}$ can be expressed approximately as

$$\mathbf{X}\mathbf{W} + \mathbf{e}_n\mathbf{b}^\top \approx \mathbf{T} \tag{1}$$

where $\mathbf{e}_n = [1, 1, \ldots, 1]^\top \in \mathbb{R}^{n\times 1}$ is a vector with all 1s.

LSR: LSR adopts $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n]^\top \in \mathbb{R}^{n\times c}$ as the regression target, where $\mathbf{y}_i$ is a label vector with -1 or 1 for the ith data sample. For example, if the ith sample belongs to the jth class, its label is

$$\mathbf{y}_i = [-1, \ldots, -1, 1, -1, \ldots, -1]$$

with only the jth element being equal to 1. The objective function of LSR is defined as

$$\min_{\mathbf{W},\mathbf{b}} \|\mathbf{X}\mathbf{W} + \mathbf{e}_n\mathbf{b}^\top - \mathbf{Y}\|_F^2 + \lambda\|\mathbf{W}\|_F^2 \tag{2}$$

where λ is a regularization parameter.

DLSR: One limitation of LSR is that the regression target needs to be 1 or -1, which is inappropriate for classification. To solve this problem, Xiang *et al.* [8] proposed a DLSR model. In DLSR, the regression target is

$$\mathbf{Y} + \mathbf{Y} \odot \mathbf{U}, \quad \text{s.t. } \mathbf{U} \succcurlyeq 0 \tag{3}$$

where $\mathbf{U} \in \mathbb{R}^{n\times c}$ is a nonnegative matrix and $\odot$ is an elementwise product. The constraint $\mathbf{U} \succcurlyeq 0$ means that each element of the matrix $\mathbf{U}$ is greater than or equal to zero, namely, $\mathbf{U}_{i,j} \geq 0, \forall i, j$. The objective function of DLSR is

$$\begin{aligned}&\min_{\mathbf{W},\mathbf{b},\mathbf{U}} \|\mathbf{X}\mathbf{W} + \mathbf{e}_n\mathbf{b}^\top - \mathbf{Y} - \mathbf{Y}\odot\mathbf{U}\|_F^2 + \lambda\|\mathbf{W}\|_F^2\\ &\text{s.t. } \mathbf{U} \succcurlyeq 0.\end{aligned} \tag{4}$$

As interpreted in [8], the core idea behind DLSR is to force the regression targets of different classes to move along opposite directions by introducing a technique called ε-dragging represented by non-negative dragging matrix $\mathbf{U}$. The dragging matrix $\mathbf{U}$ should be optimized in the learning process.

ReLSR: Motivated by [8], Zhang *et al.* [9] proposed ReLSR, which learns regression targets from input data. The objective function of ReLSR is

$$\begin{aligned}&\min_{\mathbf{W},\mathbf{b},\mathbf{T}} \|\mathbf{X}\mathbf{W} + \mathbf{e}_n\mathbf{b}^\top - \mathbf{T}\|_F^2 + \lambda\|\mathbf{W}\|_F^2\\ &\text{s.t. } \mathbf{T}_{i,y_i} - \max_{j\neq y_i}\mathbf{T}_{i,j} \geq 2, \quad i = 1, 2, \ldots, n.\end{aligned} \tag{5}$$

From (5), the regression target in ReLSR is

$$\mathbf{T}_{i,y_i} - \max_{j\neq y_i}\mathbf{T}_{i,j} \geq 2, \quad i = 1, 2, \ldots, n. \tag{6}$$

The target matrix $\mathbf{T}$ is proposed to guarantee that each sample is correctly classified with the large margin. Based on the target matrix $\mathbf{T}$, the margin between the targets of true and false classes should be larger than 2.

In this brief, a new groupwise ReLSR (GReLSR) model is proposed for multicategory classification. The new formulation of the ReLSR model is first proposed. The main superiority of the new model is that the translation values can be explicitly expressed. Then, the groupwise regularization is proposed to restrict the translation values within the same class to be similar. Specifically, the contributions and the details are highlighted as follows.

1) With the new formulation of the ReLSR model, we prove that DLSR is a special case of ReLSR with the translation values being zeros. Furthermore, it is feasible to introduce new constraints to restrict regression target, for example, the groupwise regularization proposed in this brief.
2) Owing to the introduction of groupwise regularization, the flexibility of regression targets in GReLSR falls between DLSR and ReLSR.
3) A new optimization method is proposed to solve the GReLSR model. Note that, the proposed optimization method can also be used to solve the ReLSR model under the new formulation. The new optimization method is faster than the optimization method proposed in [9].

Manuscript received June 20, 2016; revised December 26, 2016; accepted January 7, 2017. This work was supported by the National Natural Science Foundation of China under Grant 61403376, Grant 91338202, Grant 91646207, and Grant 91438105.

The authors are with the Department of National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: lfwang@nlpr.ia.ac.cn; chpan@nlpr.ia.ac.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/TNNLS.2017.2651169

The remainder of this brief is organized as follows. First, previous work is briefly introduced in Section II. The new ReLSR model and the relationship between DLSR and ReLSR are described in Section III. The proposed GReLSR model and the corresponding optimization algorithm are presented in Section IV. Experimental results are provided in Section V and some concluding remarks are given in Section VI.

II. Previous Work

LSR has attracted more attention in the past decades, and many variants have been proposed. One major limitation of LSR is that the regression target needs to be 1 or -1, which is inappropriate for classification [10], [11]. To solve this problem, many new models have been proposed, which can be mainly divided into two groups, namely, loss function improvement, and soft label learning.

The loss function improvement methods have been proposed by introducing new surrogate losses to replace the least-squares loss used in LSR [12]. For example, the classical SVM[1] utilizes the hinge loss [13] or squared hinge loss [14], while the logistic regression [16] adopts the logistic loss. A large number of loss functions (or regression targets) have been proposed in the robust regression research area, for example, Huber loss [17], Tukey loss, and so on. Huber loss [17] is widely used in robust regression. It is quadratic for small values and linear for large values.

