A projection multi-objective SVM method for multi-class classification

A Projection Multi-objective SVM Method for Multi-class Classification

Ling Liu[a], Belén Martín-Barragán[c], Francisco J. Prieto[b]

[a] *College of Applied Science, Beijing University of Technology, 100 Pingleyuan, Chaoyang District, Beijing 100124, P.R. China*
[b] *Department of Statistics, Universidad Carlos III de Madrid, C/ Madrid 126, 28903 Getafe (Madrid), Spain*
[c] *University of Edinburgh Business School, 29 Buccleuch Place, Edinburgh EH8 9JS, United Kingdom*

Abstract

Support Vector Machines (SVMs), originally proposed for classifications of two classes, have become a very popular technique in the machine learning field. For multi-class classifications, various single-objective models and multi-objective ones have been proposed. However, most of the single-objective models consider neither the different costs of different misclassifications nor the users' preferences. Although multi-objective models have taken this drawback into account, they result in large and hard second-order cone programs (SOCPs), from which we get weakly Pareto-optimal solutions. In this paper, we propose a Projected Multi-objective SVM (PM), which is a multi-objective technique that works in a higher dimensional space than the object space. For **PM**, we can characterize the associated Pareto-optimal solutions. Additionally, it significantly alleviates the computational bottlenecks for classifications with large numbers of classes. From our experimental results, we can see **PM** outperforms the single-objective multi-class SVMs (based on an all-together method, one-against-all method and one-against-one method) and other multi-objective SVMs. Compared to the single-objective multi-class SVMs, **PM** provides a wider set of options designed for different misclassification, without sacrificing training time. Compared to other multi-objective methods, **PM** promises the out-of-sample quality of the approximation of the Pareto frontier, with a considerable reduction of the computational burden.

Keywords: Multiple objective programming, Support vector machine, Multi-class multi-objective SVM, Pareto-optimal solution

1. Introduction

Data mining has become a crucial application area in modern science industry and society, due to the growing size of available databases. One of the main applications in this area is supervised classification: to obtain a model that predicts the value of one categorical variable (class) based on the information from other variables. SVM is a popular approach to solve this problem. Cortes and Vapnik (1995) proposed the classical SVM for classifications of two classes. The main idea is

Email addresses: liuling@bjut.edu.cn (Ling Liu), Belen.Martin@ed.ac.uk (Belén Martín-Barragán), fjp@est-econ.uc3m.es (Francisco J. Prieto)

to generate a discriminant hyperplane which separates the input objects. Besides good theoretical properties Vapnik (1998, 2000); Lin (2002), hundreds of applications have shown that SVM can achieve high classification accuracy e.g. Brown et al. (2000); Guyon et al. (2002); Tong and Koller (2002).

In real life, many classification problems involve more than two classes. Researchers have proposed several methods to use SVMs for multi-class classifications. These methods can be roughly grouped into two families. The first family constructs and combines several binary (two classes) classifications, such as one-against-one, one-against-all and directed acyclic graph (DAG) SVMs, e.g. Vapnik (1998); Kreßel (1999); Platt et al. (1999); Hsu and Lin (2002). Alternatively, all-together methods directly find a discriminant function by solving a single optimization problem, which attempts to classify all patterns into the corresponding classes, e.g. Bredensteiner and Bennett (1999); Weston and Watkins (1999); Crammer and Singer (2002); Hsu and Lin (2002).

The aforementioned methods are based on solving single-objective optimization problems. The main drawback of these methods is that, they consider neither the different costs for different misclassifications nor a priori information. This difference is important in many applications. For example, in medical diagnosis, it is known that the cost of misdiagnosing a patient with early gastric cancer as having gastritis is significantly different from the cost of misdiagnosing the patient as healthy. Besides, asymmetries in different misclassifications (the cost of misclassifying a healthy patient as ill is different from that of misclassifying an ill patient as healthy) are widespread. In medical diagnosis, as in many other applications, this asymmetry needs to be considered. For instance, an investor may need an SVM which can separate high volatility shares from low volatility shares as accurately as possible, while it might be acceptable to misclassify some of the low volatility shares as high volatility ones. To overcome this drawback, a simple way is to introduce a weighted single-objective function combining all criteria. These weights are rough indexes for the importances of misclassification costs, although in practice it may be difficult to come up with specific numbers to weigh these importances. An alternative way is using a multi-objective approach.

A bi-objective SVM method for classifications of two classes was proposed in Carrizosa and Martin-Barragan (2006). In that paper, they characterized all the Pareto-optimal solutions to the bi-objective SVM. Tatsumi et al. (2007a) used a multi-objective multi-class SVM method for pattern recognition. Based on all-together, one-against-all and one-against-one methods, they proposed a series of multi-objective SVMs to solve multi-class classification problems, e.g. Ishida et al. (2012); Tatsumi et al. (2007b, 2009, 2010, 2011); Tatsumi and Tanino (2014), using the $\varepsilon-$constraint method. These multi-objective multi-class SVMs have some limitations: First, the $\varepsilon-$constraint method is limited to computing weakly Pareto-optimal solutions Ehrgott (2005). Second, computational cost increases dramatically when we classify a large number of classes. Specifically, these multi-objective SVM methods need to solve a large-scale quadratic programming (QP) problem to select the values of a large number of parameters before applying the $\varepsilon-$constraint method. Consequently, each of these multi-objective approaches needs to solve a large-scale SOCP to get the classifiers. These computational problems result in very slow training procedures, which may not be efficient for large-scale real-life problems. Finally, these models ignore the costs of asymmetries: although they consider that the importance of misclassifications between classes 1 and 2 may be different from those between classes 2 and 3, the importance of misclassifying a class

1 object into class 2 is considered to be equal to that of misclassifying a class 2 object into class 1.

In this paper, we propose a Projected Multi-objective SVM (**PM**), which is a practical multi-objective multi-class SVM that works in a higher dimensional space than the object space. Our aim is to address the main limitations that we have identified in the existing multi-class SVM methods. First, by taking the different misclassification costs (including the costs of asymmetries) into account, we provide **PM** with a greater degree of flexibility and thus with a wider set of options. Second, instead of directly solving the corresponding large-scale multi-objective programming, we characterize the Pareto-optimal solutions to **PM**. Specifically, the Pareto-optimal solutions to **PM** can be constructed based on the optimal solution of a QP which can be decomposed into smaller subproblems in an efficient manner. Thus, the computational burden is significantly reduced. Finally, from our experimental results, we can see that our proposal provides an approximation of the Pareto frontier with high out-of-sample quality, using limited computational cost. As a result, **PM** is both efficient and effective.

