Incremental Support Vector Learning for Ordinal Regression

Bin Gu, *Member, IEEE*, Victor S. Sheng, *Member, IEEE*, Keng Yeow Tay, Walter Romano, and Shuo Li

***Abstract*— Support vector ordinal regression (SVOR) is a popular method to tackle ordinal regression problems. However, until now there were no effective algorithms proposed to address incremental SVOR learning due to the complicated formulations of SVOR. Recently, an interesting accurate on-line algorithm was proposed for training ν-support vector classification (ν-SVC), which can handle a quadratic formulation with a pair of equality constraints. In this paper, we first present a modified SVOR formulation based on a sum-of-margins strategy. The formulation has multiple constraints, and each constraint includes a mixture of an equality and an inequality. Then, we extend the accurate on-line ν-SVC algorithm to the modified formulation, and propose an effective incremental SVOR algorithm. The algorithm can handle a quadratic formulation with multiple constraints, where each constraint is constituted of an equality and an inequality. More importantly, it tackles the conflicts between the equality and inequality constraints. We also provide the finite convergence analysis for the algorithm. Numerical experiments on the several benchmark and real-world data sets show that the incremental algorithm can converge to the optimal solution in a finite number of steps, and is faster than the existing batch and incremental SVOR algorithms. Meanwhile, the modified formulation has better accuracy than the existing incremental SVOR algorithm, and is as accurate as the sum-of-margins based formulation of Shashua and Levin.**

***Index Terms*— Incremental learning, online learning, ordinal regression (OR), support vector machine (SVM).**

NOMENCLATURE

To make notations easier to follow, we give a summary of the notations in the following list.

α_i, g_i	The ith element of the vector α and g.
α_c, y_c, j_c	The weight, the label of a candidate sample (x_c, y_c), and the index of the two-class sample set S^j to which (x_c, y_c) belongs.
Δ	The amount of the change of each variable.
$\overline{J}$, J_-, J_+	The complement of the set J, the contracted set of J by deleting j_c, and the enlarged set of J by adding j_c.
d'_{J_-}, E_{J_-}	The subvector of d' by extracting the elements indexed by J_-, and a submatrix of E by extracting the columns indexed by J_-.
$\boxed{d'_j}$, $\boxed{\beta^c_{d'_j}}$	If $j \in J_-$, $\boxed{d'_j}$ and $\boxed{\beta^c_{d'_j}}$ stands for d'_j and $\beta^c_{d'_j}$, respectively. Otherwise, they will be ignored.
$Q_{S_S S_S}$	The submatrix of Q with the rows and columns indexed by S_S.
$\widehat{Q}_{\setminus(d'_{\overline{J'_+}}, M)^2}$	The submatrix of $\widehat{Q}$ after deleting the rows and columns corresponding to $\Delta d'_j$ indexed by $\overline{J'_+}$ and $\Delta\alpha_i$ indexed by M.
$\mathbf{M}^T$	The transpose of the matrix $\mathbf{M}$.
$\mathbf{0}$, O	A zero matrix with proper dimensions, and the $(r-1)\times(r-1)$ matrix with all zeroes except that $O_{j_c j_c} = \varepsilon$.
u_{j_c}, v_{j_c}	A $(r-1)$-dimensional column vector with all zeroes except that the j_cth position is equal to y_c and one, respectively.
$e_{S_S^j}$, $u_{S_S^j}$	A $\lvert S_S\rvert$-dimensional column vector with all zeroes except that the positions corresponding to the samples (x_i, y_i) of S_S^j are equal to -1 and y_i, respectively.

I. INTRODUCTION

IN CONVENTIONAL machine learning and data mining research, predictive learning has become a standard inductive learning, where different subproblem formulations have been identified, for example, classification, metric regression, ordinal regression (OR), and so on. In OR problems, training samples are marked by a set of ranks, which exhibit an ordering among different categories. In contrast to metric regression problems [1], the ranks for OR are of finite types and the metric distances between the ranks are not defined; in comparison with classification problems, these ranks are also different from the labels of multiple classes due to ordering information [8]. Therefore, OR is a special case in predictive learning.

Manuscript received July 8, 2013; revised July 8, 2014; accepted July 16, 2014. This work was supported in part by the Priority Academic Program Development, Jiangsu Higher Education Institutions, in part by the U.S. National Science Foundation under Grant IIS-1115417, and in part by the National Natural Science Foundation of China under Grant 61232016 and Grant 61202137. *(Corresponding author: Bin Gu.)*

B. Gu is with the Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science and Technology, Nanjing 210044, China, School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044, China, and also with the Department of Medical Biophysics, University of Western Ontario, London, ON N6A 3K7, Canada (e-mail: jsgubin@nuist.edu.cn).

V. S. Sheng is with the Department of Computer Science, University of Central Arkansas, Conway, AR 72035 USA (e-mail: ssheng@uca.edu).

K. Y. Tay is with the London Health Science Center, Victoria Hospital, London, ON N6A 5W9, Canada (e-mail: kengyeow.tay@lhsc.on.ca).

W. Romano is with St. Joseph's Health Care, London, ON M6R 1B5, Canada (e-mail: wmromano@rogers.com).

S. Li is with GE HealthCare, London, ON AL9 5EN, Canada, and also with the Department of Medical Biophysics, University of Western Ontario, London, ON N6A 3K7, Canada (e-mail: Shuo.Li@ge.com).

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.2014.2342533

In practical OR tasks, such as information retrieval [15], collaborative filtering [6], flight delays forecasting [22], and so on, training data is usually provided one example at a time. This is a so called online scenario. We use flight delays forecasting as an example. The given flight delay data streams are nonstationary, meaning that data distributions vary over time, and batch algorithms will generally fail if such ambiguous information is present and is erroneously integrated by the batch algorithm. Incremental learning algorithms are more capable in this case, because they allow the incorporation of additional training data without retraining from scratch [18].

Ever since Vapnik's [26] influential work in statistical learning theory, support vector machines (SVMs) [26] have gained profound interest because of good generalization performance [2], [20]. There are also several support vector OR (SVOR) formulations proposed to tackle OR problems. For example, Herbrich *et al*. [15] gave a SVM formulation based on a loss function between pairs of ranks (PSVM). However, the problem size of PSVM is a quadratic function of the training data size. To address this problem, Shashua and Levin [6] proposed two SVM formulations by finding multiple parallel discrimination hyperplanes. One is a fixed-margin-based formulation, and the other is a sum-of-margins-based formulation (SMF). Chu and Keerthi [8] further improved the fixed-margin-based SVOR formulation by explicitly and implicitly keeping ordinal inequalities on the thresholds, in which the explicit constraints-based SVOR was called EXC. Cardoso and Pinto da Costa [23] proposed a data replication method and mapped it into SVM, which also implicitly used the fixed-margin strategy. The problem sizes of these SVOR formulations are all linear in the training data size. In addition, more recently, Seah *et al*. [17] presented a transductive SVM learning paradigm for OR, by taking advantage of the abundance of unlabeled patterns.

Although there exist several perceptron-like algorithms proposed for incremental OR learning (see [3], [9], [10]), very little work has been done on incremental learning for SVOR. Previous works mostly focus on incremental learning for standard SVM, one-class SVM, support vector regression (SVR), and so on. For example, Cauwenberghs and Poggio [7] proposed an exact incremental learning approach (the C&P algorithm) for SVM in 2001. Later, Martin [11] extended it to SVR and proposed an accurate incremental SVR algorithm. Laskov *et al*. [18] implemented an accurate incremental one-class SVM algorithm. Karasuyama and Takeuchi [25] gave an extended algorithm that can handle multiple data samples simultaneously. Recently, Gu *et al*. [28] extended the C&P algorithm to ν-support vector classification (ν-SVC) and proposed an effective accurate on-line ν-SVC algorithm (AONSVM), which can handle the conflict between a pair of equality constraints during the process of incremental learning. Gu and Sheng [29] proved the feasibility and finite convergence of AONSVM under two assumptions (i.e., Assumptions 1 and 2 as mentioned in [29]).

To the best of our knowledge, the PSVM-based incremental algorithm (IPSVM) [12] is the only work on incremental SVOR learning. As mentioned previously, this approach is limited by the size of the problem, which is quadratic in the training data size. Therefore, it is highly desirable to design an effective incremental learning algorithm for the SVOR formulations, whose problem size is linear in the training data size. In this paper, we focus on the SMF of Shashua and Levin [6]. We first present a modified SMF (MSMF), which has multiple constraints of the mixture of an equality and an inequality. Then, we extend AONSVM to MSMF, and propose an effective incremental SVOR algorithm (ISVOR). The incremental algorithm includes two steps, i.e., relaxed adiabatic incremental adjustments (RAIAs), and strict restoration adjustments (SRA). Based on the two steps, the incremental algorithm can handle inequality constraints, and can tackle the conflicts between the equality and inequality constraints. We also provide its finite convergence analysis. Numerical experiments show that ISVOR can converge to the optimal solution in a finite number of steps, and is faster than the existing batch and incremental SVOR algorithms. Meanwhile, the modified formulation has better accuracy than the existing incremental SVOR algorithm, and is as accurate as the SMF of Shashua and Levin [6].

The main contributions of this paper are summarized as follows.

1) We propose an effective ISVOR, whose problem size is linear in the training data size. We also prove the finite convergence of ISVOR. Numerical experiments show that ISVOR is faster than the existing batch and incremental SVOR algorithms.
2) The existing incremental SVM algorithms can handle a quadratic formulation with a pair of equality constraints or an equality constraint for a binary classification problem. The ISVOR can handle a quadratic formulation with multiple constraints of the mixture of an equality and an inequality for multiple binary classification problems. The ISVOR can be viewed as a generalization of the existing incremental SVM algorithms.

The rest of this paper is organized as follows. Section II gives a modified SVOR formulation, (i.e., MSMF), and its Karush–Kuhn–Tucker (KKT) conditions. The incremental SVOR algorithm is presented in Section III. The experimental setup, results and discussion are presented in Sections IV and V. The last section gives some concluding remarks.

II. Modified SVOR Formulation

In this section, we first review SMF, then present MSMF and its dual problem. Finally, we present the KKT conditions for the solution of the dual problem.

A. Review of SMF

Without loss of generality, we consider an OR problem with r ordered categories and denote these categories as consecutive integers $Y = \{1, 2, \ldots, r\}$ to keep the known ordering information. The number of training samples in the jth category ($j \in Y$), is denoted as n^j, and the ith training sample is denoted as x_i^j ($x_i^j \in X$, where X is the input space with $X \subset \mathbb{R}^d$).

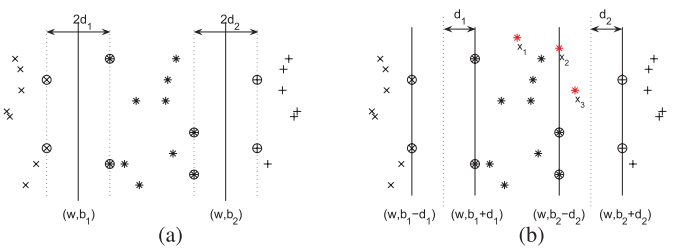


Fig. 1. (a) OR based on the sum-of-margins strategy. (w, b_1) and (w, b_2) are the parallel discrimination hyperplanes obtained from maximizing $2d_1 + 2d_2$, when $\langle w, w\rangle = 1$. Support vectors lie on the boundaries between the neighboring categories. (b) Two cases of incremental SVOR learning. If the added sample (e.g., x_3) is sandwiched between hyperplanes $(w, b_j - d_j)$ and $(w, b_j + d_j)$, adjustments will be needed; otherwise, adjustments will be unnecessary for the added sample, such as x_1 or x_2.