The soft label learning methods have been proposed to learn new soft labels to replace the hard labels. These methods can preserve the least-squares loss with new learned soft labels; as a result, closed-form solutions for regression parameters can be maintained as LSR. In [18], a new stagewise least-squares model is proposed in which labels are stagewisely updated according to the regression errors. Xiang *et al.* [8] proposed a DLSR model for multicategory classification tasks, in which a technique called ε-dragging is introduced to force the regression targets of different classes to move along opposite directions. Based on analysis of the loss functions, Wang *et al.* [19] proved that DLSR is a relaxation of L_2-SVM, and proposed margin scalable DLSR, which can explicitly control the margin as well as the number of support vectors of DLSR. Zhang *et al.* [9] proposed an ReLSR model that learns the regression targets from input data. As discussed in [9], the regression target of ReLSR is more accurate than LSR and DLSR in measuring the classification performance. However, they did not provide enough theoretical analyses of the regression target of ReLSR as well as of the relationship to DLSR, making it difficult to understand the ReLSR model comprehensively.

III. From DLSR to ReLSR

Before discussing the relationship between DLSR and ReLSR, we reformulated the regression target of ReLSR as

$$\mathbf{T}_{i,y_i} - \mathbf{T}_{i,j} \geq 2, \quad j \neq y_i, \ i = 1, 2, \ldots, n. \tag{7}$$

We also defined a new regression target

$$\mathbf{Y} + \mathbf{Y} \odot \mathbf{U} + \mathbf{a}\mathbf{e}_c^\top, \quad \text{s.t. } \mathbf{U} \succcurlyeq 0 \tag{8}$$

where $\mathbf{a} \in \mathbb{R}^{n\times 1}$ is an offset vector and $\mathbf{e}_c = [1, 1, \ldots, 1]^\top \in \mathbb{R}^{c\times 1}$. We let the regression target set of ReLSR be

$$\mathcal{A} = \{\mathbf{T} | \mathbf{T}_{i,y_i} - \mathbf{T}_{i,j} \geq 2, \quad j \neq y_i, \ i = 1, 2, \ldots, n\} \tag{9}$$

and let the new regression target set of (8) be

$$\mathcal{B} = \left\{\mathbf{Y} + \mathbf{Y} \odot \mathbf{U} + \mathbf{a}\mathbf{e}_c^\top | \mathbf{U} \succcurlyeq 0\right\}. \tag{10}$$

A new formulation of ReLSR is proposed based on $\mathcal{B}$.

A. New Formulation of ReLSR

A theorem was proposed to interpret the relationship between $\mathcal{A}$ and $\mathcal{B}$ before proposing a new formulation of the ReLSR model.

Theorem 1: Two sets $\mathcal{A}$ and $\mathcal{B}$ are equal, namely, $\mathcal{A} = \mathcal{B}$.

Proof: To prove $\mathcal{A} = \mathcal{B}$, we only need to prove following two aspects, i.e., $\mathcal{A} \subseteq \mathcal{B}$ and $\mathcal{B} \subseteq \mathcal{A}$.

$\underline{\mathcal{A} \subseteq \mathcal{B}}$: Let $\mathbf{T}$ be any item in the set $\mathcal{A}$, namely, $\mathbf{T} \in \mathcal{A}$. The ith row $\mathbf{T}_{i,*}$ can be reformulated as

$$\mathbf{T}_{i,*} = \mathbf{Y}_{i,*} + \left(\mathbf{T}_{i,*} - \mathbf{Y}_{i,*} - a_i\mathbf{e}_c^\top\right) + a_i\mathbf{e}_c^\top \tag{11}$$

where $a_i = \mathbf{T}_{i,y_i} - 1$. Denoting a matrix $\mathbf{U}$, its ith row is

$$\mathbf{U}_{i,*} = \begin{cases} 0, & j = y_i \\ \mathbf{T}_{i,y_i} - \mathbf{T}_{i,j} - 2, & j \neq y_i. \end{cases} \tag{12}$$

By substituting (12) into (11), we find that

$$\mathbf{T}_{i,*} = \mathbf{Y}_{i,*} + \mathbf{Y}_{i,*} \odot \mathbf{U}_{i,*} + a_i\mathbf{e}_c^\top.$$

By considering all rows of $\mathbf{T}$, the item

$$\mathbf{T} = \mathbf{Y} + \mathbf{Y} \odot \mathbf{U} + \mathbf{a}\mathbf{e}_c^\top, \quad \mathbf{U} \succcurlyeq 0.$$

Therefore, the item $\mathbf{T}$ belongs to the set $\mathcal{B}$, namely, $\mathbf{T} \in \mathcal{B}$. Accordingly, we can conclude that $\mathcal{A} \subseteq \mathcal{B}$.

$\underline{\mathcal{B} \subseteq \mathcal{A}}$: For any item $\mathbf{T} \in \mathcal{B}$, that is

$$\mathbf{T} = \mathbf{Y} + \mathbf{Y} \odot \mathbf{U} + \mathbf{a}\mathbf{e}_c^\top, \quad \text{s.t. } \mathbf{U} \succcurlyeq 0$$

we can find that

$$\begin{aligned}\mathbf{T}_{i,y_i} - \mathbf{T}_{i,j} &= 2 + \mathbf{Y}_{i,y_i}\mathbf{U}_{i,y_i} - \mathbf{Y}_{i,j}\mathbf{U}_{i,j} \geq 2 \\ &\quad \text{and } \forall j \neq y_i \ i = 1, 2, \ldots, n.\end{aligned}$$

Therefore, the item $\mathbf{T}$ belongs to the set $\mathcal{A}$, namely, $\mathbf{T} \in \mathcal{A}$. Accordingly, we can conclude that $\mathcal{B} \subseteq \mathcal{A}$. ■

From Theorem 1, we can see that $\mathbf{U}_{i,y_i}$ is equal to 0 (for any row of i). Hence, the set $\mathcal{B}$ can be reformulated as

$$\mathcal{B} = \left\{\mathbf{Y} + \mathbf{Y} \odot \mathbf{U} + \mathbf{a}\mathbf{e}_c^\top \middle| \mathbf{U} \succcurlyeq 0, \ \{\mathbf{U}_{i,y_i} = 0\}_{i=1}^n\right\}.$$

Correspondingly, the regression target

$$\mathbf{Y} + \mathbf{Y} \odot \mathbf{U} + \mathbf{a}\mathbf{e}_c^\top, \quad \text{s.t. } \mathbf{U} \succcurlyeq 0, \ \{\mathbf{U}_{i,y_i} = 0\}_{i=1}^n. \tag{13}$$