This paper is organized as follows: In Section 2, we briefly review some multi-class SVMs including some single-objective and multi-objective approaches. In Section 3, we describe the proposed method **PM** and characterize the Pareto-optimal solutions to it. In Section 4, we describe and comment experimental results showing that **PM** outperforms the state of the art. Finally, conclusions are presented in Section 5.

2. Multi-class classification

In what follows we assume that we have a training set $I = \{x_i\}_{i=1}^k \subseteq \mathbb{R}^l$ corresponding to m ($m \geq 3$) different classes, and let $y_i \in G = \{1, ..., m\}$ denote the class membership of vector x_i. The aim of the multi-class SVMs is to generate decision functions which helps to predict the class memberships of new objects with high accuracy.

For multi-class classification, the all-together method, one-against-all method and one-against-one method are the most commonly used single-objective methods, e.g. Vapnik (1998); Bredensteiner and Bennett (1999); Kreßel (1999); Weston and Watkins (1999); Crammer and Singer (2002); Hsu and Lin (2002). The single-objective all-together method needs to solve a large-scale optimization problem, so it is limited to small data sets Weston and Watkins (1999); Hsu and Lin (2002). The one-against-all method constructs m binary SVMs, where each SVM classifies one of the classes versus the rest. Unbalances associated to the large quantitative difference of class objects in these binary SVMs may affect their classification accuracy and generalization abilities Tatsumi et al. (2010, 2011). In this regard, some experimental results show that the one-against-all method may have a worse accuracy for some problems compared to the all-together and one-against-one methods Hsu and Lin (2002). As suggested in Hsu and Lin (2002), the one-against-one method, which constructs $m(m-1)/2$ classifiers (discriminant hyperplanes), is a more suitable approach for multi-class classification, compared to the all-together and one-against-all methods.

Multi-objective multi-class SVMs have also been proposed for multi-class classifications. Based on the all-together, one-against-all and one-against-one method, a series of multi-objective SVMs were introduced in Tatsumi et al. (2007b, 2009, 2010, 2011); Ishida et al. (2012); Tatsumi and Tanino (2014). These multi-objective SVMs share many common features. They use the same classification rule: $(\omega^p)^T x + b^p$, $p \in G$ is used as a measure of the degree of confidence that object

x belongs to class p. Then, x is assigned to the class with the highest degree of confidence. Also, their $m(m-1)$ objective functions and $(m-1)k$ constraints share the same spirit. In general, the main properties sought in the definition of an SVM-based procedure are a high generalization ability and low training classification errors. In order to achieve a higher generalization ability, geometric margins were used in the previous multi-objective SVM proposals instead of functional ones Tatsumi and Tanino (2014). In addition, they minimized certain penalty functions (defined as weighted proportions of the sums of the auxiliary variables over the geometric margins) instead of each of the auxiliary variables for the sake of high classification accuracy. In this way, they avoided optimizing a very large number of objective functions. As for the constraints, all the objects are required be correctly classified by all the associated discriminant hyperplanes, taking into account the auxiliary variables.

Tatsumi et al. (2009) introduced a multi-objective SVM based on an all-together method (MS2) with $m(l+1)+(m-1)(k+\frac{m}{2})$ variables. To reduce this large number of variables, they introduced two modified versions of this method: SM-OA (a multi-objective SVM based on a one-against-all method) Tatsumi et al. (2010), and M-OAO (a multi-objective SVM based on a one-against-one method) Ishida et al. (2012). Ishida et al. (2012) presented a hard-margin version of this method (misclassification was not allowed), which can be easily extended to a soft-margin version (SM-OAO) that allows some misclassifications. SM-OA (SM-OAO) proceeds in two phases: The first phase applies a one-against-all (one-against-one) method to obtain initial estimates $\bar{\omega}^p$ ($\bar{\omega}^{pq}$) for the parameters $\omega^p,\ p \in G$ ($\omega^{pq},\ q > p,\ p, q \in G$). In the second phase, a problem similar to MS2 is solved, after replacing ω^p with $\alpha^p \bar{\omega}^p$ (or $\sum_{q \neq p, q \in G} \alpha^{pq} \bar{\omega}^{pq}$), $p \in G$. The number of variables in SM-OA is $\frac{1}{2}m(m+3)+(m-1)k$, while for SM-OAO we have $\frac{1}{2}m(3m-1)+(m-1)k$ variables. We can see that SM-OA has the smallest number of variables among these three multi-objective SVMs. And if $l > m-1$, SM-OAO has $m(l+1-m)$ fewer variables than MS2. Thus, both SM-OA and SM-OAO should be more computationally efficient than MS2. However, the optimal coefficients (ω^*, b^*) obtained from SM-OA (SM-OAO) are also feasible to MS2, so their solutions will be no better than those provided by MS2.

The goal of solving multi-objective optimization problems is to identify their Pareto-optimal solutions. Following Deb (2001); Ehrgott (2005); Chinchuluun and Pardalos (2007), we define the Pareto-optimal solutions and weakly Pareto-optimal solutions as follows: Given a general multi-objective problem,

$$\max_{\mu \in C} (f_1(\mu), f_2(\mu), \cdots, f_h(\mu)).$$

- A feasible solution μ^* is Pareto-optimal iff there does not exist another feasible solution $\mu \in C$ such that $f_i(\mu) \geq f_i(\mu^*)$ for all $i \in \{1, 2, \cdots, h\}$, and $f_j(\mu) > f_j(\mu^*)$ for at least one $j \in \{1, 2, \cdots, h\}$.
- A feasible solution μ^* is weakly Pareto-optimal iff there does not exist another feasible solution $\mu \in C$ such that $f_i(\mu) > f_i(\mu^*)$ for all $i \in \{1, 2, \cdots, h\}$.

Practically, it is often difficult and expensive to compute the complete set of Pareto-optimal solutions in many cases (e.g., large-scale optimization problems, complex structure of the Pareto-optimal solutions). Consequently, the most common approach is to build an approximation of the Pareto-optimal solutions based on a limited number of solution values. For the above mentioned multi-objective SVMs Tatsumi et al. (2007b, 2009, 2010, 2011); Ishida et al. (2012); Tatsumi and

Tanino (2014), Tatsumi et al. suggested to use the $\varepsilon-$constraint method to transform the problem into a single-objective one, by selecting one of the objective functions while transforming the rest into constraints by setting limits to their values. As these multi-objective SVMs have $m(m-1)$ objectives, we have to introduce $m(m-1)-1$ parameters to define the constraints. Each of the solutions approximating the Pareto-optimal set would be associated with a given set of values for these parameters. The use of an $\varepsilon-$constraint method introduces significant limitations in the solutions to these multi-objective SVMs. From Ehrgott (2005), we know that the $\varepsilon-$constraint method only guarantees weakly Pareto-optimal solutions. When m is large, to obtain a reasonable approximation of the Pareto-optimal solutions, after the values of the $m(m-1)$ parameters are obtained by solving a number of large-scale QPs (e.g. OS in Tatsumi et al. (2010)), we still need to solve several computationally-expensive large-scale SOCPs (e.g. εSMOA2 in Tatsumi et al. (2010)).