To learn a mapping function $r(\cdot) : X \to Y$, Shashua and Levin [6] considered $r-1$ parallel discrimination hyperplanes, i.e., $\langle w, x\rangle - b_j$ with $b_1 \le \ldots \le b_{r-1}$, where b_j is the threshold of the jth discrimination hyperplane. Supposing $b_r = \infty$, the decision mapping function $r(\cdot)$ can be denoted as

$$r(x) = \min_{j \in Y}\{j : \langle w, x\rangle - b_j < 0\}. \tag{1}$$

Let $d_j \ge 0$ be the shortest distance from the jth discrimination hyperplane to the closest sample in the jth or $(j+1)$th category, which is the margin of the jth discrimination hyperplane (Fig. 1). Based on the sum-of-margins strategy [Fig. 1(a)], Shashua and Levin [6] tried to maximize the sum of all margins $\sum_{j=1}^{r-1} 2d_j$ with $\langle w, w\rangle = 1$, and considered all the sandwiched constraints $b_j + d_j \le b_{j+1} - d_{j+1}$, which derive the following primal problem (i.e., SMF)[1]:

$$\begin{aligned}
\min_{w,b,d,\epsilon,\epsilon^*} \quad & -\sum_{j=1}^{r-1} 2d_j + C\sum_{j=1}^{r-1}\left(\sum_{i=1}^{n^j}\epsilon_i^j + \sum_{i=1}^{n^{j+1}}\epsilon_i^{*j+1}\right) \\
\text{s.t.} \quad & \langle w, \phi(x_i^j)\rangle \le b_j - d_j + \epsilon_i^j, \\
& \epsilon_i^j \ge 0, \quad i = 1, \ldots, n^j, \\
& \langle w, \phi(x_i^{j+1})\rangle \ge b_j + d_j - \epsilon_i^{*j+1}, \\
& \epsilon_i^{*j+1} \ge 0, \quad i = 1, \ldots, n^{j+1}, \\
& \langle w, w\rangle \le 1, \quad b_j + d_j \le b_{j+1} - d_{j+1}, d_j \ge 0
\end{aligned} \tag{2}$$

where $j = 1, \ldots, r-1$, training samples x_i^j are mapped into a high-dimensional reproducing kernel Hilbert space (RKHS) [19] by the transformation function ϕ, and we have the kernel function $K(x_i, x_j) = \langle \phi(x_i), \phi(x_j)\rangle$ with $\langle \cdot, \cdot \rangle$ denoting inner product in RKHS. Furthermore, ϵ_i^j (ϵ_i^{*j+1}) is a non-negative slack variable measuring the degree of misclassification of the data x_i^j (x_i^{j+1}). The parameter C controls the tradeoff between the errors in the training samples and sum-of-margins maximization.

B. MSMF and Its Dual Problem

According to the reduction framework of OR [20], OR learning tries to learn a rank-monotonic mapping function $f(x, j)$, such that $f(x, 1) \ge \cdots \ge f(x, r-1)$. A popular approach to obtain such a function $f(x, j)$ is to use $r-1$ parallel discrimination hyperplanes as mentioned in Section II-A. The key of such an approach is to keep the thresholds b_j ordered. To sum up, there are two kinds of approaches to keep such ordinal thresholds, i.e., the explicit approach [6], [8], and the implicit approach [8], [10], [15], [17], [20], [23]. It should be noted that although Shashua and Levin [6] used an explicit approach, the ordinal thresholds were achieved by the sandwiched constraints as shown in (2) because $b_j \le b_j + d_j \le b_{j+1} - d_{j+1} \le b_{j+1}$. In this paper, we use the popular implicit approach to achieve the ordinal thresholds. Thus, a modified formulation of (2) is used here by discarding the constraint of $b_j + d_j \le b_{j+1} - d_{j+1}$. After discarding the constraint, our proposed OR formulation, (i.e., MSMF) is more favorable to design an incremental SVOR algorithm, because the primal variables b_j and d_j can be induced directly in the KKT conditions (Section II-C).

To present the dual function of the modified formulation in a compact form, we introduce some new notations.

1) Based on the reduction framework of [20], OR can be regarded as $r-1$ binary classification problems. Thus, we define the two-class training sample set $S^j = \{(x_i^j, y_i^j = -1)\}_{i=1}^{n^j} \cup \{(x_i^{j+1}, y_i^{j+1} = +1)\}_{i=1}^{n^{j+1}}$, and the extended training sample set $S = \cup_{j=1}^{r-1} S^j = \{(x_1, y_1), \ldots, (x_l, y_l)\}$, where $l = 2 \times \sum_{j=1}^{r} n^j - n^1 - n^r$.
2) We let $\lambda^j = [\lambda_1^j, \ldots, \lambda_{n^j}^j]$ and $\delta^j = [\delta_1^j, \ldots, \delta_{n^j}^j]$, where λ_i^j and δ_i^j are the Lagrangian multipliers corresponding to the first and third inequality constraints in (2), respectively. Thus, $\alpha = [\lambda^1, \delta^1, \ldots, \lambda^{r-1}, \delta^{r-1}]$ is defined to be the row vector containing all the Lagrangian multipliers λ_i^j and δ_i^j.
3) We define the kernel matrix Q as $Q_{ik} = y_i y_k K(x_i, x_k)$ for all $1 \le i, k \le l$.

Based on the above notations, the dual problem can be formulated as follows:

$$\begin{aligned}
\min_{\alpha} \quad & \frac{1}{2}\alpha Q \alpha^T \\
\text{s.t.} \quad & \sum_{i \in S^j} y_i \alpha_i = 0, \quad \sum_{i \in S^j} \alpha_i \ge 2, \quad j = 1, \ldots, r-1 \\
& 0 \le \alpha_i \le C, \quad i = 1, \ldots, l
\end{aligned} \tag{3}$$

where $i \in S^j$ is the abbreviated form of $(x_i, y_i) \in S^j$.

Once the optimal solution α is obtained, the part $\langle w, \phi(x)\rangle$ of the rank-monotonic mapping function $f(x, j)$ in RKHS can be obtained as follows:

$$\langle w, \phi(x)\rangle = \frac{\sum_{i=1}^{l} y_i \alpha_i K(x_i, x)}{\sqrt{\alpha Q \alpha^T}}. \tag{4}$$

In addition, b_j can be obtained by solving the following linear equations:

$$\frac{\sum_{i=1}^{l} y_i \alpha_i K(x_i, x_{i_1})}{\sqrt{\alpha Q \alpha^T}} - b_j + d_j = 0 \tag{5}$$

$$\frac{\sum_{i=1}^{l} y_i \alpha_i K(x_i, x_{i_2})}{\sqrt{\alpha Q \alpha^T}} - b_j - d_j = 0 \tag{6}$$

[1]Normally, $\langle w, w\rangle = 1$ is used to keep the unique form of $\langle w, x\rangle - b_j$. However, $\langle w, w\rangle = 1$ is a nonconvex constraint, which was replaced by the convex constraint $\langle w, w\rangle \le 1$ in [6] since the optimal solution w would have unit magnitude after optimizing the objective function.

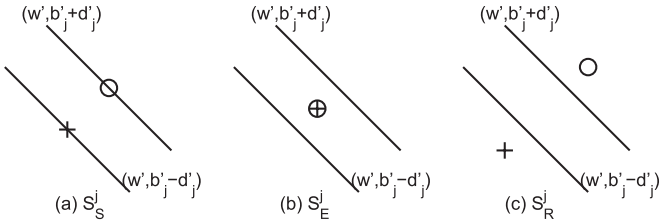


Fig. 2. Partitioning a two-class training sample set S^j, which is associated with the jth binary classification, into three independent sets by KKT-conditions. (a) S_S^j. (b) S_E^j. (c) S_R^j.

where $\{(x_{i_1}, y_{i_1}), (x_{i_2}, y_{i_2})\} \subseteq S^j$ with $y_{i_1} = -1$ and $y_{i_2} = +1$, and x_{i_1}, x_{i_2} are also support vectors with their weights $0 < \alpha_{i_1} < C$, $0 < \alpha_{i_2} < C$.

C. KKT Conditions

According to convex optimization theory [4], the solution of the dual problem (3) can be obtained by the following min–max problem:

$$\min_{0\le\alpha_i\le C}\max_{b'_j,d'_j\ge 0} W = \frac{1}{2}\sum_{i,k=1}^{l}\alpha_i\alpha_k Q_{ik} + \sum_{j=1}^{r-1} b'_j\left(\sum_{i\in S^j} y_i\alpha_i\right) + \sum_{j=1}^{r-1} d'_j\left(2 - \sum_{i\in S^j}\alpha_i\right) \tag{7}$$

where $b'_j \in \mathbb{R}$ and $d'_j \in \mathbb{R}_+$ are Lagrangian multipliers.

From the KKT theorem [5], we obtain the following KKT conditions:

$$\sum_{i\in S^j} y_i\alpha_i = 0 \tag{8}$$

$$p_j \stackrel{\text{def}}{=} \sum_{i\in S^j}\alpha_i \begin{cases} \ge 2 & \text{for } d'_j = 0 \\ = 2 & \text{for } d'_j > 0 \end{cases} \tag{9}$$

$$\forall i \in S^j : g_i \stackrel{\text{def}}{=} \frac{\partial W}{\partial\alpha_i} = \sum_{k=1}^{l}\alpha_k Q_{ik} + y_i b'_j - d'_j$$
$$\begin{cases} \ge 0 & \text{for } \alpha_i = 0 \\ = 0 & \text{for } 0 < \alpha_i < C \\ \le 0 & \text{for } \alpha_i = C. \end{cases} \tag{10}$$

According to the value of the function g_i, a two-class training sample set S^j associated with the jth binary classification is partitioned into three independent sets (Fig. 2).

1) $S_S^j = \{i \in S^j : g_i = 0,\ 0 < \alpha_i < C\}$, the set S_S^j includes margin support vectors strictly on the margins.
2) $S_E^j = \{i \in S^j : g_i \le 0,\ \alpha_i = C\}$, the set S_E^j includes error support vectors exceeding the margins.
3) $S_R^j = \{i \in S^j : g_i \ge 0,\ \alpha_i = 0\}$, the set S_R^j includes the remaining vectors ignored by the margins.

Thus, the extended training sample set S is partitioned into three independent sets, i.e., $S_S = \cup_{j=1}^{r-1} S_S^j$, $S_E = \cup_{j=1}^{r-1} S_E^j$, and $S_R = \cup_{j=1}^{r-1} S_R^j$.

In addition, according to the value of p_j in (9), we can define an active set $J \subseteq \{1, \ldots, r-1\}$ with $p_j = 2$ and $d'_j > 0$ for all $j \in J$.

TABLE I

THREE CASES OF THE CHANGE OF THE EXTENDED TRAINING SAMPLE SET S, WHEN A SAMPLE x_{new} IS ADDED INTO THE jTH CATEGORY. S_{new}^j DENOTES THE INCREMENT IN S^j, AND S_{new} DENOTES THE INCREMENT IN S, WHERE $S_{\text{new}} = S_{\text{new}}^{j-1} \cup S_{\text{new}}^j$

x_{new}^j	S_{new}^{j-1}	S_{new}^j	S_{new}
$j = 1$	$\emptyset$	$\{(x_{new}, -1)\}$	$\{(x_{new}, -1)\}$
$1 < j < r$	$\{(x_{new}, +1)\}$	$\{(x_{new}, -1)\}$	$\{(x_{new}, +1), (x_{new}, -1)\}$
$j = r$	$\{(x_{new}, +1)\}$	$\emptyset$	$\{(x_{new}, +1)\}$

TABLE II

TWO CASES OF CONFLICTS BETWEEN $\sum_{i\in S^{jc}} \Delta\alpha_i + \Delta\alpha_c = 0$ AND $\sum_{i\in S^{jc}} y_i\Delta\alpha_i + y_c\Delta\alpha_c = 0$, WHEN $|\sum_{i\in S_S^{jc}} y_i| = |S_S^{j_c}|$ AND $d'_j > 0$ WITH A SMALL INCREMENT $\Delta\alpha_c$

label of margin support vectors in S_S^{jc}		label of the new sample $(x_c, y_c) \in S_{new}^{jc}$		conflict
+1	-1	+1	-1	yes/no
√		√		no
√			√	yes
	√	√		yes
	√		√	no

III. INCREMENTAL SVOR LEARNING

In this section, we consider the incremental SVOR learning algorithm for the dual problem (3). When a sample x_{new} is added into the jth category, there correspondingly exists increments in S and S^j (Table I). We define the increments as S_{new} and S_{new}^j, respectively. Initially, we set the weight α_c of each sample (x_c, y_c) in S_{new} to zero. If this assignment satisfies the KKT conditions, adjustments are not needed. However, if this assignment violates the KKT conditions, additional adjustments become necessary [Fig. 1(b)]. The goal of the incremental SVOR algorithm is to find an effective method for updating the weights without retraining from scratch, when a sample in S_{new} violates the KKT conditions.

Compared with the formulations of standard SVM, one-class SVM, SVR, and ν-SVC, our SVOR formulation (3) has the following challenges, which prevent us from directly using the existing incremental SVM algorithms, including the C&P algorithm and AONSVM.