By considering the new target described in (13), the ReLSR model can be reformulated as

$$\begin{aligned}\min_{\mathbf{W},\mathbf{b},\mathbf{U},\mathbf{a}} \ & \|\mathbf{X}\mathbf{W} + \mathbf{e}_n\mathbf{b}^\top - \mathbf{Y} - \mathbf{Y} \odot \mathbf{U} - \mathbf{a}\mathbf{e}_c^\top\|_F^2 + \lambda\|\mathbf{W}\|_F^2 \\ \text{s.t. } & \mathbf{U} \succcurlyeq 0, \quad \{\mathbf{U}_{i,y_i} = 0\}_{i=1}^n.\end{aligned} \tag{14}$$

In (14), the constraint is represented with the dragging matrix $\mathbf{U}$. Therefore, it is easier to interpret the relationship between DLSR and ReLSR. Specifically, DLSR is a special case of ReLSR with the translation values being zeros (see Section III-B). Furthermore, it is feasible to introduce new constraints to restrict the translation values (as well as the regression target). In this brief, we propose that the groupwise regularization restrict the translation values, so that a new regression target is obtained implicitly (see Section IV).

B. Regression Target Analysis

The target matrix in LSR is $\mathbf{Y}$, in which the false class is -1 and the true class is 1. When the dragging matrix $\mathbf{U}$ is the zero matrix, the target matrix $\mathbf{Y} + \mathbf{Y} \odot \mathbf{U}$ becomes $\mathbf{Y}$. Hence, we can find that $\mathbf{Y} \subset \mathbf{Y} + \mathbf{Y} \odot \mathbf{U}$ (the constraint on $\mathbf{U}$ is ignored for the convenience of description). The main difference between DLSR [see (3)] and ReLSR [see (13)] is that ReLSR introduces a matrix $\mathbf{a}\mathbf{e}_c^\top$. Each row, e.g., the ith row, of $\mathbf{a}\mathbf{e}_c^\top$ is a constant vector, namely,

[1] The SVM proposed in [13] is L_1-SVM, in [14] is L_2-SVM, and in [15] is multiple rank multilinear SVM.

$a_i\mathbf{e}_c^\mathsf{T}$, which explicitly represents a translation value a_i of all class labels. Therefore, it is easy to find that $\mathbf{Y}+\mathbf{Y}\odot\mathbf{U} \subset \mathbf{Y}+\mathbf{Y}\odot\mathbf{U}+\mathbf{a}\mathbf{e}_c^\mathsf{T}$. To summarize, we obtain

$$\mathbf{Y} \subset \mathbf{Y}+\mathbf{Y}\odot\mathbf{U} \subset \mathbf{Y}+\mathbf{Y}\odot\mathbf{U}+\mathbf{a}\mathbf{e}_c^\mathsf{T}. \tag{15}$$

From the preceding equation, we see that when the translation values of all class labels are zeros, the regression target of ReLSR degrades into DLSR. Hence, we conclude that DLSR is a special case of ReLSR with the translation values being zeros.

IV. Groupwise ReLSR Model

The regression target of ReLSR is made more flexible than that of DLSR by introducing the translation matrix $\mathbf{a}\mathbf{e}_c^\mathsf{T}$. However, the translation values $\{a_i\}_{i=1}^n$ of all samples (each row corresponds to a sample) are considered independently. As a result, for samples of the same class, their regression targets can be significantly different. One potential solution is to add constraints on the translation vector $\mathbf{a}$ (or the translation values $\{a_i\}_{i=1}^n$). In this brief, we propose a new groupwise constraint, in which the samples in same class should have similar translation values. Before introducing the groupwise constraint, we first present a generalized regularized ReLSR model, given by

$$\begin{aligned}\min_{\mathbf{W},\mathbf{b},\mathbf{U},\mathbf{a}} \quad & \|\mathbf{X}\mathbf{W}+\mathbf{e}_n\mathbf{b}^\mathsf{T}-\mathbf{Y}-\mathbf{Y}\odot\mathbf{U}-\mathbf{a}\mathbf{e}_c^\mathsf{T}\|_F^2 \\ & +\ \lambda\|\mathbf{W}\|_F^2+\gamma\, R(\mathbf{a}) \\ \text{s.t. } & \mathbf{U}\succcurlyeq 0,\quad \{\mathbf{U}_{i,y_i}=0\}_{i=1}^n \end{aligned}\tag{16}$$

where $R(\mathbf{a})$ is a regularization and γ is a positive constant.

In this brief, the specified groupwise regularization $R(\mathbf{a})$ is proposed, which is defined as

$$R(\mathbf{a})=\sum_{j=1}^{c}\sum_{i\in\mathcal{S}_j}(a_i-\mu_j)^2 \tag{17}$$

where $\mathcal{S}_j$ collects the indexes of the samples belonging to the jth class, and $\Pi=\{\mu_j\}_{j=1}^c$ represents all parameters.

To facilitate the description, the groupwise regularization in (17) is simplified as $R(\mathbf{a},\Pi)$ with the parameter Π. Combining it into the generalized regularized ReLSR, the GReLSR model is

$$\begin{aligned}\min_{\mathbf{W},\mathbf{b},\mathbf{U},\mathbf{a},\Pi} \quad & \|\mathbf{X}\mathbf{W}+\mathbf{e}_n\mathbf{b}^\mathsf{T}-\mathbf{Y}-\mathbf{Y}\odot\mathbf{U}-\mathbf{a}\mathbf{e}_c^\mathsf{T}\|_F^2 \\ & +\lambda\|\mathbf{W}\|_F^2+\gamma\, R(\mathbf{a},\Pi) \\ \text{s.t. } & \mathbf{U}\succcurlyeq 0,\quad \{\mathbf{U}_{i,y_i}=0\}_{i=1}^n. \end{aligned}\tag{18}$$

A. Analysis of Groupwise Regularization

The core idea behind the proposed groupwise regularization is to ensure that the samples within the same class share the same translation values, e.g., the samples in the jth class should be close to the cluster μ_j. Therefore, the regression targets in the same class are similar to each other.

Generally, DLSR model can be regarded as a special case of ReLSR, in which the zero translation vector $\mathbf{0}$ is used, namely, $\mathbf{a}=\mathbf{0}$. With the zero translation vector, the translation values are the same as each other, which can be treated as an enhanced hard version of the groupwise regularization. Therefore, the flexibility of regression targets in GReLSR falls between those in DLSR and ReLSR.