3. Projected multi-objective SVM

For simplicity, we will consider the case of a linear classifier, given that a nonlinear classifier can be considered as a linear one embedded in a richer object space. As discussed in Section 2, among the all-together, one-against-all and one-against-one methods, the most efficient single-objective method for multi-class classification is the one-against-one method. We construct the discriminant hyperplanes as follows:

- The discriminant hyperplane to separate class p data against class q data is given by:

$$L^{pq}: \quad \omega^{pq}x + b^{pq} = 0,\ q > p,\ p, q \in G,$$

  where $\omega^{pq} \in \mathbb{R}^l,\ q > p,\ p, q \in G$ are row vectors.

Ideally, we would like to have all class p objects lying above hyperplane L^{pq}, and all class q objects lying below L^{pq}. If there exist hyperplanes such that the training objects satisfy this ideal situation, we say that the training objects are linearly separable.

Considering the costs of different classification errors and any asymmetries in the misclassification costs, we construct the following single-objective SVM, which tries to classify all the classes simultaneously and is based on using a quadratic loss function:

$$\begin{aligned} \min_{\omega,b,\xi} \quad & \|\omega\|^2 + \sum_{p=1}^{m}\sum_{q\neq p} c^{pq} \sum_{x\in I_p} (\xi_x^{pq})^2, \\ \text{s.t.} \quad & \omega^{pq}x + b^{pq} + \xi_x^{pq} \geq 1, x \in I_p, q > p, p, q \in G, \\ & -\omega^{pq}x - b^{pq} + \xi_x^{qp} \geq 1, x \in I_q, q > p, p, q \in G, \\ & \xi_x^{pq} \geq 0, x \in I_p, q \neq p, p, q \in G, \end{aligned} \tag{1}$$

where $\omega = (\omega^{12}, \omega^{13}, \cdots, \omega^{(m-1)m})$ and $I_p = \{x \in I | x \in \text{class } p\}$. Note that Eq. (1) is separable by pairs of classes. Hence, we have:

Property 3.1. *(ω_*, b_*, ξ_*) is optimal for Eq. (1) if and only if the sub-vectors $(\omega_*^{pq}, b_*^{pq}, \xi_*^{pq}, \xi_*^{qp}),\ q > p,\ p, q \in G$ are optimal for the binary problems:*

$$\min_{\omega^{pq}, b^{pq}, \xi^{pq}, \xi^{qp}} \quad \frac{1}{2}\|\omega^{pq}\|^2 + c^{pq} \sum_{x\in I_p} (\xi_x^{pq})^2 + c^{qp} \sum_{x\in I_q} (\xi_x^{qp})^2,$$

$$
\begin{aligned}
s.t.\quad & \omega^{pq}x + b^{pq} + \xi_x^{pq} \geq 1,\ x \in I_p,\ q > p,\ p,q \in G, \\
& -\omega^{pq}x - b^{pq} + \xi_x^{qp} \geq 1,\ x \in I_q,\ q > p,\ p,q \in G, \\
& \xi_x^{pq} \geq 0,\ x \in I_p,\ q \neq p,\ p,q \in G.
\end{aligned} \tag{2}
$$

However, even if estimates of the misclassification costs are available, it is not easy to find appropriate values for the weights $(c^{pq}, p \neq q, p, q \in G)$ in the objective function of Eq. (1). In practice, decision makers prefer to have different options so that they can choose the most suitable one. This approach implies solving many single-objective problems, and it can be made more efficient if appropriate multi-objective SVMs are used. In this paper, for multi-class classification, we propose an efficient multi-objective model for which we can characterize all its Pareto-optimal solutions. It is based on projecting the object space onto a higher dimensional space, in which we can define the geometric margins in a tractable way. The projection of interest is defined as:

$$
\Delta_x^{pq} = \left((\delta_x^{12})^T, (\delta_x^{13})^T, \cdots, (\delta_x^{(m-1)m})^T\right)^T,\ p < q,\ p,q \in G,\ \text{with } \delta_x^{ij} = \begin{cases} x, & \text{if } (i,j) = (p,q); \\ 0, & \text{otherwise.} \end{cases}
$$

Note that we have $\omega\Delta_x^{pq} = \omega^{pq}x$.

We are interested in a SVM formulation whose objective functions integrate both the maximization of the geometric margins and the minimization of the classification errors. We incorporate the classification errors into our model by redefining the geometric margins after embedding the slack variables (as measures of misclassification) into them. We consider the following additional projection:

$$
\Delta_{\xi x}^{pq} = \left((\delta_{\xi x}^{12})^T, (\delta_{\xi x}^{21})^T, \cdots, (\delta_{\xi x}^{m(m-1)})^T\right)^T,\ q > p,\ p,q \in G,
$$

where

$$
\delta_{\xi x}^{ij} = \begin{cases} \frac{1}{\sqrt{c^{pq}}}e_n, & \text{if } (i,j) = (p,q) \text{ and } x \text{ is the } n\text{-th object in class } p, \\ 0, & \text{if } (i,j) \neq (p,q),\ i \neq j,\ i,j \in G, \end{cases}
$$

and e_n is the n-th unit vector.

In the projected space we can construct the hyperplane classifying class p objects against class q objects as:

$$
PL^{pq} : \left(\omega, \sqrt{C}\xi\right)\left((\Delta_x^{pq})^T, (\Delta_{\xi x}^{pq})^T\right)^T + b^{pq} = 0,\ q > p,\ p,q \in G,
$$

where, $\sqrt{C}\xi = \left(\sqrt{c^{12}}\xi^{12}, \sqrt{c^{21}}\xi^{21}, \cdots, \sqrt{c^{m(m-1)}}\xi^{m(m-1)}\right)$ and $\xi^{pq},\ q \neq p,\ p,q \in G$ is the row vector which contains all the $\xi_x^{pq}, x \in I_p$.

We define the geometric margin from object x to hyperplane L^{pq} as the Euclidean distance from point $\Gamma_x^{pq} = \left((\Delta_x^{pq})^T, (\Delta_{\xi x}^{pq})^T\right)^T$ to hyperplane PL^{pq} in the projected space.