1) If $|\sum_{i\in S_S^{jc}} y_i| = |S_S^{j_c}|$, $d'_j > 0$, and the label of an added sample (x_c, y_c) in $S_{\text{new}}^{j_c}$ is different from those of the margin support vectors in $S_S^{j_c}$, there exists a conflict (referred to as Conflict-1) between (8) and (9) with a small increment of α_c (Table II). Conflict-1 is different to the one in ν-SVC, because an additional condition $d'_j > 0$ must be considered here.
2) The SVOR formulation (3) has multiple constraints of the mixture of an equality and an inequality, which is more complicated than a pair of equality constraints in ν-SVC, and an equality constraint in standard SVM, one-class SVM, and SVR.

To address these challenges, we propose an incremental SVOR algorithm (i.e., ISVOR, see Algorithm 1), which includes two steps, similar to AONSVM.

The first step is RAIA. Because there may exist Conflict-1 between (8) and (9) as shown in Table II, the feasible updating path leading to the eventual satisfaction of the KKT conditions will not be guaranteed. To overcome this problem, the limitation on the enlarged j_cth two-class training

Algorithm 1 Incremental SVOR Algorithm

Input: α, d' g, p, S, R, x_{new} (α, d', g and p satisfy the KKT conditions of S, x_{new} is the new sample added into the jth category.).

Output: α, d' g, p, S, R.

1: Compute S_{new} according to Table I.
2: **while** $S_{new} \neq \emptyset$ **do**
3: Read (x_c, y_c) from S_{new}, initial its weight $\alpha_c \leftarrow 0$ and compute g_c.
4: Update $S_{new} \leftarrow S_{new} - \{(x_c, y_c)\}$, $S^{j_c} \leftarrow S^{j_c} \cup \{(x_c, y_c)\}$, and $S \leftarrow S \cup \{(x_c, y_c)\}$.
5: **while** $g_c < 0$ and $\alpha_c < C$ **do**
6: Compute $\beta_{b'}^c$, $\beta_{d'_{J'_-}}^c$, $\beta_{S'_S}^c$, ρ_j^c, and γ_i^c.
7: Compute the maximal increment $\Delta\alpha_c^{max}$.
8: Update α, g, b', d', p, J'_-, $\overline{J'_+}$, S_S, S_E and S_R.
9: Update the inverse matrix R.
10: **end while**
11: Compute the inverse matrix $\widehat{R}$ based on R.
12: **while** $p_{j_c} < 2$ or ($p_{j_c} > 2$ & $d'_{j_c} > 0$) **do**
13: Compute $\widehat{\beta}_{b'}$, $\widehat{\beta}_{d'_{J'_+}}$, $\widehat{\beta}_{S'_S}$, $\widehat{\rho}_j$, and $\widehat{\gamma}_i$.
14: Compute the critical adjustment quantity $\Delta\zeta_{j_c}^*$.
15: Update α, g, b', d', p, J'_-, $\overline{J'_+}$, S_S, S_E and S_R.
16: Update the inverse matrix $\widehat{R}$.
17: **end while**
18: Compute the inverse matrix R based on $\widehat{R}$.
19: **end while**

samples imposed by inequality (9) is removed from this step, similar to AONSVM. In addition, our basic idea is gradually increasing α_c under the condition of rigorously keeping all the samples satisfying the KKT conditions, except that the inequality restriction (9) should be held for the weights of the enlarged j_cth two-class training samples (Fig. 3). This procedure is described with pseudocode in lines 5–10 of Algorithm 1, and the details are stated in Section III-A.

The second step is SRA, whose objective is to restore the inequality restriction (9) on the enlarged j_cth two-class training samples. Our idea is gradually adjusting p_{j_c} under the condition of rigorously keeping all samples satisfying the KKT conditions, until all the samples satisfy the KKT conditions (Fig. 4). In addition, to avoid the recurrence of the conflict (referred to as Conflict-2) between (8) and (9) if $|\sum_{i\in S_S^{j_c}} y_i| = |S_S^{j_c}|$ during the adjustments for p_{j_c}, a trick is used in this step, similar to AONSVM. This procedure is described with pseudocode in lines 11–17 of Algorithm 1, and the details are discussed in Section III-B.

Although ISVOR and AONSVM share the similar two step procedure, ISVOR has a more general framework for processing the objective function than AONSVM, from the following two aspects.

1) ISVOR can handle multiple binary classification problems simultaneously, especially the singularity of the key matrix. However, AONSVM just manages one binary classification problem, and does not need to take account of the singularity of the key matrix.
2) ISVOR can handle multiple inequality constraints in the objective function. However, AONSVM can only handle a pair of equality constraints.

If only considering two categories in OR, and transforming the inequality constraint into an equality constraint as in [28], ISVOR degenerates to AONSVM. If further discarding the inequality constraint from the above formulation, ISVOR degenerates to the C&P algorithm, similar to AONSVM [29]. Thus, ISVOR can be viewed as a generalization of the C&P algorithm and AONSVM.

A. Relaxed Adiabatic Incremental Adjustment

During the incremental adjustment for α_c, the weights of the samples in S_S, the Lagrange multipliers b'_j and d'_j should also be adjusted accordingly, to keep all the samples satisfying the KKT conditions, except that the restriction (9) should be held for the weights of the enlarged j_cth two-class training samples. Thus, we have the following linear system:

$$\forall j \neq j_c: \sum_{k\in S_S^j} y_k \Delta\alpha_k = 0 \tag{11}$$

$$j = j_c: \sum_{k\in S_S^j} y_k \Delta\alpha_k + y_c \Delta\alpha_c = 0 \tag{12}$$

$$\forall j \in J_-: \Delta p_j = \sum_{k\in S_S^j} \Delta\alpha_k = 0 \tag{13}$$

$$\forall i \in S_S^j: \Delta g_i = \sum_{k\in S_S} \Delta\alpha_k Q_{ik} + y_i \Delta b'_j - \boxed{\Delta d'_j} + \Delta\alpha_c Q_{ic} = 0 \tag{14}$$

where $j = 1, \ldots, r-1$. We let $\Delta b' = [\Delta b'_1, \ldots, \Delta b'_{r-1}]^T$, $\Delta d' = [\Delta d'_1, \ldots, \Delta d'_{r-1}]^T$, $E = [e_{S_S^1}, \ldots, e_{S_S^{r-1}}]$, and $U = [u_{S_S^1}, \ldots, u_{S_S^{r-1}}]$. Thus, the linear system (11)–(14) can be further rewritten as

$$\underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & U^T \\ \mathbf{0} & \mathbf{0} & E_{J_-}^T \\ U & E_{J_-} & Q_{S_S S_S} \end{bmatrix}}_{\widetilde{Q}_{\setminus (d'_{J_-})^2}} \cdot \begin{bmatrix} \Delta b' \\ \Delta d'_{J_-} \\ \Delta\alpha_{S_S} \end{bmatrix} = - \begin{bmatrix} \mathrm{u}_{j_c} \\ \mathbf{0} \\ Q_{S_S c} \end{bmatrix} \Delta\alpha_c. \tag{15}$$

It can be concluded that $\widetilde{Q}_{\setminus (d'_{J_-})^2}$ becomes singular in the following two cases.

SC-1: The first singular case is that $|\sum_{i\in S_S^j} y_i| = |S_S^j|$ for some $j \in J_-$, i.e., the samples of S_S^j only have one kind of labels for some $j \in J_-$. For example:
a) if $\forall i \in S_S^j$, $y_i = +1$, we have $e_{S_S^j} - u_{S_S^j} = \mathbf{0}$;
b) if $\forall i \in S_S^j$, $y_i = -1$, we have $e_{S_S^j} + u_{S_S^j} = \mathbf{0}$.
We define the index set $J'_- = \left\{ j \in J_- : |\sum_{i\in S_S^j} y_i| \neq |S_S^j| \right\}$. Thus, if $|J_- - J'_-| \neq 0$, $\widetilde{Q}_{\setminus (d'_{J_-})^2}$ becomes singular.

SC-2: The second singular case is that $|M'_j| > 1$ for some $j \in \{2, \ldots, r-1\}$. The M'_j is defined as $M'_j = \left\{ (x_i^j, -1) \in S_S^j : (x_i^j, +1) \in S_S^{j-1} \right\}$.

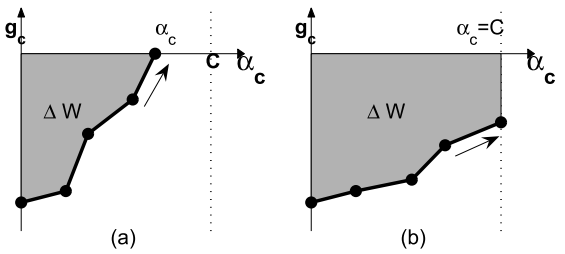


Fig. 3. RAIA. The two cases of (x_c, y_c) after RAIA (a) becomes a margin support vector or (b) it becomes an error support vector.

Supposing that there exist four samples indexed by i_1, i_2, k_1, and k_2, respectively, where $\{i_1, k_1\} \subset S_S^{j-1}$, $\{i_2, k_2\} \subset S_S^j$, $x_{i_1} = x_{i_2}$, and $x_{k_1} = x_{k_2}$. According to (14), we have $\Delta g_{i_1} + \Delta g_{i_2} = \Delta g_{k_1} + \Delta g_{k_2}$, which means $\widetilde{Q}_{i_1*} + \widetilde{Q}_{i_2*} = \widetilde{Q}_{k_1*} + \widetilde{Q}_{k_2*}$. In this case, it is easy to verify that $\widetilde{Q}_{\setminus(d'_{J_-})^2}$ is a singular matrix. When $M'_j \neq \emptyset$, we define M_j as the contracted set which is obtained by deleting any one sample from M'_j. Obviously, M_j is also an empty set when $M_j^+ = \emptyset$. Further, we let $M = M_2 \cup \ldots \cup M_{r-1}$, and $S'_S = S_S - M$. Thus, if $M \neq \emptyset$, $\widetilde{Q}_{\setminus(d'_{J_-})^2}$ is singular.

Now, we let $\widetilde{Q}_{\setminus(d'_{J'_-}, M)^2}$ denote the contracted matrix of $\widetilde{Q}_{\setminus(d'_{J'_-})^2}$. Similar to the analysis in [29, Th. 2], we can prove that $\widetilde{Q}_{\setminus(d'_{J'_-}, M)^2}$ has the inverse matrix R. Thus, the linear relationship between $\Delta b'$, $\Delta d'_{J'_-}$, $\Delta \alpha_{S'_S}$, and $\Delta \alpha_c$ can be easily solved as follows:

$$\begin{bmatrix} \Delta b' \\ \Delta d'_{J'_-} \\ \Delta \alpha_{S'_S} \end{bmatrix} = -R \begin{bmatrix} \mathrm{u}_{j_c} \\ \mathbf{0} \\ Q_{S'_S c} \end{bmatrix} \Delta \alpha_c \stackrel{\text{def}}{=} \begin{bmatrix} \beta_{b'}^c \\ \beta_{d'_{J'_-}}^c \\ \beta_{S'_S}^c \end{bmatrix} \Delta \alpha_c. \tag{16}$$

Substituting (16) into (13), we get the linear relationship between Δp_j and $\Delta \alpha_c$ as follows:

$$\Delta p_j = \left(\sum_{k \in S_S^j} \beta_k^c \right) \Delta \alpha_c \stackrel{\text{def}}{=} \rho_j^c \Delta \alpha_c. \tag{17}$$

Obviously, $\forall j \in J_-$, we have $\rho_j^c = 0$.

Finally, substituting (16) into (14), we can get the linear relationship between Δg_i ($\forall i \in S^j$) and $\Delta \alpha_c$ as follows:

$$\begin{aligned} \Delta g_i &= \left(\sum_{k \in S_S} \beta_k^c Q_{ik} + y_i \beta_{b'_j}^c - \boxed{\beta_{d'_j}^c} + Q_{\mathrm{ic}} \right) \\ \Delta \alpha_c &\stackrel{\text{def}}{=} \gamma_i^c \Delta \alpha_c \end{aligned} \tag{18}$$

where $j = 1, \ldots, r-1$. Obviously, $\forall i \in S_S$, we have $\gamma_i^c = 0$.