B. Optimization of GReLSR

The objective function in (18) is jointly convex with respect to all variables. Thus, the alternating optimization method [20] is adopted to solve (18) in the following three steps.

Step 1: Given $\mathbf{U}$, $\mathbf{a}$, and Π, the optimal $\mathbf{W}$ and $\mathbf{b}$ are calculated by

$$\mathbf{W}=(\mathbf{X}^\mathsf{T}\mathbf{H}\mathbf{X}+\lambda\mathbf{I}_d)^{-1}\mathbf{X}^\mathsf{T}\mathbf{H}\mathbf{T} \tag{19}$$

and

$$\mathbf{b}=\frac{\mathbf{T}^\mathsf{T}\mathbf{e}_n-\mathbf{W}^\mathsf{T}\mathbf{X}^\mathsf{T}\mathbf{e}_n}{n} \tag{20}$$

where $\mathbf{T}=\mathbf{Y}+\mathbf{Y}\odot\mathbf{U}+\mathbf{a}\mathbf{e}_c^\mathsf{T}$ is an estimated label matrix, $\mathbf{I}_d\in\mathbb{R}^{d\times d}$ is an identity matrix, and $\mathbf{H}=\mathbf{I}_n-\frac{1}{n}\mathbf{e}_n\mathbf{e}_n^\mathsf{T}$.

Step 2: Given $\mathbf{W}$, $\mathbf{b}$, and Π, the optimal $\mathbf{U}$ and $\mathbf{a}$ are

$$\begin{aligned}\min_{\mathbf{U},\mathbf{a}} \quad & \|\mathbf{R}-\mathbf{Y}\odot\mathbf{U}-\mathbf{a}\mathbf{e}_c^\mathsf{T}\|_F^2+\gamma\, R(\mathbf{a},\Pi) \\ \text{s.t. } & \mathbf{U}\succcurlyeq 0,\quad \{\mathbf{U}_{i,y_i}=0\}_{i=1}^n \end{aligned}\tag{21}$$

where $\mathbf{R}=\mathbf{X}\mathbf{W}+\mathbf{e}_n\mathbf{b}^\mathsf{T}-\mathbf{Y}$ is a regression error matrix. Equation (21) can be decomposed into n independent subproblems. The ith subproblem is the learning of the ith rows of $\mathbf{U}$ and $\mathbf{a}$. Let the ith rows of $\mathbf{R}$, $\mathbf{Y}$, $\mathbf{U}$, and $\mathbf{a}$ be $\mathbf{r}$, $\mathbf{y}$, $\mathbf{u}$, and a, and the ith sample belongs to the kth class, which indicates that the cluster should be μ_k. Hence, the ith subproblem is

$$\begin{aligned}\min_{\mathbf{u},a} \quad & \sum_{j=1,j\neq y_i}^{c}(r_j+u_j-a)^2+(a-r_{y_i})^2+\gamma\,(a-\mu_k)^2 \\ \text{s.t. } & u_j\geq 0,\quad j\neq y_i \end{aligned}\tag{22}$$

where r_i is the ith element of $\mathbf{r}$ (same for y_i). A new optimization method is proposed in Section IV-C.

Step 3: Given $\mathbf{W}$, $\mathbf{b}$, $\mathbf{U}$, and $\mathbf{a}$, the optimal $\Pi=\{\mu_j\}_{j=1}^c$ is calculated by

$$\mu_j=\frac{\sum_{i\in\mathcal{S}_j}a_i}{\mathrm{Card}(\mathcal{S}_j)},\quad j=1,2,\ldots,c \tag{23}$$

where $\mathrm{Card}(\mathcal{S}_j)$ is the size of the set $\mathcal{S}_j$.

The dragging matrix $\mathbf{U}$, translation vector $\mathbf{a}$, and groupwise (or cluster) parameter Π are all set to be zeros in the initialization. Iterating the above three steps, we can obtain the optimal values of the regression parameters $\mathbf{W}^\star$ and $\mathbf{b}^\star$ (as well as the optimal values of the dragging matrix $\mathbf{U}^\star$, translation vector $\mathbf{a}^\star$, and groupwise parameter $\Pi^\star$). Based on $\mathbf{W}^\star$ and $\mathbf{b}^\star$, each test sample $\mathbf{x}$ is classified by

$$\arg\max_{j=1,2,\ldots,c}\ \mathbf{W}_{*j}^{\star\mathsf{T}}\mathbf{x}+b_j^\star$$

where $\mathbf{W}_{*i}^\star$ is the jth column of $\mathbf{W}^\star$.

C. Optimization of (22)

Before introducing the optimization algorithm of (22), we first give the following lemma.

Lemma 1: The closed-form solution of

$$\min_u\ (u-z)^2,\quad \text{s.t. } u\geq 0$$

is $u=\begin{cases} z, & z>0 \\ 0, & z\leq 0 \end{cases}$.

Fixing the translation value a, the dragging values $\{u_j\}_{j=1,j\neq y_i}^c$ can be calculated according to Lemma 1. Assuming that the regression errors $\{r_j\}_{j=1,j\neq y_i}^c$ are arranged in the descending order, namely

$$r_1\geq r_2\geq\cdots\geq r_c$$

and the corresponding solution of $\{u_j\}_{j=1,j\neq y_i}^{c}$ is

$$\begin{aligned} u_1 &= 0 \\ &\cdots \\ u_l &= 0 \\ u_{l+1} &= a - r_{l+1} \\ &\cdots \\ u_c &= a - r_c \end{aligned}$$

in which $a - r_l \leq 0$ and $a - r_{l+1} > 0$. In such a case, the cost of the objective function of (22) is

$$\sum_{j=1}^{l}(r_j - a)^2 + (a - r_{y_i})^2 + \gamma\,(a - \mu_k)^2. \tag{24}$$

As shown in (24), the cost is proportional to the number of zeros in $\{u_j\}_{j=1,j\neq y_i}^{c}$. To decrease the number of zeros, the translation value a should be large. If $a \geq r_1$, all items are not zeros. However, with a large translation value a, the cost of $(a - r_{y_i})^2 + \gamma\,(a - \mu_k)^2$ may be large. Accordingly, the translation value a can start from c_1.