- The geometric margin from object $x \in I_p$ to hyperplane L^{pq} is:

$$
\varrho_x^{pq}(\omega, \sqrt{C}\xi, b) = \frac{\left|\left(\omega, \sqrt{C}\xi\right)\left((\Delta_x^{pq})^T, (\Delta_{\xi x}^{pq})^T\right)^T + b^{pq}\right|}{\left\|\left(\omega, \sqrt{C}\xi\right)\right\|} = \frac{\omega^{pq}x + \xi^{pq} + b^{pq}}{\left\|\left(\omega, \sqrt{C}\xi\right)\right\|},\ q > p,\ p,q \in G,
$$

  while for $q < p$ we have $\varrho_x^{pq}(\omega, \sqrt{C}\xi, b) = (-\omega^{qp}x + \xi^{pq} - b^{qp})/\|(\omega, \sqrt{C}\xi)\|,\ p,q \in G$.
- The geometric margin for class p objects with respect to hyperplane L^{pq} is:

$$
\varrho^{pq}(\omega, \sqrt{C}\xi, b) = \min_{x \in I_p} \varrho_x^{pq}(\omega, \sqrt{C}\xi, b),\ p \neq q,\ p,q \in G.
$$

We maximize all the geometric margins $\varrho^{pq},\ q \neq p,\ p,q \in G$ by solving the following multi-objective problem:

$$\begin{aligned} \max_{\omega,b,\xi} \quad & \left(\varrho^{12}(\omega,\sqrt{C}\xi,b), \varrho^{21}(\omega,\sqrt{C}\xi,b), \cdots, \varrho^{(m-1)m}(\omega,\sqrt{C}\xi,b), \varrho^{m(m-1)}(\omega,\sqrt{C}\xi,b)\right), \\ \text{s.t.} \quad & \omega^{pq}x + b^{pq} + \xi_x^{pq} > 0,\ x \in I_p,\ q > p,\ p,q \in G, \\ & -\omega^{pq}x - b^{pq} + \xi_x^{qp} > 0,\ x \in I_q,\ q > p,\ p,q \in G, \\ & \xi_x^{pq} \geq 0,\ x \in I_p,\ p \neq q,\ p,q \in G. \end{aligned} \tag{3}$$

We refer to the above multi-objective optimization problem Eq. (3) as **PM** (Projected Multi-objective SVM). As in Hsu and Lin (2002), we use majority voting (also known as 'Max Wins') to define our classification rule: For object x, if $\omega^{pq}x + b^{pq} > 0$, then the vote for the p-th class is increased by one. Otherwise, the vote for the q-th class is increased by one. After this procedure is completed, x is assigned to the class with the largest vote. In the case that two classes have identical votes, the one with smaller index is selected.

To find the Pareto-optimal solutions to Eq. (3), we define the following single-objective minimax weighted problem:

$$\begin{aligned} \max_{\omega,b,\xi} \min \quad & \left(\varrho^{12}(\omega,\sqrt{C}\xi,b), \theta^{21}\varrho^{21}(\omega,\sqrt{C}\xi,b), \cdots, \theta^{m(m-1)}\varrho^{m(m-1)}(\omega,\sqrt{C}\xi,b)\right), \\ \text{s.t.} \quad & \omega^{pq}x + b^{pq} + \xi_x^{pq} > 0, x \in I_p, q > p, p,q \in G, \\ & -\omega^{pq}x - b^{pq} + \xi_x^{qp} > 0, x \in I_q, q > p, p,q \in G, \\ & \xi_x^{pq} \geq 0, x \in I_p, q \neq p, p,q \in G. \end{aligned} \tag{4}$$

Eq. (4) is an intermediate step in the computation of the weakly Pareto-optimal solutions to **PM**, where the values θ^{pq} can be seen as the proportions of the geometric margin ϱ^{12} over each of the geometric margins ϱ^{pq}. The following lemma establishes the relationship between Eq. (4) and **PM**.

Lemma 3.2. *(1) The optimal solution of Eq. (4) is weakly Pareto-optimal to **PM**;*
*(2) Every weakly Pareto-optimal solution of **PM** is optimal to Eq. (4), for some specific values of* $\boldsymbol{\theta} = (\theta^{21}, \cdots, \theta^{(m-1)m}, \theta^{m(m-1)}) > 0$.

The proof can be found in AppendixA. The close relationship between Eq. (4) and **PM** promises us an efficient way to find the set of weakly Pareto-optimal solutions for **PM**. Specifically, we analyze the form of the optimal solutions to Eq. (4) rather than directly solving **PM**. The following theorem describes the set of weakly Pareto-optimal solutions to **PM**.

Theorem 3.3. *The set of weakly Pareto-optimal solutions for **PM** is :*

$$\left\{(\boldsymbol{\omega},\boldsymbol{b},\boldsymbol{\xi}) = (\mu\omega_\theta, \mu b_\theta, \mu\xi_\theta) \mid \mu > 0,\ \theta^{pq} > 0,\ p < q,\ p,q \in G\right\},$$

where $\theta^{12} = 1$, $\omega_\theta^{pq} = \frac{\theta^{pq}+\theta^{qp}}{2\theta^{pq}\theta^{qp}}\omega_1^{pq}$, $b_\theta^{pq} = \frac{\theta^{qp}-\theta^{pq}}{2\theta^{pq}\theta^{qp}} + \frac{\theta^{pq}+\theta^{qp}}{2\theta^{pq}\theta^{qp}}b_1^{pq}$ *and* $\xi_\theta^{pq} = \frac{\theta^{pq}+\theta^{qp}}{2\theta^{pq}\theta^{qp}}\xi_1^{pq}$, *for all* $q > p,\ p,q \in G$, *with* (ω_1, b_1, ξ_1) *being an optimal solution for Eq. (1).*

Proof. Using Lemma 3.2, we only need to prove that the optimal solutions to Eq. (4) have the form $(\mu\omega_\theta, \mu b_\theta, \mu\xi_\theta),\ \mu > 0$. First, using the definition of geometric margins, we can rewrite Eq. (4) as:

$$\begin{aligned}
\min_{\omega,b,\xi}\ & \frac{\left\|\left(\omega,\sqrt{C}\xi\right)\right\|}{\min\left\{\min\limits_{x\in I_1}[\omega^{12}x+b^{12}+\xi_x^{12}],\cdots,\theta^{m(m-1)}\min\limits_{x\in I_m}[-\omega^{(m-1)m}x-b^{(m-1)m}+\xi_x^{m(m-1)}]\right\}},\\
\text{s.t.}\ & \omega^{pq}x+b^{pq}+\xi_x^{pq}>0, x\in I_p,\ q>p,\ p,q\in G,\\
& -\omega^{pq}x-b^{pq}+\xi_x^{qp}>0, x\in I_q,\ q>p,\ p,q\in G,\\
& \xi_x^{pq}\geq 0, x\in I_p,\ p\neq q,\ p,q\in G.
\end{aligned}\tag{5}$$