1) Some Details of RAIA: Once the linear relationships between $\Delta b'$, $\Delta d'_{J'_-}$, $\Delta \alpha_{S'_S}$, Δp, Δg, and $\Delta \alpha_c$ are available, the maximal increment $\Delta \alpha_c^{\max}$ can be computed for each incremental adjustment (Fig. 3), such that a certain sample migrates among the sets S_S, S_R, and S_E, or a certain index migrates between J'_- and $\overline{J'_+}$. There are four cases considered to account for such structural changes.

1) Some α_i in S_S reaches a bound. Compute the sets: $I_P^{S_S} = \{i \in S_S : \beta_i^c > 0\}$, $I_N^{S_S} = \{i \in S_S : \beta_i^c < 0\}$. Thus, the maximum possible weight updates are

$$\Delta \alpha_i^{\max} = \begin{cases} C - \alpha_i, & \text{if } i \in I_P^{S_S} \\ -\alpha_i, & \text{if } i \in I_N^{S_S} \end{cases}$$

and the maximal possible $\Delta \alpha_c^{S_S}$ before a certain sample in S_S moves to S_R or S_E is $\Delta \alpha_c^{S_S} = \min_{i \in I_P^{S_S} \cup I_N^{S_S}} (\Delta \alpha_i^{\max} / \beta_i^c)$.

2) A certain d'_j or p_j reaches zero. Compute the sets: $I^{J'_-} = \{j \in J'_- : \beta_{d'_j}^c < 0\}$, $I^{\overline{J'_+}} = \{j \in \overline{J'_+} : \rho_j^c < 0\}$. Thus, the maximal possible $\Delta \alpha_c^{J'_-, \overline{J'_+}}$ before a certain index in J'_- ($\overline{J'_+}$) migrates to $\overline{J'_+}$ (J'_-) is $\Delta \alpha_c^{J'_-, \overline{J'_+}} = \min \left\{ \min_{j \in I^{J'_-}} (-d'_j / \beta_{d'_j}^c), \min_{j \in I^{\overline{J'_+}}} (-p_j / \rho_j^c) \right\}$.

3) A certain g_i corresponding to a sample in S_R or S_E reaches zero. Compute the sets: $I_P^{S_E} = \{i \in S_E : \gamma_i^c > 0\}$, $I_N^{S_R} = \{i \in S_R : \gamma_i^c < 0\}$. Thus, the maximal possible $\Delta \alpha_c^{S_R, S_E}$ before a certain sample in S_R or S_E migrates to S_S is $\Delta \alpha_c^{S_R, S_E} = \min_{i \in I_P^{S_E} \cup I_N^{S_R}} (-g_i / \gamma_i^c)$.

4) α_c reaches the upper bound or g_c reaches zero. The maximal possible $\Delta \alpha_c^c$, before the new candidate sample (x_c, y_c) satisfies the restriction (7) of the KKT conditions, is $\Delta \alpha_c^c = \min \{C - \alpha_c, \ -g_c / \gamma_c^c\}$.

Finally, the smallest of the four values

$$\Delta \alpha_c^{\max} = \min \left\{ \Delta \alpha_c^{S_S}, \Delta \alpha_c^{J_-, \overline{J'_+}}, \Delta \alpha_c^{S_R, S_E}, \Delta \alpha_c^c \right\} \tag{19}$$

constitutes the maximal increment of $\Delta \alpha_c$.

Based on the maximal increment $\Delta \alpha_c^{\max}$, we can update α, g, b', d', and p according to (16)–(18), and J'_-, $\overline{J'_+}$, S_S, S_E and S_R according to (19).

Once the components of the set S'_S or J'_- are changed, i.e., a sample is either added to or removed from the set S'_S, or an index is added into or removed from the set J'_-, the changes of the inverse matrix can be found in Lemma 5 and 6 of [29]. In addition, after SRA, the inverse matrix R for the next round of RAIA can also be computed from $\widehat{R}$ using the same contracted rule.

2) Finite Convergence of RAIA: Obviously, RAIA is an iterative procedure. Thus, we are concerned about its finite convergence, which is the foundation of the usefulness of RAIA. Specifically, the finite convergence of RAIA means that a new candidate sample (x_c, y_c) will satisfy the KKT conditions in a finite number of steps, except that the restriction (9) does not need to hold for all the weights of the enlarged j_cth two-class training samples. In this section, we will prove it. To prove the finite convergence of RAIA, we first prove that the objective function W is strictly monotonically decreasing during RAIA (Theorem 1).

Theorem 1: During RAIA, the objective function W is strictly monotonically decreasing.

Proof: During RAIA, suppose that the previous adjustment is indexed by k, the immediate next is indexed by $k+1$,

and let $\beta_{S_R}^c = \mathbf{0}$, $\beta_{S_E}^c = \mathbf{0}$, $\beta_c^c = 1$, then we have

$$\begin{aligned}
&W^{[k+1]} \\
&= \frac{1}{2}\sum_{i_1,i_2\in S}\left(\alpha_{i_1}^{[k]}+\beta_{i_1}^{[k]}\Delta\alpha_c^{[k]}\right)\left(\alpha_{i_2}^{[k]}+\beta_{i_2}^{[k]}\Delta\alpha_c^{[k]}\right)Q_{i_1i_2} \\
&\quad+\sum_{j=1}^{r-1}\left(b_j'^{[k]}+\beta_{b_j'}^{[k]}\Delta\alpha_c^{[k]}\right)\sum_{i\in S^j}y_i\left(\alpha_i^{[k]}+\beta_i^{[k]}\Delta\alpha_c^{[k]}\right) \\
&\quad+\sum_{j=1}^{r-1}\left(d_j'^{[k]}+\beta_{d_j'}^{[k]}\Delta\alpha_c^{[k]}\right)\left(\sum_{i\in S^j}\alpha_i^{[k]}+\beta_i^{[k]}\Delta\alpha_c^{[k]}-2\right) \\
&= W^{[k]}+\sum_{i\in S}g_i^{[k]}\beta_i^{[k]}\Delta\alpha_c^{[k]}+\frac{1}{2}\sum_{i\in S}\gamma_i^{[k]}\beta_i^{[k]}\left(\Delta\alpha_c^{[k]}\right)^2 \\
&= W^{[k]}+g_c^{[k]}\Delta\alpha_c^{[k]}+\frac{1}{2}\gamma_c^{[k]}\left(\Delta\alpha_c^{[k]}\right)^2 \\
&= W^{[k]}+\left(g_c^{[k]}+\frac{1}{2}\gamma_c^{[k]}\Delta\alpha_c^{[k]}\right)\Delta\alpha_c^{[k]}.
\end{aligned}$$

In other words, $W^{[k+1]}-W^{[k]} = \left(g_c^{[k]}+1/2\gamma_c^{[k]}\Delta\alpha_c^{[k]}\right)\Delta\alpha_c^{[k]}$. Similar to [29, Corollary 8], we can prove that the maximal increment $\Delta\alpha_c^{\max} > 0$ for each RAIA. In addition, it is easy to verify that $g_c^{[k]}+1/2\gamma_c^{[k]}\Delta\alpha_c^{[k]} < 0$, so we have $W^{[k+1]} - W^{[k]} < 0$. This completes the proof. ∎

Let $(W^{[1]}, W^{[2]}, W^{[3]}, \ldots)$ be the sequence generated during RAIA. Based on Theorem 1, we know that $(W^{[1]}, W^{[2]}, W^{[3]}, \ldots)$ is a monotonically decreasing sequence. To further prove the finite convergence of RAIA, we can show that the sequence is finite and converges to the KKT conditions, except that the restriction (9) does not need to hold for all the weights of the enlarged j_cth two-class training samples, which is similar to [29, Th. 14].

B. SRA

After RAIA, the KKT conditions are satisfied by all the samples, except that the inequality restriction (9) is satisfied by the enlarged j_cth two-class training samples. In the SRA step, we gradually adjust p_{j_c} to restore the inequality restriction (9), so that the KKT conditions are satisfied by all the samples.

For each adjustment of p_{j_c}, the weights of the samples in S_S, the Lagrange multipliers b' and d' should also be adjusted accordingly, to keep all the samples satisfying the KKT conditions. Thus, we have the following linear system:

$$\forall j: \quad \sum_{k\in S_S^j} y_k\Delta\alpha_k = 0 \tag{20}$$

$$\forall j\in J_-': \quad \sum_{k\in S_S^j}\Delta\alpha_k = 0 \tag{21}$$

$$j = j_c: \sum_{k\in S_S^j}\Delta\alpha_k + \varepsilon\Delta d_{j_c}' + \Delta\zeta_{j_c} = 0 \tag{22}$$

$$\forall i\in S_S^j: \Delta g_i = \sum_{k\in S_S}\Delta\alpha_k Q_{ik} + y_i\Delta b_j' - \boxed{\Delta d_j'} = 0 \tag{23}$$

where $j = 1,\ldots,r-1$, $\Delta\zeta_{j_c}$ is the introduced variable for adjusting p_{j_c}, ε is any negative number, and $\varepsilon\Delta d_{j_c}'$ in (22) is an extra term. The trick of using the extra term can prevent the reoccurrence of Conflict-2, similar to [28].

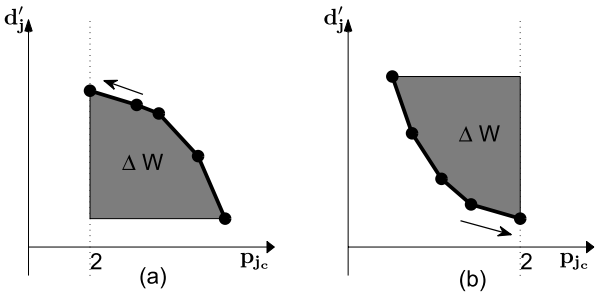


Fig. 4. SRA. The objective of SRA is to restore the inequality restriction (9) on the enlarged j_cth two-class training samples. Initial condition of p_{j_c} may be (a) $p_{j_c} > 2$ or (b) $p_{j_c} < 2$.

Then, the linear system (20)–(23) can be further rewritten as

$$\underbrace{\begin{bmatrix}\mathbf{0} & \mathbf{0} & U^T\\ \mathbf{0} & O & E^T\\ U & E & Q_{S_SS_S}\end{bmatrix}}_{\widehat{Q}}\cdot\begin{bmatrix}\Delta b'\\ \Delta d'\\ \Delta\alpha_{S_S}\end{bmatrix} = -\begin{bmatrix}\mathbf{0}\\ \mathrm{v}_{j_c}\\ \mathbf{0}\end{bmatrix}\Delta\zeta_{j_c}. \tag{24}$$

We let $\widehat{Q}_{\setminus(d'_{J'_+},M)^2}$ denote the contracted matrix of $\widehat{Q}$. Similar to the analysis of [29, Th. 7], we can prove that $\widehat{Q}_{\setminus(d'_{J'_+},M)^2}$ has the inverse matrix $\widehat{R}$. Then, the linear relationship between $\Delta b'$, $\Delta d'_{J'_+}$, $\Delta\alpha_{S'_S}$ and $\Delta\zeta_{j_c}$ can be obtained as follows:

$$\begin{bmatrix}\Delta b'\\ \Delta d'_{J'_+}\\ \Delta\alpha_{S'_S}\end{bmatrix} = -\widehat{R}\cdot\begin{bmatrix}\mathbf{0}\\ \mathrm{v}_{j_c}\\ \mathbf{0}\end{bmatrix}\Delta\zeta_{j_c} \overset{\text{def}}{=} \begin{bmatrix}\widehat{\beta}_{b'}\\ \widehat{\beta}_{d'_{J'_+}}\\ \widehat{\beta}_{S'_S}\end{bmatrix}\Delta\zeta_{j_c}. \tag{25}$$

From (25), we have $\sum_{i\in S^{j_c}}\Delta\alpha_i = -(1+\varepsilon\widehat{\beta}_{d'_{j_c}})\Delta\zeta_{j_c}$,[2] which implies that the control of the adjustment of p_{j_c} can be achieved by $\Delta\zeta_{j_c}$.

Substituting (25) into (21), we get the linear relationship between Δp_j and $\Delta\zeta_{j_c}$ as follows:

$$\Delta p_j = \left(\sum_{k\in S_S^j}\widehat{\beta}_k\right)\Delta\zeta_{j_c} \overset{\text{def}}{=} \widehat{\rho}_j\Delta\zeta_{j_c}. \tag{26}$$

Obviously, $\forall j\in J_-'$, we have $\widehat{\rho}_j = 0$.