Fixing $\{u_j\}_{j=1}^{m-1}$ ($m-1$ is the number of zeros, and $\{u_j = 0\}_{j=1}^{m-1}$), the solution of a is

$$a = \frac{\sum_{j=1}^{m-1} r_j + \alpha}{m - 1 + \beta} \tag{25}$$

where $\alpha = r_{y_i} + \gamma\,\mu_k$ and $\beta = 1 + \gamma$. Supposing the optimal solution of $a > r_{m+1}$, the solution of $\{u_j\}_{j=1,j\neq y_i}^{c}$ should be

$$\{u_j = 0\}_{j=1}^{m-1} \text{ and } \{u_j = a - r_j\}_{j=m}^{c}. \tag{26}$$

In other words, the solution represented in (25) and (26) is a convergence solution.

On the basis of the above analysis, the new algorithm described in Algorithm 1 is proposed to optimize (22).

As illustrated in Algorithm 1, the computational complexity of the sort algorithm in Step 2 is $O(c \log c)$, the complexity from Steps 3 to 8 is $O(c)$ (note that the parameter a can be calculated incrementally), and the complexity of Step 9 is $O(c)$. Hence, the complexity of Algorithm 1 is $O(c \log c)$. Our algorithm can be easily and efficiently parallelized.

By setting $\gamma = 0$, Algorithm 1 can also be used to solve the ReLSR model with the same complexity. The optimization of (22) is the same as the *retargeting* algorithm in [9]. The complexity of the *retargeting* algorithm is $O(c^2)$. Therefore, the new optimization method, which is derived from the new formulation of the ReLSR model, is faster.

D. Convergence Analysis of GReLSR

In Section IV-B, we proposed an iterative method of solving the GReLSR model. To analyze the convergence of the alternating optimization algorithm, we first denoted the objective function in (18) as $\mathcal{F}(\mathbf{W}, \mathbf{b}, \mathbf{U}, \mathbf{a}, \Pi)$. We then have the following lemma, showing the convergence of the GReLSR model.

Lemma 2: The alternating optimization algorithm monotonically decreases the value of $\mathcal{F}(\mathbf{W}, \mathbf{b}, \mathbf{U}, \mathbf{a}, \Pi)$.

Proof: Denote the value of the objective function at the tth iteration by $\mathcal{F}(\mathbf{W}^t, \mathbf{b}^t, \mathbf{U}^t, \mathbf{a}^t, \Pi^t)$. During the $(t+1)$th iteration, we first fix $\mathbf{U}^t$, $\mathbf{a}^t$, and Π^t, and solve the subproblem

$$\min_{\mathbf{W},\mathbf{b}} \mathcal{F}(\mathbf{W}, \mathbf{b}, \mathbf{U}^t, \mathbf{a}^t, \Pi^t).$$

Algorithm 1 Optimization of (22)

```
Data: The regression errors $\{r_j\}_{j=1}^{c}$ and the cluster $\mu_k$.
Result: The dragging values $\{u_j\}_{j=1,j\neq y_i}^{c}$ and the translation
        value $a$.
1 $\alpha = r_{y_i} + \gamma\,\mu_k$; $\beta = 1 + \gamma$;
2 Sorting the regression errors $\{r_j\}_{j=1,j\neq y_i}^{c}$ in descending order.
  The assignment of the dragging values is according to the order
  of the regression errors.;
3 for $m = 1 (m \neq y_i)$ to $c$ do
4     $a = \frac{\sum_{j=1}^{m-1} r_j + \alpha}{m-1+\beta}$;
5     if $a \geq r_m$ then
6         Break;
7     end
8 end
9 $\{u_j = 0\}_{j=1}^{m-1}$ and $\{u_j = a - r_j\}_{j=m}^{c}$ $(m \neq y_i)$;
```

The optimal solution is $\mathbf{W}^{t+1}$, $\mathbf{b}^{t+1}$. Since the above problem is convex, we thereby have

$$\mathcal{F}(\mathbf{W}^{t+1}, \mathbf{b}^{t+1}, \mathbf{U}^t, \mathbf{a}^t, \Pi^t) \leq \mathcal{F}(\mathbf{W}^t, \mathbf{b}^t, \mathbf{U}^t, \mathbf{a}^t, \Pi^t). \tag{27}$$

We then fix $\mathbf{W}^{t+1}$, $\mathbf{b}^{t+1}$, and Π^t, and solve the subproblem

$$\min_{\mathbf{U},\mathbf{a}} \mathcal{F}(\mathbf{W}^{t+1}, \mathbf{b}^{t+1}, \mathbf{U}, \mathbf{a}, \Pi^t).$$

The objective function of this subproblem is quadratic. Hence, it is convex with respect to $\mathbf{U}$ and $\mathbf{a}$. It is easy to find that the constraint is also convex. Hence, this problem is a convex problem. With the optimal solution being $\mathbf{U}^{t+1}$, $\mathbf{a}^{t+1}$, we obtain

$$\begin{aligned} &\mathcal{F}(\mathbf{W}^{t+1}, \mathbf{b}^{t+1}, \mathbf{U}^{t+1}, \mathbf{a}^{t+1}, \Pi^t) \\ &\quad \leq \mathcal{F}(\mathbf{W}^{t+1}, \mathbf{b}^{t+1}, \mathbf{U}^t, \mathbf{a}^t, \Pi^t). \end{aligned} \tag{28}$$

Next, we fix $\mathbf{W}^{t+1}$, $\mathbf{b}^{t+1}$, $\mathbf{U}^{t+1}$, and $\mathbf{a}^{t+1}$, and solve the subproblem

$$\min_{\Pi} \mathcal{F}(\mathbf{W}^{t+1}, \mathbf{b}^{t+1}, \mathbf{U}^{t+1}, \mathbf{a}^{t+1}, \Pi).$$

Owing to the convexity of this subproblem, it follows that:

$$\begin{aligned} &\mathcal{F}(\mathbf{W}^{t+1}, \mathbf{b}^{t+1}, \mathbf{U}^{t+1}, \mathbf{a}^{t+1}, \Pi^{t+1}) \\ &\quad \leq \mathcal{F}(\mathbf{W}^{t+1}, \mathbf{b}^{t+1}, \mathbf{U}^{t+1}, \mathbf{a}^{t+1}, \Pi^t). \end{aligned} \tag{29}$$

Combining (27), (28), and (30), we obtain

$$\mathcal{F}(\mathbf{W}^{t+1}, \mathbf{b}^{t+1}, \mathbf{U}^{t+1}, \mathbf{a}^{t+1}, \Pi^{t+1}) \leq \mathcal{F}(\mathbf{W}^t, \mathbf{b}^t, \mathbf{U}^t, \mathbf{a}^t, \Pi^t).$$

This completes the proof. ■

V. Experimental Results

We compare the proposed GReLSR model with the seven benchmark multicategory models, including L_1-SVM [13], [14], L_2-SVM, multiclass SVM (MC-SVM) [21], logistic regression [16], LSR, DLSR [8], and ReLSR [9], on a range of different data sets. It is worth noting that as a special case of the GReLSR model, the ReLSR model is implemented by setting the weighting parameter γ to 0.