As the objective function of Eq. (5) is homogeneous in (ω,b,ξ), we can standardize the value of denominator of the objective function, and solve the following problem to get the optimal solution to Eq. (5):

$$\begin{aligned}
\min_{\omega,b,\xi}\ & \left\|\left(\omega,\sqrt{C}\xi\right)\right\|,\\
\text{s.t.}\ & \min\left\{\min_{x\in I_1}[\omega^{12}x+b^{12}+\xi_x^{12}],\cdots,\theta^{m(m-1)}\min_{x\in I_m}[-\omega^{(m-1)m}x-b^{(m-1)m}+\xi_x^{m(m-1)}]\right\}=1,\\
& \xi_x^{pq}\geq 0,\ x\in I_p,\ q\neq p,\ p,q\in G.
\end{aligned}\tag{6}$$

Eq. (6) is equivalent to

$$\begin{aligned}
\min_{\omega,b,\xi}\ & \left\|\left(\omega,\sqrt{C}\xi\right)\right\|,\\
\text{s.t.}\ & \theta^{pq}[\omega^{pq}x+b^{pq}+\xi_x^{pq}]\geq 1,\ x\in I_p,\ q>p,\ p,q\in G,\\
& \theta^{qp}[-\omega^{pq}x-b^{pq}+\xi_x^{qp}]\geq 1,\ x\in I_q,\ q>p,\ p,q\in G,\\
& \xi_x^{pq}\geq 0,\ x\in I_p,\ q\neq p,\ p,q\in G,
\end{aligned}\tag{7}$$

and to its quadratic version:

$$\begin{aligned}
\min_{\omega,b,\xi}\ & \left\|\left(\omega,\sqrt{C}\xi\right)\right\|^2,\\
\text{s.t.}\ & \theta^{pq}[\omega^{pq}x+b^{pq}+\xi_x^{pq}]\geq 1,\ x\in I_p,\ q>p,\ p,q\in G,\\
& \theta^{qp}[-\omega^{pq}x-b^{pq}+\xi_x^{qp}]\geq 1,\ x\in I_q,\ q>p,\ p,q\in G,\\
& \xi_x^{pq}\geq 0,\ x\in I_p,\ q\neq p,\ p,q\in G.
\end{aligned}\tag{8}$$

As the objective function of Eq. (8) is strictly convex for (ω,ξ), its optimal solution $(\omega_\theta,\xi_\theta)$ is unique. Besides, as the objective of Eq. (8) is quadratic (positive semidefinite) and the constraints are affine functions, its KKT conditions are necessary and sufficient for optimality. These KKT conditions for Eq. (8) are:

$$\begin{aligned}
& 2\omega_\theta^{pq}=\theta^{pq}\sum_{x\in I_p}\lambda_{\theta x}^{pq}x^T-\theta^{qp}\sum_{x\in I_q}\lambda_{\theta x}^{qp}x^T,\ q>p,\ p,q\in G,\\
& \sum_{x\in I_p}\theta^{pq}\lambda_{\theta x}^{pq}-\theta^{qp}\sum_{x\in I_q}\lambda_{\theta x}^{qp}=0,\ q>p,\ p,q\in G,\\
& 2c^{pq}\xi_{\theta x}^{pq}=\theta^{pq}\lambda_{\theta x}^{pq}+\tau_{\theta x}^{pq},\ x\in I_p,\ p\neq q,\ p,q\in G,\\
& \lambda_{\theta x}^{pq}[\theta^{pq}\omega_\theta^{pq}x+\theta^{pq}b_\theta^{pq}+\theta^{pq}\xi_{\theta x}^{pq}-1]=0,\ x\in I_p,\ q>p,\ p,q\in G,\\
& \lambda_{\theta x}^{qp}[-\theta^{qp}\omega_\theta^{pq}x-\theta^{qp}b_\theta^{pq}+\theta^{qp}\xi_{\theta x}^{qp}-1]=0,\ x\in I_q,\ q>p,\ p,q\in G,\\
& \xi_{\theta x}^{pq}\geq 0,\lambda_{\theta x}^{pq}\geq 0,\tau_{\theta x}^{pq}\geq 0,\ x\in I_p,\ p\neq q,\ p,q\in G,
\end{aligned}$$

$$\theta^{pq}[\omega_\theta^{pq}x + b_\theta^{pq} + \xi_{\theta x}^{pq}] \geq 1,\ x \in I_p,\ q > p,\ p, q \in G,$$
$$\theta^{qp}[-\omega_\theta^{pq}x - b_\theta^{pq} + \xi_{\theta x}^{qp}] \geq 1,\ x \in I_q,\ q > p,\ p, q \in G,$$
$$\tau_{\theta x}^{pq}[-\xi_{\theta x}^{pq}] = 0,\ x \in I_p,\ q \neq p,\ p, q \in G.$$

From these KKT conditions, we can see that $(\lambda_\theta^{pq}, \lambda_\theta^{qp}) \neq 0,\ q > p,\ p, q \in G$. Without loss of generality we have that, for each $p, q \in G$ with $q > p$, there exists some $x_\theta^{pq} \in I_p$ such that $\lambda_{\theta x_\theta^{pq}}^{pq} \neq 0$. Then we get

$$b_\theta^{pq} = \frac{1}{\theta^{pq}} - \omega_\theta^{pq} x_\theta^{pq} - \xi_{\theta x_\theta^{pq}}^{pq},\ q > p,\ p, q \in G. \tag{9}$$

Thus, the set of optimal solutions for Eq. (8) is nonempty. From the uniqueness of $(\omega_\theta, \xi_\theta)$ and Eq. (9), we have that Eq. (8) has a unique optimal solution. Also, note that for $\theta = (1, \ldots, 1)$, problems Eq. (8) and Eq. (1) coincide.

Suppose (ω_1, b_1, ξ_1) is optimal to Eq. (1) and (λ_1, τ_1) are the corresponding KKT multiplier vectors. Then letting

$$\begin{aligned}
\omega_\theta^{pq} &= \frac{\theta^{pq} + \theta^{qp}}{2\theta^{pq}\theta^{qp}}\omega_1^{pq},\ q > p,\ p, q \in G,\\
b_\theta^{pq} &= \frac{\theta^{qp} - \theta^{pq}}{2\theta^{pq}\theta^{qp}} + \frac{\theta^{qp} + \theta^{pq}}{2\theta^{pq}\theta^{qp}}b_1^{pq},\ q > p,\ q, p \in G,\\
\xi_{\theta x}^{pq} &= \frac{\theta^{pq} + \theta^{qp}}{2\theta^{pq}\theta^{qp}}\xi_{1x}^{pq},\ x \in I_p,\ p \neq q,\ p, q \in G,\\
\lambda_{\theta x}^{pq} &= \frac{\theta^{pq} + \theta^{qp}}{2\theta^{pq}\theta^{qp}}\frac{1}{\theta^{pq}}\lambda_{1x}^{pq},\ x \in I_p,\ p \neq q,\ p, q \in G,\\
\tau_{\theta x}^{pq} &= \frac{\theta^{pq} + \theta^{qp}}{2\theta^{qp}\theta^{pq}}\tau_{1x}^{pq},\ x \in I_p,\ p \neq q,\ p, q \in G,
\end{aligned} \tag{10}$$

it holds that $(\omega_\theta, b_\theta, \xi_\theta)$ is the unique optimal solution of Eq. (8), since it satisfies the KKT conditions. Then, for any $\mu > 0$ we have that $(\mu\omega_\theta, \mu b_\theta, \mu\xi_\theta)$ is optimal to Eq. (4). □

We now show that these weakly Pareto-optimal solutions are also Pareto-optimal to **PM**.