Finally, substituting (25) into (23), we can get the linear relationship between Δg_i ($\forall i\in S^j$) and $\Delta\zeta_{j_c}$ as follows:

$$\Delta g_i = \left(\sum_{k\in S_S}\widehat{\beta}_k Q_{ik} + y_i\widehat{\beta}_{b'_j} - \boxed{\widehat{\beta}_{d'_j}}\right)\Delta\zeta_{j_c} \overset{\text{def}}{=} \widehat{\gamma}_i\Delta\zeta_{j_c} \tag{27}$$

where $j = 1,\ldots,r-1$. Obviously, $\forall i\in S_S$, we have $\widehat{\gamma}_i = 0$.

1) Some Details of SRA: Similar to RAIA, we need to compute the critical adjustment quantity $\Delta\zeta_{j_c}^*$ for each restoration adjustment (Fig. 4), such that a certain sample migrates among the sets S_S, S_R, and S_E, or a certain index migrates between J_-' and J_+'. If $p_{j_c} > 2$, we will compute the maximal adjustment quantity $\Delta\zeta_{j_c}^{\max}$, and let $\Delta\zeta_{j_c}^* = \Delta\zeta_{j_c}^{\max}$. Otherwise, we will compute the minimal adjustment quantity $\Delta\zeta_{j_c}^{\min}$, and

[2]Similar to the analysis of [29, Th. 9], we can prove that $(1+\varepsilon\widehat{\beta}_{d'_{j_c}}) \geq 0$ under the condition that $\varepsilon < 0$.

let $\Delta\zeta_{j_c}^{*} = \Delta\zeta_{j_c}^{\min}$. Four scenarios are considered to account for such structural changes.

1) A certain α_i in S_S reaches a bound. First, compute the sets: $I_P^{S_S} = \{i \in S_S : \widehat{\beta}_i > 0\}$, $I_N^{S_S} = \{i \in S_S : \widehat{\beta}_i < 0\}$. Two possible cases are considered for the critical adjustment quantity $\Delta\zeta_{j_c}^{S_S}$ before a certain sample in S_S moves to S_R or S_E:
   a) when $p_{j_c} > 2$, the possible weight updates are
   $$\Delta\alpha_i^{\max} = \begin{cases} C - \alpha_i, & \text{if } i \in I_P^{S_S} \\ -\alpha_i, & \text{if } i \in I_N^{S_S} \end{cases}$$
   then the maximal possible $\Delta\zeta_{j_c}^{S_S} = \min_{i \in I_P^{S_S} \cup I_N^{S_S}} (\Delta\alpha_i^{\max}/\widehat{\beta}_i)$;
   b) when $p_{j_c} < 2$, the possible weight updates are
   $$\Delta\alpha_i^{\max} = \begin{cases} -\alpha_i, & \text{if } i \in I_P^{S_S} \\ C - \alpha_i, & \text{if } i \in I_N^{S_S} \end{cases}$$
   then the minimal possible $\Delta\zeta_{j_c}^{S_S} = \max_{i \in I_P^{S_S} \cup I_N^{S_S}} (\Delta\alpha_i^{\max}/\widehat{\beta}_i)$.
2) A certain d'_j or p_j reaches zero. We consider two cases for the maximal possible $\Delta\alpha_c^{J'_-, \overline{J'_+}}$ before a certain index in J'_- ($\overline{J'_+}$) migrates to $\overline{J'_+}$ (J'_-):
   a) when $p_{j_c} > 2$, we compute the sets: $I^{J'_-} = \{j \in J'_- : \beta_{d'_j}^c < 0\}$, $I^{\overline{J'_+}} = \{j \in \overline{J'_+} : \rho_j^c < 0\}$. $\Delta\alpha_c^{J'_-, \overline{J'_+}} = \min\left\{\min_{j \in I^{J'_-}}(-d'_j/\beta_{d'_j}^c), \min_{j \in I^{\overline{J'_+}}}(-p_j/\rho_j^c)\right\}$;
   b) when $p_{j_c} < 2$, we compute the sets: $I^{J'_-} = \{j \in J'_- : \beta_{d'_j}^c > 0\}$, $I^{\overline{J'_+}} = \{j \in \overline{J'_+} : \rho_j^c > 0\}$. $\Delta\alpha_c^{J'_-, \overline{J'_+}} = \max\left\{\max_{j \in I^{J'_-}}(-d'_j/\beta_{d'_j}^c), \max_{j \in I^{\overline{J'_+}}}(-p_j/\rho_j^c)\right\}$.
3) A certain g_i in S_R or S_E reaches zero. Two cases are considered for the critical adjustment quantity $\Delta\zeta_{j_c}^{S_R, S_E}$ before a certain sample in S_R or S_E moves to S_S:
   a) when $p_{j_c} > 2$, we compute the sets: $I_P^{S_E} = \{i \in S_E : \widehat{\gamma}_i > 0\}$, $I_N^{S_R} = \{i \in S_R : \widehat{\gamma}_i < 0\}$. $\Delta\zeta_{j_c}^{S_R, S_E} = \min_{i \in I_P^{S_E} \cup I_N^{S_R}}(-g_i/\widehat{\gamma}_i)$;
   b) when $p_{j_c} < 2$, we compute the sets: $I_N^{S_E} = \{i \in S_E : \widehat{\gamma}_i < 0\}$, $I_P^{S_R} = \{i \in S_R : \widehat{\gamma}_i > 0\}$. $\Delta\zeta_{j_c}^{S_R, S_E} = \max_{i \in I_N^{S_E} \cup I_P^{S_R}}(-g_i/\widehat{\gamma}_i)$.
4) The restriction (9) on the enlarged j_cth two-class training samples is restored. Two cases must be considered for the critical adjustment quantity $\Delta\zeta_{j_c}^{\omega}$, before p_{j_c} and d'_{j_c} satisfy the inequality restriction (9):
   a) when $p_{j_c} > 2$, the maximal possible $\Delta\zeta_{j_c}^{\omega} = \min\left\{(p_{j_c} - 2/1 + \varepsilon\widehat{\beta}_{d'_{j_c}}), (d'_{j_c}/\widehat{\beta}_{d'_{j_c}})\right\}$[3];
   b) when $p_{j_c} < 2$, the minimal possible $\Delta\zeta_{j_c}^{\omega} = (p_{j_c} - 2/1 + \varepsilon\widehat{\beta}_{d'_{j_c}})$.

[3]We can prove that $\widehat{\beta}_{d'_{j_c}} > 0$ based on $\widehat{\beta}_{d'_{j_c}} = -\widehat{R}_{d'_{j_c} d'_{j_c}} = -\det(\widehat{Q}_{\setminus(d'_{J'_-}, M)(d'_{J'_-}, M)})/\det(\widehat{Q}_{\setminus(d'_{J'_+}, M)(d'_{J'_+}, M)})$, and the analysis in [29, Th. 7].

TABLE III
DATA SETS USED IN THE EXPERIMENTS

Dataset	Sample size	Attributes	Categories
Bank	4,500	32	5(Regression)
Computer Activity	8,192	21	5(Regression)
Friedman	40,000	10	5(Regression)
Census	22,784	16	5(Regression)
Abalone	4,177	8	5(Regression)
Winequality-red	1,599	11	6(OR)
Winequality-white	4,898	11	7(OR)
Spine Image	350	5	5(OR)

Finally, if $p_{j_c} > 2$, the smallest of the four values

$$\Delta\zeta_{j_c}^{\max} = \min\left\{\Delta\zeta_{j_c}^{S_S}, \Delta\alpha_c^{J'_-, \overline{J'_+}}, \Delta\zeta_{j_c}^{S_R, S_E}, \Delta\zeta_{j_c}^{\omega}\right\} \quad (28)$$

constitutes the maximal increment of $\Delta\zeta_{j_c}$. Otherwise, the largest of the four values

$$\Delta\zeta_{j_c}^{\min} = \max\left\{\Delta\zeta_{j_c}^{S_S}, \Delta\alpha_c^{J'_-, \overline{J'_+}}, \Delta\zeta_{j_c}^{S_R, S_E}, \Delta\zeta_{j_c}^{\omega}\right\} \quad (29)$$

constitutes the minimal decrement of $\Delta\zeta_{j_c}$.

After the critical adjustment quantity $\Delta\zeta_{j_c}^{*}$ is determined, α, g, b', d', p, J'_-, $\overline{J'_+}$, S_S, S_E, and S_R can be updated according to (25)–(27), and (28) or (29), respectively.

Similar to RAIA, once the components of the set S'_S or J'_- are changed, the inverse matrix R can be updated by the expanded rule of Lemma 5, or the contracted rule of Lemma 6, in [29]. After RAIA, the inverse matrix $\widehat{R}$ for the forthcoming round of SRA can be expanded based on R using the same expanded rule.

2) Finite Convergence of SRA: SRA tries to adjust p_{j_c} to restore the inequality restriction (9), so that the KKT conditions are satisfied by all the samples. Obviously, SRA is also an iterative procedure. Thus, the finite convergence is the cornerstone of the usefulness of SRA. In this section, we will prove the finite convergence of SRA.

To prove the finite convergence of SRA, we first prove Theorem 2, which demonstrates that the objective function W is strictly monotonically increasing during SRA.

Theorem 2: During SRA, the objective function W is strictly monotonically increasing.

Proof: During SRA, let $\widehat{\beta}_{S_R} = \mathbf{0}$, $\widehat{\beta}_{S_E} = \mathbf{0}$, the superscript $[k]$ denotes the kth adjustment, then we have

$$\begin{aligned} & W^{[k+1]} \\ &= \frac{1}{2}\sum_{i_1, i_2 \in S}\left(\alpha_{i_1}^{[k]} + \widehat{\beta}_{i_1}^{[k]}\Delta\zeta_{j_c}^{*[k]}\right)\left(\alpha_{i_2}^{[k]} + \widehat{\beta}_{i_2}^{[k]}\Delta\zeta_{j_c}^{*[k]}\right)Q_{i_1 i_2} \\ &\quad + \sum_{j=1}^{r-1}\left(b'^{[k]}_j + \widehat{\beta}_{b'_j}^{[k]}\Delta\zeta_{j_c}^{*[k]}\right)\sum_{i \in S^j} y_i\left(\alpha_i^{[k]} + \widehat{\beta}_i^{[k]}\Delta\zeta_{j_c}^{*[k]}\right) \\ &\quad + \sum_{j=1}^{r-1}\left(\sum_{i \in S^j}\alpha_i^{[k]} + \widehat{\beta}_i^{[k]}\Delta\zeta_{j_c}^{*[k]} - 2\right)\left(d'^{[k]}_j + \widehat{\beta}_{d'_j}^{[k]}\Delta\zeta_{j_c}^{*[k]}\right) \\ &= W^{[k]} + \sum_{i \in S} g_i^{[k]}\widehat{\beta}_i^{[k]}\Delta\zeta_{j_c}^{*[k]} + \left(\sum_{i \in S^{j_c}}\alpha_i^{[k]} - 2\right)\widehat{\beta}_{d'_{j_c}}^{[k]}\Delta\zeta_{j_c}^{*[k]} \\ &\quad + \frac{1}{2}\widehat{\beta}_{d'_{j_c}}^{[k]}\sum_{i \in S^{j_c}}\widehat{\beta}_i^{[k]}\left(\Delta\zeta_{j_c}^{*[k]}\right)^2 + \frac{1}{2}\sum_{i \in S}\widehat{\gamma}_i^{[k]}\widehat{\beta}_i^{[k]}\left(\Delta\zeta_{j_c}^{*[k]}\right)^2 \end{aligned}$$

TABLE IV
NUMBERS OF OCCURRENCES OF CONFLICT-1, CONFLICT-2, SC-1, AND SC-2 ON THE EIGHT DATA SETS OVER 50 TRIALS. NOTE THAT L, P, AND G ARE THE ABBREVIATIONS OF LINEAR, POLYNOMIAL, AND GAUSSIAN KERNELS, RESPECTIVELY