A. Parameter Settings

Motivated by [9], the hyperparameter λ for the LSR, DLSR, ReLSR and proposed GReLSR models was set as follows:

$$\lambda = \widehat{\lambda}\frac{\mathrm{tr}(\mathbf{X}^{\mathsf{T}}\mathbf{H}\mathbf{X})}{\mathrm{tr}(\mathbf{I}_d)} = \frac{\widehat{\lambda}}{d}\mathrm{tr}(\mathbf{X}^{\mathsf{T}}\mathbf{H}\mathbf{X})$$

where $\mathrm{tr}(\cdot)$ is a matrix trace operation. Here, fivefold cross validation was used to determine the optimal hyperparameter λ by setting $\widehat{\lambda}$ from

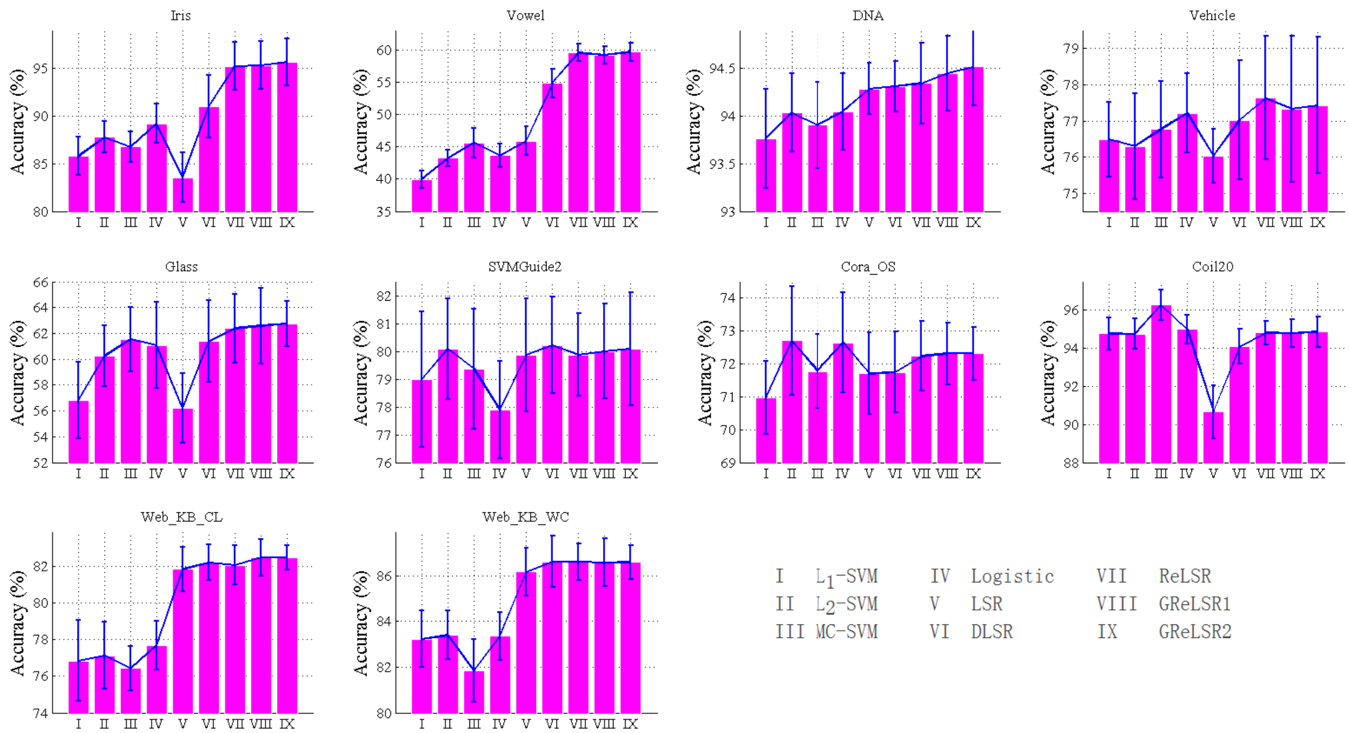

Fig. 1. Classification accuracy and standard deviation of different models on ten machine learning data sets.

the interval of [0 : 0.1 : 1]. In our GReLSR model, two strategies were used to determine the parameter γ, that is, either set to be the constant value 1, or determined by cross validation. The two models and their results are indicated by GReLSR1 and GReLSR2, respectively.

For L_1-SVM, L_2-SVM, MC-SVM, and logistic regression, we used LIBLINEAR[2] to implement them. The major parameter in these methods is the regularization parameter C, which is also determined using the cross-validation technique from the candidate set $\{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$.

B. Results on State-of-the-Art Machine Learning Data Sets

1) Data Sets: Ten machine learning datasets (without normalization) are shown in Table I, including **Iris**, **Vowel**, **DNA**, **Vehicle**, **Glass**, **SVMGuide2**, **Cora-OS** [22], **Coil20** [23] **WebKB-CL**, and **WebKB-WC**, all adopted to evaluate the performance of the GReLSR model. The first six data sets were downloaded from the LIBSVM Web site.[3] Each data set was randomly partition it into two parts, with 40% of the samples selected for training and the rest for testing. Please refer to [19] for the details of the data sets.

2) Results: The comparative results on ten machine learning data sets are shown in Fig. 1. In most cases, our two models, both GReLSR1 and GReLSR2, gave better results than the other approaches, including the ReLSR model. On the basis of cross validation of the weighting parameter γ, the results of GReLSR2 were better than, or at least equal to, those of GReLSR1. Especially with a constant weighting parameter, GReLSR1 also provided higher recognition results on six data sets, including **Iris**, **DNA**, **Glass**, **SVMGuided2**, **Core_OS**, and **Web_KB_CL**, as compared with ReLSR. In most cases, the GReLSR2 results were better than ReLSR. However, in **Vehicle** and **Web_KB_WC**, ReLSR was better. Fortunately, our recognition results were higher than the others except for ReLSR in **Vehicle** and for DLSR and ReLSR in **Web_KB_WC**, which indicates that the groupwise regularization may not break the ReLSR model.