Corollary 3.4. *The Pareto-optimal solution set of **PM** will be given by:*

$$\{(\boldsymbol{\omega}, \boldsymbol{b}, \boldsymbol{\xi}) = (\mu\omega_\theta, \mu b_\theta, \mu\xi_\theta) \mid \mu > 0,\ \theta^{pq} > 0,\ p < q,\ p, q \in G\},$$

where $\theta^{12} = 1$, $\omega_\theta^{pq} = \frac{\theta^{pq}+\theta^{qp}}{2\theta^{pq}\theta^{qp}}\omega_1^{pq}$, $b_\theta^{pq} = \frac{\theta^{qp}-\theta^{pq}}{2\theta^{pq}\theta^{qp}} + \frac{\theta^{pq}+\theta^{qp}}{2\theta^{pq}\theta^{qp}}b_1^{pq}$ *and* $\xi_\theta^{pq} = \frac{\theta^{pq}+\theta^{qp}}{2\theta^{pq}\theta^{qp}}\xi_1^{pq}$, *for all* $q > p,\ p, q \in G$, *with* (ω_1, b_1, ξ_1) *being optimal to Eq. (1).*

Proof. We need to prove that the weakly Pareto-optimal solutions to **PM** are also Pareto-optimal.

Let (ω_*, b_*, ξ_*) be a weakly Pareto-optimal solution to **PM**. Then, there exist some $\theta > 0$ and $\mu > 0$ such that $(\mu\omega_*, \mu b_*, \mu\xi_*)$ will be optimal to Eq. (8). Suppose (ω_*, b_*, ξ_*) is not Pareto-optimal to **PM**. For any $\mu > 0$, we have $\varrho^{pq}(\omega, \sqrt{C}\xi, b) = \varrho^{pq}(\mu\omega, \sqrt{C}\mu\xi, \mu b)$. So $(\mu\omega_*, \mu b_*, \mu\xi_*), \forall \mu > 0$ will not be Pareto-optimal to **PM** either. Then, there exists (ω_0, b_0, ξ_0), feasible to **PM**, such that:

$$\varrho^{pq}(\omega_0, \sqrt{C}\xi_0, b_0) \geq \varrho^{pq}(\mu\omega_*, \sqrt{C}\mu\xi_*, \mu b_*),\ p \neq q,\ p, q \in G, \tag{11}$$

with at least one $(i, j),\ i \neq j,\ i, j \in G$, such that $\varrho^{ij}(\omega_0, \sqrt{C}\xi_0, b_0) > \varrho^{ij}(\mu\omega_*, \sqrt{C}\mu\xi_*, \mu b_*)$.

Without loss of generality, we can take $\|(\omega_0, \sqrt{C}\xi_0)\| = \|\mu(\omega_*, \sqrt{C}\xi_*)\|$. We have from Eq. (11):

$$\omega_0^{pq}x + b_0^{pq} + \xi_{0x}^{pq} \geq \mu\omega_*^{pq}x + \mu b_*^{pq} + \mu\xi_{*x}^{pq},\ x \in I_p,\ p \neq q,\ p, q \in G. \tag{12}$$

From Eq. (12) and $(\mu\omega_*, \mu b_*, \mu\xi_*)$ satisfying the constraints of Eq. (8), we have that (ω_0, b_0, ξ_0) is feasible to Eq. (8). As $\|(\omega_0, \sqrt{C}\xi_0)\| = \|(\mu\omega_*, \sqrt{C}\xi_*)\|$, it follows that (ω_0, b_0, ξ_0) is optimal to Eq. (8). Since Eq. (8) has a unique optimal solution, then $\omega_0 = \mu\omega_*,\ b_0 = \mu b_*,\ \xi_0 = \mu\xi_*$. Thus, we have:

$$\varrho^{pq}(\omega_0, \sqrt{C}\xi_0, b_0) = \varrho^{pq}(\mu\omega_*, \sqrt{C}\xi_*, \mu b_*),\ \forall p \neq q,\ p, q \in G.$$

This contradicts our assumption that Eq. (11) holds with at least one strict inequality. We then conclude that (ω_*, b_*, ξ_*) is Pareto-optimal to **PM**. □

From Corollary 3.4, we can see that the Pareto-optimal solutions to **PM** are based on the solutions to the quadratic optimization problem Eq. (1). This problem has the added advantage of being decomposable into binary classification problems, with the corresponding computational advantages when the problem size increases.

4. Computational experiments

To gauge the performance of the proposed PM, we have conducted several experiments. We provide comparison with several alternative multi-class SVMs that have been described in the literature. These experiments have been conducted on the following datasets: IRIS, WINE, SEEDS, VEHICLE, CAR (Car Evaluation), GLASS, SCC (Synthetic Control Chart Time Series) and CTG (Cardiotocography, raw data). All of them are available in the UCI Machine Learning Repository. A summary of the information for these data sets is listed in Table 1.

Table 1. Data set description

Data set	size of the data set	No. of Dim.	No. of classes
IRIS	150	4	3
WINE	178	13	3
SEEDS	210	7	3
VEHICLE	846	18	4
CAR	1728	16	4
GLASS	214	9	6
SCC	600	60	6
CTG	2126	35	10

The first group of experiments compares **PM** with other multi-class SVMs (both the single-objective and multi-objective methods) in terms of their training classification accuracy, testing classification accuracy and training time. We consider that one method is superior to another when it has higher accuracy and lower computational costs. In this paper, we compare the performances of the all-together method (AT)Vapnik (1998), one-against-all method (OAA)Hsu and Lin (2002), one-against-one method (OAO)Kreßel (1999), MS2, SM-OA, SM-OAO and **PM**.

Table 2 shows these measures averaged over 100 replications of random splittings of the dataset into a training sample (80%) used to compute the classifiers and a testing sample (20%) used to compute their testing accuracy.