Dataset	Size	Conflict-1			Conflict-2			SC-1			SC-2			Dataset	Size	Conflict-1			Conflict-2			SC-1			SC-2		
		L	P	G	L	P	G	L	P	G	L	P	G			L	P	G	L	P	G	L	P	G	L	P	G
Bank	10	0	0	0	0	1	0	5	0	0	86	139	136	Abalone	10	14	23	6	4	20	0	1	0	0	24	10	6
	15	0	1	0	0	0	0	0	0	0	118	151	134		15	0	7	1	24	5	0	0	0	0	42	12	9
	20	0	0	0	0	0	0	0	0	0	166	246	132		20	2	6	0	0	1	0	0	0	0	65	13	18
	25	0	0	0	0	0	0	0	0	3	188	206	132		25	3	0	0	2	0	0	0	0	0	54	22	10
	30	0	0	0	0	0	0	0	0	0	203	212	134		30	6	2	0	0	1	0	0	0	0	76	36	29
Computer Activity	10	1	15	0	0	6	0	0	0	0	29	114	112	Winequality-red	10	4	19	0	21	20	0	0	0	0	47	185	189
	15	0	11	0	1	16	0	0	0	0	60	96	105		15	2	13	0	28	9	0	2	0	0	178	275	194
	20	1	5	0	0	5	0	3	0	0	43	86	110		20	0	16	0	0	0	0	0	0	0	319	348	227
	25	0	1	0	0	1	0	0	0	0	93	66	121		25	13	4	0	0	12	0	0	0	0	380	352	251
	30	0	0	0	0	2	0	0	0	0	177	96	108		30	0	7	0	0	5	0	0	0	0	362	347	237
Friedman	10	1	9	12	0	6	2	0	0	0	81	94	100	Winequality-white	10	0	23	0	0	22	0	0	2	0	61	161	198
	15	0	8	2	0	1	0	0	1	0	261	218	298		15	0	19	0	0	13	0	0	0	0	215	298	200
	20	0	9	14	0	5	3	0	2	1	318	305	289		20	0	11	0	0	5	0	0	0	0	283	311	198
	25	0	9	13	0	0	0	0	1	0	354	315	291		25	1	0	0	0	5	0	2	0	0	382	327	203
	30	2	8	12	0	2	1	0	0	0	327	298	307		30	0	3	0	0	7	0	0	0	1	355	325	200
Census	10	3	6	0	23	13	0	1	0	0	112	115	107	Spine Image	10	30	22	4	7	34	0	0	0	0	45	33	156
	15	2	5	1	0	5	0	0	0	0	130	110	121		15	20	29	5	20	0	0	0	2	0	133	119	199
	20	0	4	0	2	4	0	5	0	0	138	125	113		20	22	31	6	3	6	2	1	0	0	225	238	214
	25	4	2	0	0	4	0	0	0	0	155	107	127		25	12	27	5	2	7	0	2	0	0	308	287	214
	30	3	0	0	0	1	0	0	0	0	169	126	132		30	9	10	4	2	9	0	2	0	2	278	257	226

$$
\begin{aligned}
&= W^{[k]} + \frac{1}{2}\widehat{\beta}^{[k]}_{d'_{j_c}} \sum_{i\in S^{j_c}} \widehat{\beta}^{[k]}_i \left(\Delta\zeta^{*\,[k]}_{j_c}\right)^2 \\
&\quad + \widehat{\beta}^{[k]}_{d'_{j_c}} \Delta\zeta^{*\,[k]}_{j_c} \left(\sum_{i\in S^{j_c}} \alpha^{[k]}_i - 2\right) \\
&= W^{[k]} + \left(\sum_{i\in S^{j_c}} \alpha^{[k]}_i - 2 - \frac{1}{2}\left(1+\varepsilon\widehat{\beta}^{[k]}_{d'_{j_c}}\right)\Delta\zeta^{*\,[k]}_{j_c}\right) \\
&\quad \cdot \widehat{\beta}^{[k]}_{d'_{j_c}} \Delta\zeta^{*\,[k]}_{j_c}.
\end{aligned}
$$

In other words, $W^{[k+1]} - W^{[k]} = \left(\sum_{i\in S^{j_c}} \alpha^{[k]}_i - 2 - 1/2\left(1+\varepsilon\widehat{\beta}^{[k]}_{d'_{j_c}}\right)\Delta\zeta^{*\,[k]}_{j_c}\right)\widehat{\beta}^{[k]}_{d'_{j_c}}\Delta\zeta^{*\,[k]}_{j_c}$. Similar to the analysis in [29, Th. 9], we can prove that $(1+\varepsilon\widehat{\beta}_{d'_{j_c}}) \geq 0$ under the condition that $\varepsilon < 0$. Similar to [29, Corollary 8], we can prove that $\Delta\zeta^*_{j_c} > 0$ if $p_{j_c} > 2$, and $\Delta\zeta^*_{j_c} < 0$ if $p_{j_c} < 2$, for each SRA. Similar to the analysis in [29, Th. 7], we can prove $\widehat{\beta}^{[k]}_{d'_{j_c}} > 0$. Furthermore, it is easy to verify that $\left(\sum_{i\in S^{j_c}} \alpha^{[k]}_i - 2 - (1/2)\left(1+\varepsilon\widehat{\beta}^{[k]}_{d'_{j_c}}\right)\Delta\zeta^{*\,[k]}_{j_c}\right)\Delta\zeta^{*\,[k]}_{j_c} > 0$. So, $W^{[k+1]} - W^{[k]} > 0$. This completes the proof. ∎

Similar to RAIA, we let $(W^{[1]}, W^{[2]}, W^{[3]}, \ldots)$ be the sequence generated during SRA. Based on Theorem 2, we can know that $(W^{[1]}, W^{[2]}, W^{[3]}, \ldots)$ is a monotonically increasing sequence. Similar to the analysis of [29, Th. 15], we can further prove that the sequence is finite and converges to the optimal solution to $\min_{0\leq\alpha_i\leq C} \max_{b'_j, d'_j \geq 0} W$. Combined with the finite convergence of RAIA, it can be concluded that ISVOR converges to the optimal solution to $\min_{0\leq\alpha_i\leq C} \max_{b'_j, d'_j \geq 0} W$ in a finite number of steps.

IV. EXPERIMENTAL SETUP

A. Design of Experiments

To demonstrate the usefulness of ISVOR, and show its advantage in terms of computation efficiency, we conduct a detailed experimental study.

To demonstrate the usefulness of ISVOR, we investigate the existence of the conflicts, the singularities of $\widetilde{Q}_{\setminus(d'_{\overline{J}_-})^2}$ and $\widehat{Q}$, and the finite convergence of ISVOR. To validate the existence of the conflicts, we count the events of Conflict-1 and Conflict-2 during RAIA and SRA, over 50 trials. To investigate the singularities of $\widetilde{Q}_{\setminus(d'_{\overline{J}_-})^2}$ and $\widehat{Q}$, we count the occurrences of SC-1 and SC-2 (Section III-A), during ISVOR over 50 trials. To illustrate the fast convergence of ISVOR empirically, we investigate the average numbers of the iterations of RAIA, and SRA, over 20 trials.

To show the advantage of ISVOR, we compare the running time of ISVOR with the batch algorithm (i.e., the SMO algorithm [8]) of SMF and EXC, which are called SMO-SMF and SMO-EXC, respectively, and the incremental algorithm of PSVM (i.e., IPSVM). We also compare the generalization performance of SMF, EXC, PSVM, and MSMF, which correspond to the above four algorithms (i.e., SMO-SMF, SMO-EXC, IPSVM, and ISVOR, respectively). Specifically, to have a better comparison of the generalization performance, two evaluation metrics are utilized to quantify the accuracy of predicted ordinal scales $\{\widehat{j_1}, \ldots, \widehat{j_n}\}$ with respect to true targets $\{j_1, \ldots, j_n\}$. They are:

1) mean absolute error (MAE): i.e., $1/n \sum_{i=1}^n |\widehat{j_i} - j_i|$;
2) mean zero-one error (MZE): i.e., $1/n \sum_{i=1}^n [\widehat{j_i} \neq j_i]$.[4]

B. Implementation

As mentioned in Section III, our incremental scenario is processing one added sample at a time. When a sample is added into the original $\sum_{j=1}^r n^j$ training samples, IPSVM and ISVOR update the weights without retraining from scratch based on their corresponding methods. However, the batch algorithms SMO-SMF and SMO-EXC retrain the weights from scratch for the original samples with the added one.

[4]The boolean test $[\cdot]$ is 1 if the inner condition is true, and 0 otherwise.

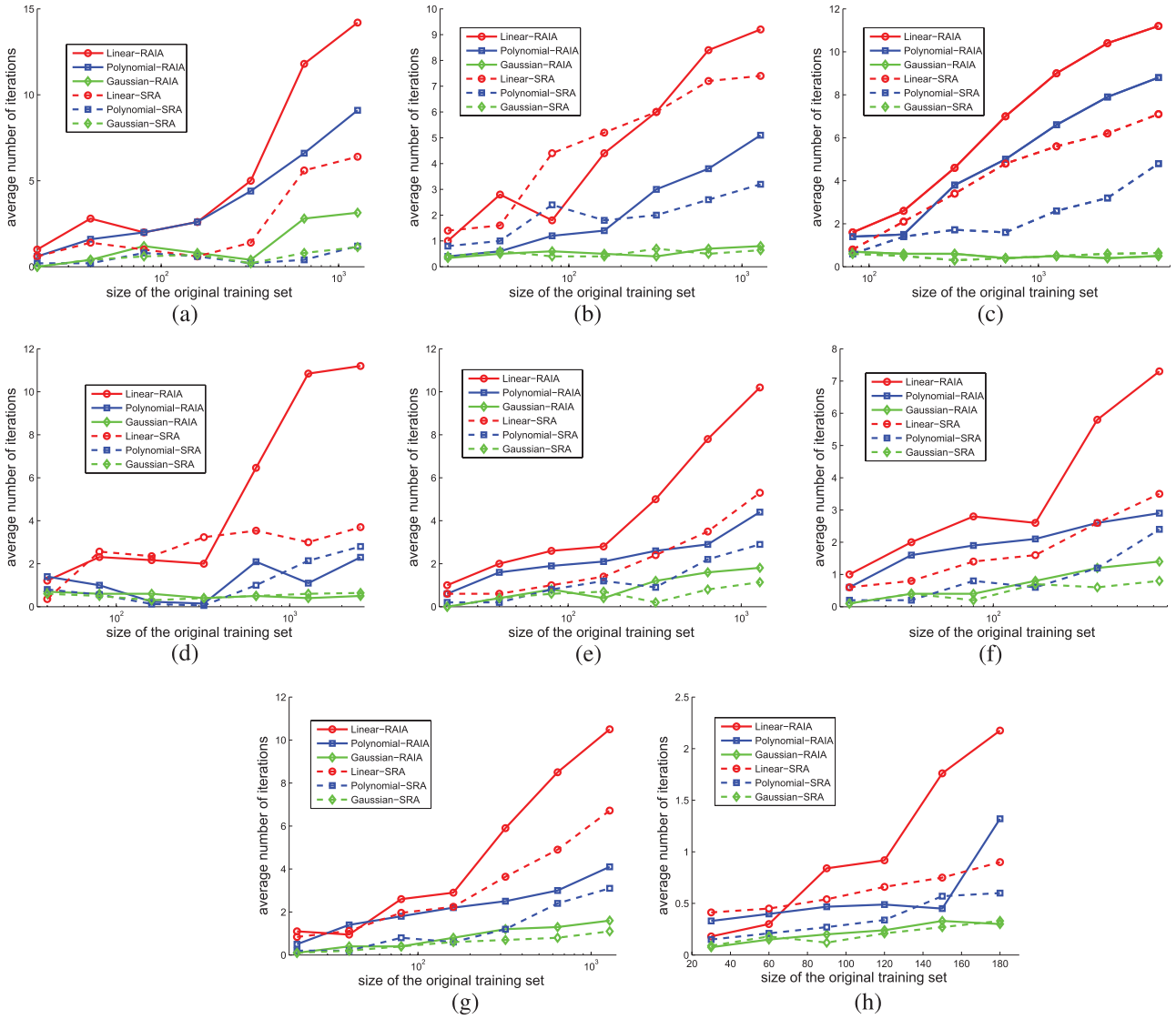


Fig. 5. Average numbers of iterations of RAIA and SRA on the different data sets. (a) Bank. (b) Computer Activity. (c) Friedman. (d) Census. (e) Abalone. (f) Winequality-red. (g) Winequality-white. (h) Spine Image.

We implemented ISVOR in MATLAB, and used the MATLAB code of [7] to implement IPSVM directly. We also implemented SMO-SMF in MATLAB. Specially, SMO-EXC was implemented in [8] in C, and this C implementation is used in our experiments. Generally speaking, a program written in C always runs much faster than the same one written in MATLAB, and hence, it is inappropriate to compare the running time of the MATLAB and C programs directly. However, the running time of SMO-EXC is still reported to compare with the other three algorithms, to a certain extent.