TABLE I

BRIEF DESCRIPTION OF THE DATA SETS. IN THE **CORA-OS**, **WEBKB-CL**, AND **WEBKB-WC** DATA SETS, PCA IS APPLIED TO PROJECT THEM INTO 200-D SUBSPACE

Info. / Data Set	Classes	Features	Total Num.	Train Num.
Iris	3	4	150	60
DNA	3	180	3186	1274
Glass	6	9	214	86
SVMGuide2	3	20	391	156
Vehicle	4	18	846	338
Vowel	11	10	990	396
Cora_OS	4	6737(200)	1246	499
Coil20	20	256	1440	576
WebKB-CL	7	4134(200)	827	331
WebKB-WC	7	4189(200)	1210	484

C. Face Recognition Results

We further evaluated our model on three widely used face recognition data sets, namely, AR [24],[4] CMU-PIE[5] [25] and

[2] Available at www.csie.ntu.edu.tw/~cjlin/liblinear.

[3] Available at: www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/.

[4] For the AR dataset (100 persons), all images were cropped and resized to 165 × 120. For each person, seven images from *Session* 1 were selected as the training images, while the remaining seven images from *Session* 2 were used for testing.

[5] For the CMU-PIE dataset (68 persons), we selected only the frontal face images from all of the face images. The face images were cropped and resized to 64 × 64. For each person, we selected five images for training.

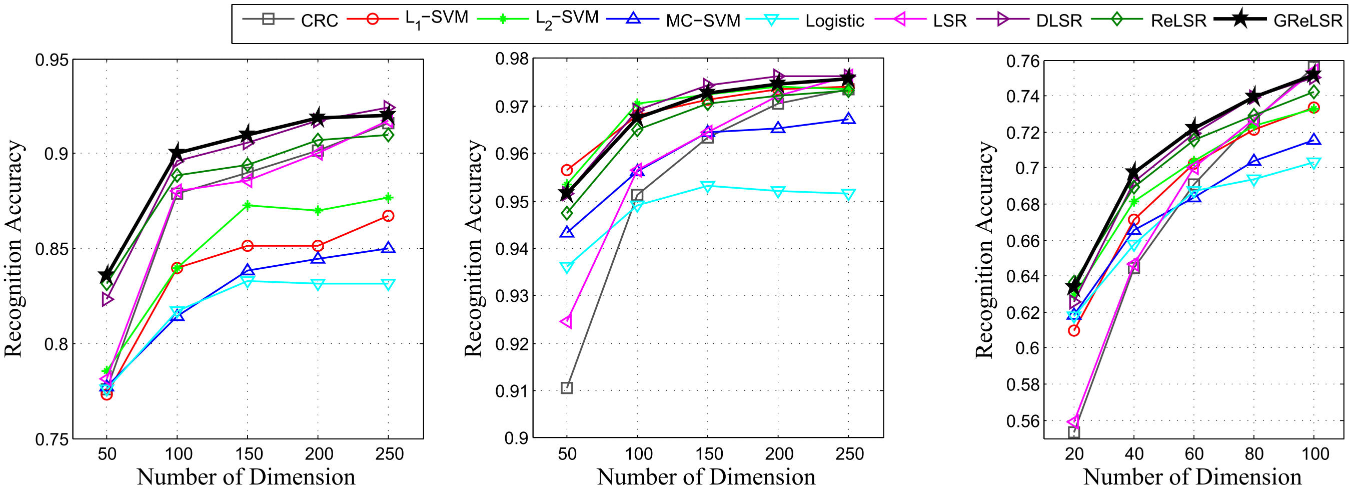

Fig. 2. Comparisons on three widely used face recognition data sets. From left to right: AR, CMU-PIE, and Extended Yale B.

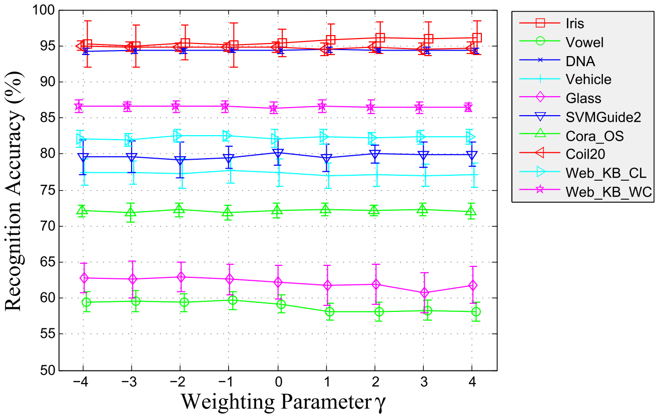

Fig. 3. Recognition accuracies on ten machine learning data sets with different values of the weighting parameter γ. Logarithmic values of the weighting parameters are shown on the x axis. For example, -4 means that the weighting parameter is 10^{-4}.

Extended Yale B[6] [26]. We compared the GReLSR1 model with the above seven benchmark models[7] and one representation-based face recognition method, namely the collaborative representation classification model. For each data set, we utilized principal component analysis (PCA) to reduce the dimension of each image, and selected five different dimensions. The comparative results are shown in Fig. 2. From this figure, we can see that GReLSR1 performed better than the other methods as a whole. In particular, the results of GReLSR1 were better than those of ReLSR.

D. Parameter Evaluation

In this section, we evaluated the proposed GReLSR model with the different weighting parameter setting, and the results are shown in Fig. 3. Here, the weighting parameters were set to $\{10^n\}_{n=-4}^{4}$. The results show that the GReLSR model was stable with the weighting parameter γ, which indicated that introducing groupwise regularization did not break ReLSR model. It is worth noting that, with a specified weighting parameter, GReLSR provided better recognition results than ReLSR on the ten data sets, which partially validates that adding groupwise regularization is necessary. Furthermore, we found that when the parameter $\gamma = 10^{-2}$, 10^{-1} or 1, the classification results of GReLSR were better than ReLSR. Thereby, without using the cross-validation technique, it is suggested that the parameter γ be selected between 10^{-2} and 1.