We have chosen the parameters required by the different methods in the following way: for AT, OAA and OAO, we set the trade-off parameter $c = 1$; for MS2, SM-OA and SM-OAO, we take

$c^{rs} = 10,\ (r,s) = (1,2)$ and fix (ε^{-rs}, μ) as suggested in Tatsumi et al. (2007b, 2009, 2010, 2011); Tatsumi and Tanino (2014); for **PM**, we take $c^{pq} = 1,\ q \neq p,\ p,q \in G$. For every replication of each dataset, we solve all the SVM methods (AT, OAA, OAO, MS2, SM-OA and SM-OAO) once and record the corresponding training classification accuracy, testing classification accuracy and training time. For **PM**, we solve Eq. (1) once and choose the best performance (accuracy) from 100 Pareto-optimal solutions to **PM** obtained randomly using Corollary 3.4.

Table 2. Mean results to compare the performances of the multi-class SVMs

		AT	OAA	OAO	MS2	SM-OA	SM-OAO	**PM**
IRIS	tr.ac	0.9859	0.9513	0.9871	0.6667	0.9845	0.9838	**0.9902**
	te.ac	0.9773	0.9387	0.9753	0.6667	0.9750	0.9723	**0.9870**
	tr.t(s)	**1.0001**	3.0293	3.0435	2.0449	4.0132	4.0770	*1.2147*
WINE	tr.ac	0.9865	0.9965	0.9959	0.8910	0.9978	0.5122	**0.9995**
	te.ac	0.9415	0.9594	0.9500	0.8718	0.9568	0.4982	**0.9741**
	tr.t(s)	**1.7463**	3.9034	3.8853	3.0276	5.1931	5.3379	*1.8346*
SEEDS	tr.ac	0.9518	0.9416	0.9360	0.9412	0.9421	0.8978	**0.9570**
	te.ac	0.9310	0.9255	0.9150	0.9112	0.9212	0.8886	**0.9493**
	tr.t(s)	**0.6690**	1.9245	1.8638	1.6396	2.4636	2.6796	*0.6835*
VEHICLE	tr.ac	0.8404	0.8208	0.8481	0.8392	0.8106	0.6585	**0.8525**
	te.ac	0.7984	0.7896	0.7859	0.8014	0.7789	0.6436	**0.8016**
	tr.t(s)	**2.9827**	9.3945	10.2526	24.7038	12.2850	12.4967	*3.4189*
CAR	tr.ac	0.8910	0.8548	0.9047	0.8829	0.8744	0.8711	**0.9072**
	te.ac	0.8845	0.8463	0.8959	0.8788	0.8669	0.8681	**0.9059**
	tr.t(s)	**0.6279**	1.4334	1.9986	37.3476	2.5928	2.9248	*1.1151*
GLASS	tr.ac	0.6807	0.6478	0.6866	0.6081	0.6366	0.3556	**0.7007**
	te.ac	0.6387	0.5810	0.6302	0.5665	0.5972	0.3375	**0.6617**
	tr.t(s)	**0.8233**	4.0339	9.7277	2.6385	4.8015	10.4632	*1.0687*
SCC	tr.ac	1	0.9900	1	0.51	1	0.9907	**1**
	te.ac	0.9827	0.9353	0.9865	0.5037	0.9426	0.9549	**0.9926**
	tr.t(s)	15.0726	5.0784	17.1662	25.7054	6.1961	13.8103	**2.2722**
CTG	tr.ac	1	1	1	1	1	0.6770	**1**
	te.ac	0.9999	**1**	0.9743	0.9997	0.9999	0.6717	*0.9821*
	tr.t(s)	**14.0502**	15.9397	55.0768	6410.388	41.3981	91.1074	18.1388

Note: The results marked in bold are the best average results. The results marked in italics are the second best average results, when they have been obtained by **PM**. In the following tables, we use the same way to highlight the results.

[1] tr.ac= training classification accuracy,
[2] te.ac= testing classification accuracy,
[3] tr.t(s)= training time measured in seconds.

Table 2 shows that **PM** outperforms the other models for most of the data sets. In particular, **PM** always shows the best mean training classification accuracy. And except for the CTG dataset, **PM** also gives the best mean testing classification accuracy. Even for the CTG dataset, **PM** achieves the second-best mean testing classification accuracy. In terms of the mean training time, **PM** requires only slightly more time than the best value, which is obtained based on the single-objective method AT. For the SCC dataset, **PM** takes the shortest time.

Nevertheless, as our main goal is to find a reasonably detailed representation of the set of all Pareto-optimal solutions to the classification problems, these measures are not the ones most

suitable for comparing the performances of the different multi-objective SVMs. We have already mentioned that for MS2, SM-OA and SM-OAO, we can obtain the weakly Pareto-optimal solutions by using the $\varepsilon-$constraint method, while for **PM**, we are able to provide a characterization for its Pareto-optimal solutions corresponding to a specific set of parameters. In both cases it is too expensive to determine all the Pareto-optimal solutions, as the structure of this set is very complex, particularly for high dimensions. Our aim is to find a good and efficient approximation of the Pareto-optimal solution sets.

In the second group of experiments, we compare different multi-objective methods in terms of the quality of their approximations of the Pareto-optimal solution sets: we say that a method outperforms another when it approximates the Pareto-optimal solution set better than the other. In this paper, we use the epsilon and hypervolume indicators, defined in terms of the test accuracy values as objectives to measure the performance of these multi-objective SVMs. These indicators have the important property of being Pareto compliant (whenever an approximation set A is preferable to B with respect to weak Pareto dominance, the indicator value for A should be at least as good as the indicator value for B Fonseca et al. (2005). Following Fonseca et al. (2005),

- The hypervolume indicator $I_H(A)$ calculates the proportion of the objective space that is weakly dominated by an approximation set of Pareto-optimal solutions A.
- The epsilon indicator is defined as $I_{\epsilon+} = \inf_{\epsilon\in\mathbb{R}}\{\forall z^2 \in R, \exists z^1 \in A \text{ such that } z^1 \preceq_{\epsilon+} z^2\}$, where R is an reference set.

For the hypervolume indicator, we take the objective space as the hypercube which contains all possible testing classification accuracy $(a_1, a_2, \ldots, a_m)$, where a_i is the testing classification accuracy of class i. Based on the definition of the hypervolume indicator, it holds that the values of the hypervolume indicators will be in $[0, 1]$, and the method which has the largest hypervolume indicator (closest to 1) outperforms the others.

For the epsilon indicator, we select the reference set R as $\{(1, \ldots, 1)\} \subset \mathbb{R}^m$, which corresponds to the ideal test accuracy. From the definition of the epsilon indicator, it holds that the values of the epsilon indicators will be in $[-1, 0]$ and the method which has the largest value of its epsilon indicator (closest to 0) outperforms the others.