Experiments were performed on a 2.5-GHz Intel Core i5 machine with 8-GB RAM. For kernels, the linear kernel $K(x_1, x_2) = x_1 \cdot x_2$, polynomial kernel $K(x_1, x_2) = (x_1 \cdot x_2 + 1)^d$, and Gaussian kernel $K(x_1, x_2) = \exp(-\|x_1 - x_2\|^2/2\sigma^2)$ are used in our experiments, where the parameters d and σ are set to 2, and 2.2361, respectively, unless otherwise specified. In addition, the regularization parameter C is fixed to 10 unless otherwise specified. The parameter, ε of SRA, is fixed to -1 throughout all the experiments.[5] The tolerance parameters of SMO-SMF and SMO-EXC are all fixed to 10^{-10}.

C. Data Sets

Table III summarizes the characteristics of eight data sets used in our experiments, where five data sets are from regression problems, and the other three are real OR data sets. The detailed description of each data set is stated as follows.

1) Regression Data Sets: The data sets Bank, Computer Activity, Census, and Abalone are available at http://www.gatsby.ucl.ac.uk/~chuwei/ordinalregression.html,

[5]Similar with AONSVM [28], it is easy to verify that ε can determine $\Delta\zeta^*_{j_c}$, but is independent with the structural changes of the sets S_S, S_R, S_E, J'_-, and $\overline{J'_+}$.

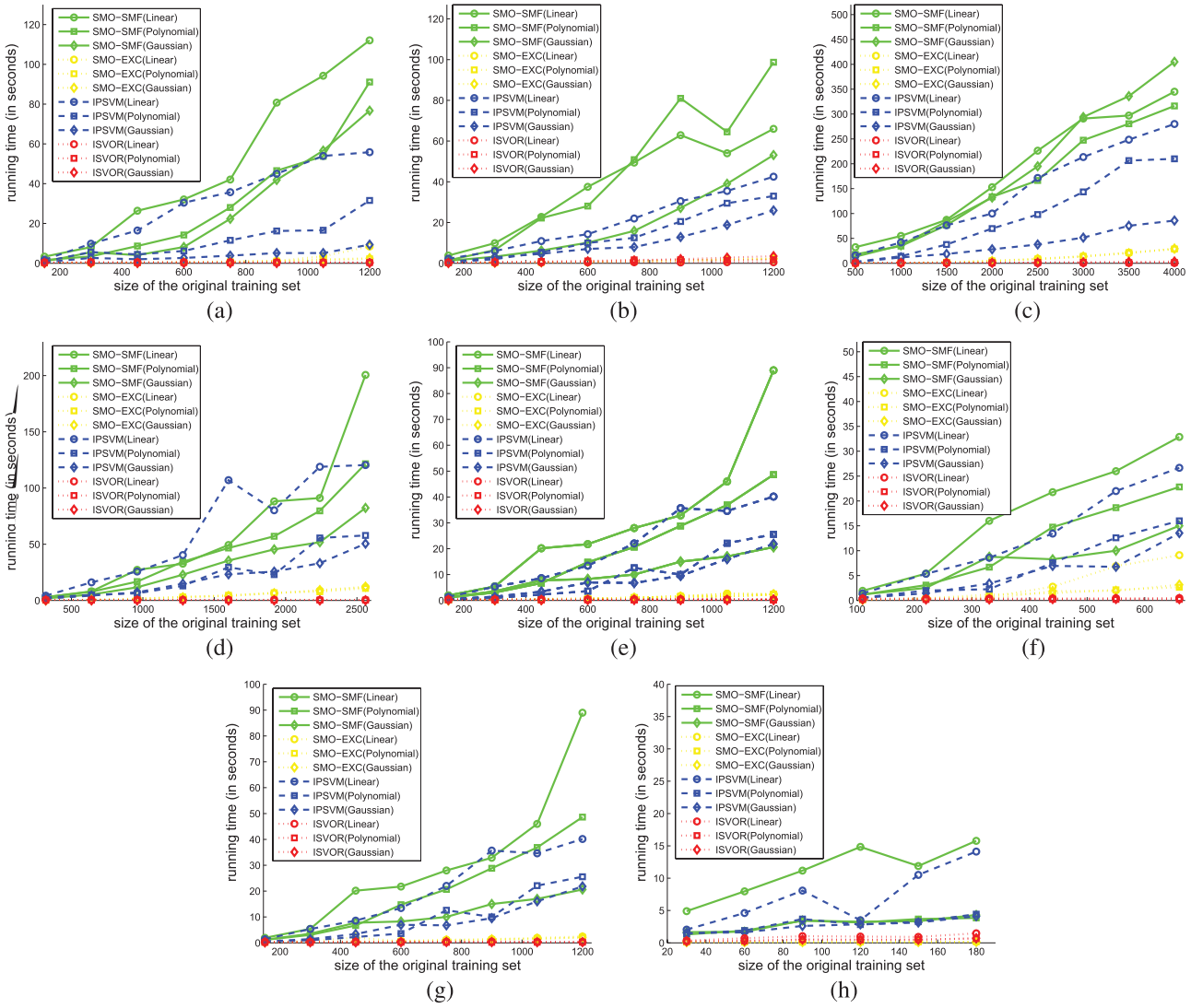


Fig. 6. Running time of SMO-SMF, SMO-EXC, IPSVM and ISVOR (in seconds) on the different data sets. (a) Bank. (b) Computer Activity. (c) Friedman. (d) Census. (e) Abalone. (f) Winequality-red. (g) Winequality-white. (h) Spine Image.

the data set Friedman is available at http://mldata.org/repository/data/viewslug/friedman-datasets-fri_c2_250_5/. Originally, these benchmark data sets are used for metric regression problems. To make them suitable for OR problems, we discretized the target values of the samples into five ordinal quantities using equal interval binning.

2) Real OR Data Sets: Winequality-red and Winequality-white are from the UCI machine learning repository [30]. They are real OR data sets.

The spine image data set was collected by us from the London, Canada. The data set is to diagnose a degenerative disc disease based on Pfirrmann *et al.* [31] grading system (Grade 1 healthy and Grade 5 advanced), depending on five image texture features (including contrast, correlation, energy, homogeneity, and mean signal intensity) quantified from magnetic resonance imaging. The data set contains 350 records, where 20, 137, 90, 82, and 21 records were marked Grade 1, 2, 3, 4, and 5, respectively, by an experienced radiologist.

V. Experimental Results and Discussion

A. Usefulness of ISVOR

1) Existence of the Conflicts and the Singularities: When the size of the original training set is 10, 15, 20, 25, and 30, for each data set, Table IV presents the corresponding numbers of occurrences of Conflict-1 and Conflict-2. From this table, we find that the two kinds of conflicts happen with a high probability on the linear and polynomial kernels, and especially on the Spine Image data set. This is because the lower the dimension of the data or the RKHS, the higher the probability that all the margin support vectors in S_S^{jc} have one and the same label. Thus, it is essential to handle the conflicts during the incremental SVOR learning. Our ISVOR can avoid these conflicts effectively.

Table IV also presents the numbers of occurrences of SC-1 and SC-2 on the eight data sets, where the original training sample size of each data set is also set as 10, 15, 20, 25, and 30, respectively. From this table, we find that SC-2 happens with a higher probability than SC-1 does.

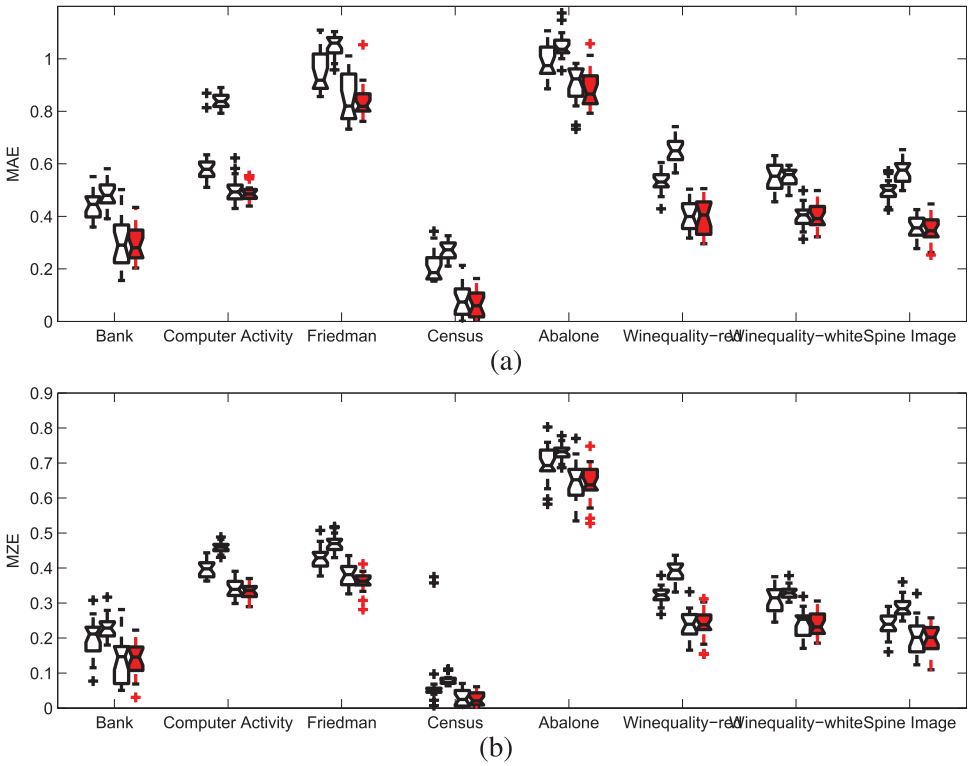


Fig. 7. Results of MAE and MZE, over 20 trials. The grouped boxes represent the results of PSVM, EXC, SMF, and MSMF from left to right on different data sets. The notched-boxes have lines at the lower, median, and upper quartile values. The whiskers are lines extended from each end of the box to the most extreme data value within 1.5 × IQR (Interquartile Range) of the box. Outliers are data with values beyond the ends of the whiskers, which are displayed by plus signs. (a) MAE. (b) MZE.

Although SC-1 happens with a low probability, the possibility of the occurrences still cannot be excluded. Thus, it is very significant that ISVOR handles the two singular cases. Our ISVOR can handle the singularities of $\widetilde{Q}_{\setminus(d'_{\underline{\mathcal{I}}_-})^2}$ and $\widehat{Q}$ effectively.

2) Finite Convergence: We randomly select the samples with the data size shown in the horizontal axis of Fig. 5 for each data set, such as 150, 300, 450, 600, 750, 900, 1050, 1200, and so on, as the original training set, and try to demonstrate the average numbers of the iterations of RAIA, and SRA, when incorporating a new sample into the original training set. Specifically, the average number is obtained by counting the iterations for the increased extended training sample in S_{new}, over 20 trials, regardless of whether whose initial value of the function g is less than 0, or not. Fig. 5 shows the average numbers of iterations of RAIA, and SRA, with different kernels on the different data sets. It is obvious that RAIA and SRA exhibit quick convergence for all data sets and kernels, especially with the Gaussian kernel. Based on Fig. 5, we can conclude that ISVOR avoids the infeasible updating paths as far as possible, and successfully converges to the optimal solution with a fast convergence speed.

B. Comparison With Other Methods

1) Running Time: We randomly select the samples with the data size shown in the horizontal axis of Fig. 6 for each data set as the original training set, similar to the setting in the experiments of finite convergence, and record the running time of SMO-SMF, SMO-EXC, IPSVM, and ISVOR, when incorporating a new sample. Fig. 6 shows the average running time of SMO-SMF, IPSVM, SMO-EXC, and ISVOR with different kernels on the different data sets, over 20 trails. The results clearly show that our ISVOR is generally much faster than SMO-SMF, and IPSVM, on all the data sets and kernels. It should be noted that SMO-EXC is implemented by C. It is inappropriate to compare the running time of the two implementations in different languages directly, as mentioned in Section IV-B. Even so, we still observe that ISVOR is faster than SMO-EXC on the Bank, Friedman, Census, Abalone, Winequality-red, and Winequality-white data sets. If the time of SMO-EXC is multiplied by the ratio between the MATLAB implementation and the C implementation, it would be obvious that ISVOR is much faster than SMO-EXC. To sum up, we can conclude that ISVOR is much faster than SMO-SMF, IPSVM, and SMO-EXC.