VI. Conclusion and Discussion

By reformulating the regression target of ReLSR, we conclude that the difference between DLSR and ReLSR is whether or not translation values should be utilized. Unfortunately, ReLSR does not use an additional constraint to restrict translation values. On the basis of this observation, we propose a groupwise constraint, which requires that the translation values within the same class should be similar. By adding the groupwise constraint as a regularization into ReLSR, a new GReLSR model is proposed for multicategory classification tasks. Extensive results testified to the superior performance of GReLSR compared with the other methods.

A. Constraints

The GReLSR model has two main constraints. First, the optimization problem is divided into three subproblems. Although all subproblems are convex and the convergence of optimization can be guaranteed, the convergence speed could be slow, as it relies on the number of iterations. Second, the performance of the GReLSR model depends on the choice of the parameter γ. Cross validation is good technique to determine the parameter γ; however, it is partially dominated by the number of training data.

B. Extensions

The GReLSR model has following potential extensions, which will be our future effects. First, by utilizing the $L_{2,1}$ norm on the regression matrix $\mathbf{W}$, the proposed GReLSR can be adopted for feature selection tasks as [2] and [8]. Second, the proposed groupwise retargeted least-squares error can also be used to train other classifiers that use least-squares error as the optimization criterion, such as ANNs. Third, the groupwise regularization can be replaced by others, such as sparse regularization. Fourth, the kernel methods [27] can be introduced to allow the GReLSR model to process the nonlinear case.

[6] For the Extended Yale B data set (38 persons), we selected 2414 frontal face images. For each person, we selected five images for training.

[7] In this experiment, the parameter λ for LSR, DLSR, ReLSR, and GReLSR1 was uniformly set to a constant value of 0.01.

Acknowledgment

The authors would like thank LetPub for its linguistic assistance during the preparation of this paper.

References

[1] Z. Xu, I. King, M. R.-T. Lyu, and R. Jin, "Discriminative semi-supervised feature selection via manifold regularization," *IEEE Trans. Neural Netw.*, vol. 21, no. 7, pp. 1033–1047, Jul. 2010.

[2] F. Nie, H. Huang, X. Cai, and C. H. Ding, "Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization," in *Proc. NIPS*, 2010, pp. 1813–1821.

[3] C. Chen and X. Yan, "Optimization of a multilayer neural network by using minimal redundancy maximal relevance-partial mutual information clustering with least square regression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 6, pp. 1177–1187, Jun. 2015.

[4] S. An, W. Liu, and S. Venkatesh, "Face recognition using kernel ridge regression," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Minneapolis, MN, USA, Jun. 2007, pp. 1–8.

[5] T. Strutz, *Data Fitting and Uncertainty: A Practical Introduction to Weighted Least Squares and Beyond*. Wiesbaden, Germany: Vieweg, 2010.

[6] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Statist. Soc. B (Methodol.)*, vol. 58, no. 1, pp. 267–288, 1996.

[7] L. Jiao, L. Bo, and L. Wang, "Fast sparse approximation for least squares support vector machine," *IEEE Trans. Neural Netw.*, vol. 18, no. 3, pp. 685–697, May 2007.

[8] S. Xiang, F. Nie, G. Meng, C. Pan, and C. Zhang, "Discriminative least squares regression for multiclass classification and feature selection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 11, pp. 1738–1754, Nov. 2012.

[9] X.-Y. Zhang, L. Wang, S. Xiang, and C.-L. Liu, "Retargeted least squares regression algorithm," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 9, pp. 2206–2213, Sep. 2015.

[10] C. M. Bishop, *Pattern Recognition and Machine Learning* (Information Science and Statistics). Secaucus, NJ, USA: Springer-Verlag, 2006.

[11] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY, USA: Springer-Verlag, 2001.

[12] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe, "Convexity, classification, and risk bounds," *J. Amer. Statist. Assoc.*, vol. 101, no. 473, pp. 138–156, 2006.

[13] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[14] K.-W. Chang, C.-J. Hsieh, and C.-J. Lin, "Coordinate descent method for large-scale L2-loss linear support vector machines," *J. Mach. Learn. Res.*, vol. 9, pp. 1369–1398, Jun. 2008.

[15] C. Hou, F. Nie, C. Zhang, D. Yi, and Y. Wu, "Multiple rank multi-linear SVM for matrix data classification," *Pattern Recognit.*, vol. 47, no. 1, pp. 454–469, 2014.

[16] D. W. Hosmer, Jr., and S. Lemeshow, *Applied Logistic Regression*, 2nd ed. Hoboken, NJ, USA: Wiley, 2000.

[17] P. J. Huber, "Robust estimation of a location parameter," *Ann. Math. Statist.*, vol. 53, no. 1, pp. 73–101, 1964.

[18] S.-H. Yang and B.-G. Hu, "A stagewise least square loss function for classification," in *Proc. SIAM Int. Conf. Data Mining*, Atlanta, GA, USA, Apr. 2008, pp. 120–131.

[19] L. Wang, X.-Y. Zhang, and C. Pan, "MSDLSR: Margin scalable discriminative least squares regression for multicategory classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 12, pp. 2711–2717, Dec. 2016.

[20] P. Tseng, "Further applications of a splitting algorithm to decomposition in variational inequalities and convex programming," *Math. Program.*, vol. 48, no. 1, pp. 249–263, 1990.

[21] K. Crammer and Y. Singer, "On the algorithmic implementation of multiclass kernel-based vector machines," *J. Mach. Learn. Res.*, vol. 2, pp. 265–292, Mar. 2001.

[22] A. K. McCallum, K. Nigam, J. Rennie, and K. Seymore, "Automating the construction of Internet portals with machine learning," *Inf. Retr.*, vol. 3, no. 2, pp. 127–163, 2000.

[23] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (COIL-20)," Dept. Comput. Sci., Columbia Univ., New York, NY, USA, Tech. Rep. CUCS-005-96, 1996.

[24] A. Martinez and R. Benavente, "The AR face database," CVC, Barcelona, Spain, Tech. Rep. 24, Jun. 1998.

[25] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression (PIE) database," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recognit.*, Washington, DC, USA, May 2002, pp. 53–58.

[26] K.-C. Lee, J. Ho, and D. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 684–698, May 2005.

[27] C. Zhang, F. Nie, and S. Xiang, "A general kernelization framework for learning algorithms based on kernel PCA," *Neurocomputing*, vol. 73, nos. 4–6, pp. 959–967, 2010.