2) Generalization Performance: Gaussian kernel is used for comparing the generalization performance. A 5-fold cross validation with a two-step grid search strategy is used to determine the optimal values of model parameters (the Gaussian kernel parameter σ and the regularization factor C) involved in the problem formulations: the initial search is done on a 3×7 coarse grid linearly spaced in the region $\{(\log_{10} C, \log_{10} 2\sigma^2) | 1 \le \log_{10} C \le 3, -3 \le \log_{10} 2\sigma^2 \le 3\}$, followed by a fine search on a 9×9 uniform grid linearly spaced by 0.2 in the $(\log_{10} C, \log_{10} 2\sigma^2)$ space.

We randomly select the samples with the data size shown in Table V for each data set as the validation set. The optimal

TABLE V
SPLITTING OF EACH DATA SET IN THE EXPERIMENTS OF COMPARING THE GENERALIZATION PERFORMANCE

Dataset	Validation set	Training set	Test set
Bank	1,000	2,500	1,000
Computer Activity	2,000	3,192	3,000
Friedman	10,000	20,000	10,000
Census	5,000	12,784	5,000
Abalone	1,000	2,177	1,000
Winequality-red	400	799	400
Winequality-white	1,200	2,498	1,200
Spine Image	85	180	85

parameter values of each formulation for MAE and MZE are computed by the above 5-fold cross validation procedure on the validation set. The remaining samples of each data set are further randomly split into a training set, and a test set, with the data sizes shown in Table V, over 20 trials. For each trial, both MAE and MZE on the test set are computed by a model learned from the training set with the corresponding optimal parameter values computed by the 5-fold cross validation procedure. Fig. 7(a) and (b) shows the MAEs and MZEs of the PSVM, EXC, SMF, and MSMF, respectively, on the different data sets. It is easy to find that MSMF has better accuracy than PSVM and EXC, and is almost as accurate as SMF. Thus, we can conclude that our modified formulation has better accuracy than the existing incremental SVOR algorithm, and is as accurate as the SMF of Shashua and Levin [6].

VI. CONCLUSION

To extend AONSVM to the SVOR formulation (3), we first presented a modified formulation of SVOR based on maximizing sum-of-margins which has multiple constraints of the mixture of an equality and an inequality, then proposed its incremental algorithm. We also provided the finite convergence analysis for it. Numerical experiments showed that the incremental algorithm can converge to the optimal solution in a finite number of steps, and is faster than the existing batch and incremental SVOR algorithms. Meanwhile, the modified formulation has better accuracy than the existing incremental SVOR algorithm, and is as accurate as the SMF of Shashua and Levin [6].

Theoretically, the decremental SVOR learning can also be designed in a similar manner. Because the proposed incremental SVOR algorithm can handle multiple inequality constraints, and can tackle the conflicts between the equality and inequality constraints, we believe that it can be extended to other SVOR formulations of [6] and [8] based on the framework of parametric quadratic programming [32]. In the future, we hope to provide the feasibility analysis for the incremental SVOR algorithm, similar to the work of [29], and extend the incremental SVOR algorithm to multiple kernel learning [33].

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their constructive comments on the early versions of this paper, and Dr. J. Qin for assistance with the data analytics Cloud at Shared Hierarchical Academic Research Computing Network provided through the Southern Ontario Smart Computing Innovation Platform.

REFERENCES

[1] G. Huang, S. Song, C. Wu, and K. You, "Robust support vector regression for uncertain input and output data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 11, pp. 1690–1700, Nov. 2012.

[2] S. Agarwal, "Generalization bounds for some ordinal regression algorithms," in *Proc. 19th Int. Conf. Algorithmic Learn. Theory (ALT)*, Berlin, Germany, 2008, pp. 7–21.

[3] Y. Amit, S. Shalev-Shwartz, and Y. Singer, "Online learning of complex prediction problems using simultaneous projections," *J. Mach. Learn. Res.*, vol. 9, pp. 1399–1435, Jan. 2008.

[4] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge Univ. Press, 2004.

[5] W. Karush, "Minima of functions of several variables with inequalities as side constraints," M.S. thesis, Dept. Math., Univ. Chicago, Chicago, IL, USA, 1939.

[6] A. Shashua and A. Levin, "Taxonomy of large margin principle algorithms for ordinal regression problems," in *Advances in Neural Information Processing Systems 15*. Cambridge, MA, USA: MIT Press, 2002.

[7] G. Cauwenberghs and T. Poggio, "Incremental and decremental support vector machine learning," in *Advances in Neural Information Processing Systems 13*. Cambridge, MA, USA: MIT Press, 2001, pp. 409–415. [Online]. Available: http://www.isn.ucsd.edu/svm/incremental/

[8] W. Chu and S. S. Keerthi, "Support vector ordinal regression," *Neural Comput.*, vol. 19, no. 3, pp. 792–815, 2007. [Online]. Available: http://www.gatsby.ucl.ac.uk/∼chuwei/svor.htm

[9] K. Crammer and Y. Singer, "A new family of online algorithms for category ranking," in *Proc. 25th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, New York, NY, USA, 2002, pp. 151–158.

[10] K. Crammer and Y. Singer, "Online ranking by projecting," *Neural Comput.*, vol. 17, no. 1, pp. 145–175, 2005.

[11] M. Martin, "On-line support vector machine regression," in *Proc. 13th Eur. Conf. Mach. Learn. (ECML)*, London, U.K., 2002, pp. 282–294.

[12] B. Gu, J. Wang, and H. Chen, "On-line off-line ranking support vector machine and analysis," in *Proc. IEEE Int. Joint Conf. Neural Netw. (IJCNN)*, 2008, pp. 1365–1370.

[13] L. Gunter and J. Zhu, "Computing the solution path for the regularized support vector regression," in *Advances in Neural Information Processing Systems 18*. Cambridge, MA, USA: MIT Press, 2005, pp. 483–490.

[14] T. Hastie, S. Rosset, R. Tibshirani, J. Zhu, and N. Cristianini, "The entire regularization path for the support vector machine," *J. Mach. Learn. Res.*, vol. 5, pp. 1391–1415, Oct. 2004.

[15] R. Herbrich, T. Graepel, and K. Obermayer, "Support vector learning for ordinal regression," in *Proc. 9th Int. Conf. Artif. Neural Netw.*, vol. 1. 1999, pp. 97–102.

[16] A. S. Householder, *The Theory of Matrices in Numerical Analysis*. New York, NY, USA: Dover, 1974.

[17] C.-W. Seah, I. W. Tsang, and Y.-S. Ong, "Transductive ordinal regression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 7, pp. 1074–1086, Jul. 2012.

[18] P. Laskov, C. Gehl, S. Krüger, and K. R. Müller, "Incremental support vector learning: Analysis, implementation and applications," *J. Mach. Learn. Res.*, vol. 7, pp. 1909–1936, Jan. 2006.

[19] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2001.

[20] L. Li and H. T. Lin, "Ordinal regression by extended binary classification," in *Advances in Neural Information Processing Systems 19*. Cambridge, MA, USA: MIT Press, 2007, pp. 865–872.

[21] C.-J. Lin, "On the convergence of the decomposition method for support vector machines," *IEEE Trans. Neural Netw.*, vol. 12, no. 6, pp. 1288–1298, Nov. 2001.

[22] M. V. McCrea, H. D. Sherali, and A. A. Trani, "A probabilistic framework for weather-based rerouting and delay estimations within an airspace planning model," *Transp. Res. C, Emerg. Technol.*, vol. 16, no. 4, pp. 410–431, 2008.

[23] J. S. Cardoso and J. F. Pinto da Costa, "Learning to classify ordinal data: The data replication method," *J. Mach. Learn. Res.*, vol. 8, pp. 1393–1429, Jan. 2007.

[24] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, "New support vector algorithms," *Neural Comput.*, vol. 12, no. 5, pp. 1207–1245, 2000.

[25] M. Karasuyama and I. Takeuchi, "Multiple incremental decremental learning of support vector machines," *IEEE Trans. Neural Netw.*, vol. 21, no. 7, pp. 1048–1059, Jul. 2010.

[26] V. Vapnik, *Statistical Learning Theory*. New York, NY, USA: Wiley, 1998.
[27] G. Wang, D.-Y. Yeung, and F. H. Lochovsky, "A kernel path algorithm for support vector machines," in *Proc. 24th Int. Conf. Mach. Learn. (ICML)*, New York, NY, USA, 2007, pp. 951–958.
[28] B. Gu, J. D. Wang, Y. C. Yu, G. S. Zheng, Y. F. Huang, and T. Xu, "Accurate on-line ν-support vector learning," *Neural Netw.*, vol. 27, pp. 51–59, Mar. 2012.
[29] B. Gu and V. S. Sheng, "Feasibility and finite convergence analysis for accurate on-line ν-support vector machine," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 8, pp. 1304–1315, Aug. 2013.
[30] A. Frank and A. Asuncion. (2010). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml
[31] C. W. Pfirrmann, A. Metzdorf, M. Zanetti, J. Hodler, and N. Boos, "Magnetic resonance classification of lumbar intervertebral disc degeneration," *Spine*, vol. 26, no. 17, pp. 1873–1878, 2001.
[32] K. Ritter, "On parametric linear and quadratic programming problems," in *Proc. Int. Congr. Math. Program.*, 1984, pp. 307–335.
[33] X. Xu, I. W. Tsang, and D. Xu, "Soft margin multiple kernel learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 5, pp. 749–761, May 2013.

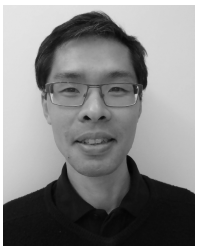
Bin Gu (M'08) received the B.S. and Ph.D. degrees in computer science from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2005 and 2011, respectively.

He joined the School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, in 2010, as a Lecturer, where he is currently an Associate Professor. He is also currently a Post-Doctoral Fellow with the University of Western Ontario, London, ON, Canada. His current research interests include machine learning, data mining, and medical image analysis.

Victor S. Sheng (M'11) received the Ph.D. degree in computer science from the University of Western Ontario, London, ON, Canada, in 2007.

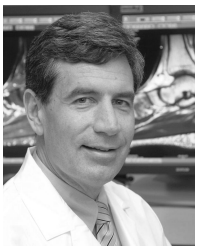
He was an Associate Research Scientist and NSERC Post-Doctoral Follow in Information Systems with Stern Business School, New York University, New York, NY, USA. He is an Assistant Professor with the Department of Computer Science, University of Central Arkansas, Conway, AR, USA, and the founding Director of the Data Analytics Laboratory. His current research interests include data mining, machine learning, and related applications.

Prof. Sheng is a member of the IEEE Computer Society. He is a PC Member for a number of international conferences and a reviewer for several international journals. He was a recipient of the Best Paper Runner-Up Award from the 2008 International Conference on Knowledge Discovery and Data Mining, and the Best Paper Award from the 2011 Industrial Conference on Data Mining.



Keng Yeow Tay received the Medical degree from the University of New South Wales, Sydney, NSW, Australia.

He had his subspecialty fellowship training in diagnostic neuroradiology at Addenbrooke's Hospital, Cambridge, U.K., and the University of Toronto, Toronto, ON, Canada. He is currently an Assistant Professor of Radiology with the Schulich School of Medicine and Dentistry, University of Western Ontario, London, ON. His current research interests include diagnostic neuroradiology, and head and neck radiology.



Walter Romano received the Medical degree from the University of Western Ontario (UWO), London, ON, Canada, and the Fellowship in high-risk obstetrical ultrasound and MRI from the University of Michigan, Ann Arbor, MI, USA.

He is currently an Adjunct Professor of Radiology with the Schulich School of Medicine at UWO. He is the Medical Director of Imaging with the Saint Thomas Elgin General Hospital, Thomas, ON.

Shuo Li received the bachelor's and master's degrees in computer science from Anhui University, Hefei, China, and the Ph.D. degree in computer science from Concordia University, Montreal, QC, Canada.

He is a Research Scientist and Project Manager with General Electric (GE) Healthcare, Mississauga, ON, Canada. He is also an Adjunct Research Professor with the University of Western Ontario, London, ON, Canada, and an Adjunct Scientist with the Lawson Health Research Institute, London. He is currently leading the Digital Imaging Group of London as the Scientific Director. He serves as a Guest Editor and an Associate Editor in prestigious journals in the field. His current research interests include intelligent medical imaging systems, with a main focus on automated medial image analysis and visualization.

Dr. Li's Ph.D. thesis received the Doctoral Prize from Concordia University, which gives to the most deserving graduating student in the faculty of engineering and computer science. He was a recipient of several GE internal awards.