To obtain a more stable performance measure, we have applied the procedure described below; in each replication we have selected a different subset of 80% of our observations as a training sample and the remaining observations as our testing sample. We have used Matlab R2014a and Mosek 7 to solve the optimization problems. As Mosek sometimes gives us 'unknown' solutions, we have only kept the 'optimal' and 'near-optimal' solutions and discarded the other results, to ensure the reliability of the results.

Step 1: For $i = 1, \cdots, 50$, repeat:
- Step 1.1: Arrange the objects in a random order. Choose the last 20% objects as the testing objects and leave the rest as the training objects.
- Step 1.2: Do:
  - 1.2.a: If the method used is MS2, SM-OA or SM-OAO, then
    - Step 1.2.a1: As in Tatsumi et al. (2007b, 2009, 2010, 2011); Tatsumi and Tanino (2014), obtain the parameters $(\varepsilon_0^{-rs}, \mu_0)$ by solving corresponding single-objective SVMs such as OS in Tatsumi et al. (2010).
    - Step 1.2.a2: Let $(r, s) = (1, 2)$ and $c = 10$. Choose uniformly 100 different values

for (ε^{-rs}, μ) with ε^{-rs} from $[0.1\varepsilon_0^{-rs}, 2\varepsilon_0^{-rs}]$ and μ from $[\mu_0 + 0.01, \mu_0 + 50]$. Then, solve 100 SOCPs such as εSMOA2 Tatsumi et al. (2010) defined from each value of (ε^{-rs}, μ). Keep the 'optimal' and 'near-optimal' solutions. [1]

∗ 1.2.b: If the method used is **PM**, then

· Step 1.2.b1: Use a 10-fold cross validation method to choose the values c^{pq}, $q \neq p$, $p, q \in G$, in the objective function of (1), and compute the corresponding optimal solution (ω_1, b_1).

· Step 1.2.b2: Generate 100 uniform random values z^{pq} in $(0, 1)$ for any $p \neq q$, $p, q \in G$, and take $\theta^{pq} = \frac{z^{12}}{z^{pq}}$. [2] Then use Corollary 3.4 with (ω_1, b_1) to obtain 100 Pareto-optimal solutions to **PM**.

– Step 1.3: Calculate the testing accuracy set based on the solutions obtained from Step 1.2 for each of these multi-objective approaches.

– Step 1.4: From the sets of testing classification accuracy, calculate the corresponding hypervolume and epsilon indicator values for each multi-objective method.

Step 2: Calculate a statistical summary for these epsilon and hypervolume indicators.

Note that to obtain an indicator for **PM** we only need to solve the single-objective SVM (1) once, while for MS2, SM-OA or SM-OAO we have to solve 100 SOCPs and 50 QPs, and not all of them are guaranteed to provide a solution. Additionally, with **PM** we get approximation sets, each composed of exactly 100 testing classification accuracy vectors. Consequently, **PM** provides a richer approximation in a shorter time, compared with MS2, SM-OA and SM-OAO.

The following boxplots (Figure 1 to Figure 8) and tables (Table 3 to Table 10)show the experimental results and statistical information (mean, variance, minimum, 25 percentile, median, 75 percentile and maximum) which summarize the results for the hypervolume and epsilon indicators.

Table 3. Statistic information of epsilon and hypervolume indicators for IRIS data

Epsilon indicators for IRIS data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	-0.3040	0.0139	-0.6	-0.4	-0.3	-0.2	-0.1	100	59.8069
SM-OA	-0.068	0.0059	-0.3	-0.1	-0.1	0	0	100	201.7863
SM-OAO	-0.244	0.0715	-1	-0.4	-0.1	0	0	83.46	128.1809
PM	**-0.0280**	0.0025	-0.2	-0.1	0	0	0	**100**	**0.9278**
Hypervolume indicators for IRIS data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	0.7515	0.0107	0.5018	0.7031	0.7717	0.8203	0.9689	100	59.8069
SM-OA	0.9310	0.0006	0.6985	0.8975	0.8975	1	1	100	201.7863
SM-OAO	0.7919	0.0744	0	0.6985	0.9395	1	1	83.46	128.1809
PM	**0.9953**	0.0001	0.9579	0.9902	1	1	1	**100**	**0.9278**

[1] 25%= 25 percentile, 75%= 75 percentile.
[2] set size= the average approximate set size for each of the multi-objective approaches.
[3] time(s)= the average time (measured in seconds) for getting a hypervolume and epsilon indicator.

[1]As we only keep the 'optimal' and 'near-optimal' solutions, we get at most 100 weakly Pareto-optimal solutions for the corresponding multi-objective method.

[2]From Lemma 3.2, we have $\theta^{pq} = \frac{\varrho_*^{12}}{\varrho_*^{pq}}$.

Figure 1. Boxplots of epsilon and hypervolume indicators for IRIS data

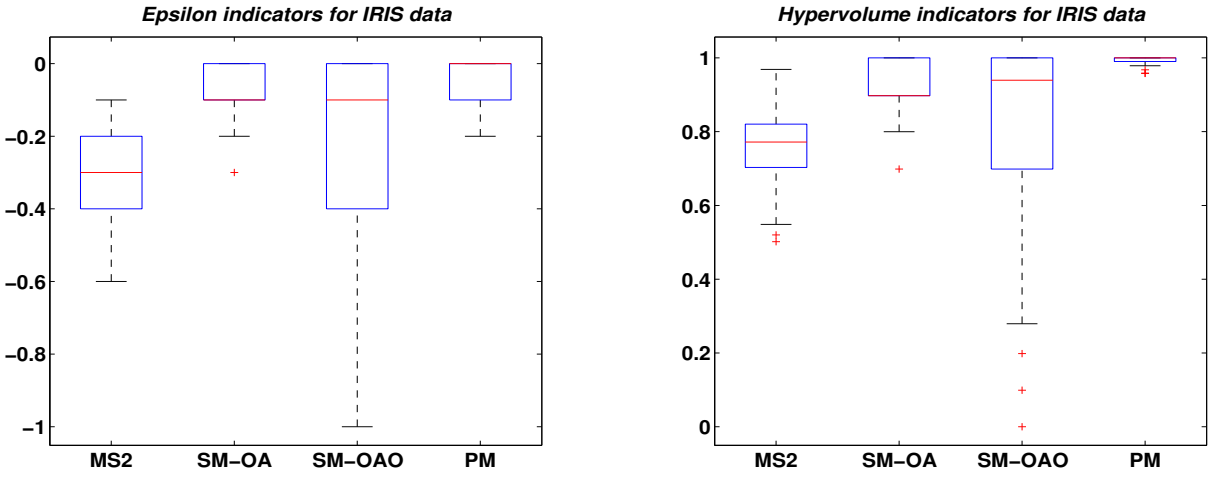

Figure 2. Boxplots of epsilon and hypervolume indicators for WINE data

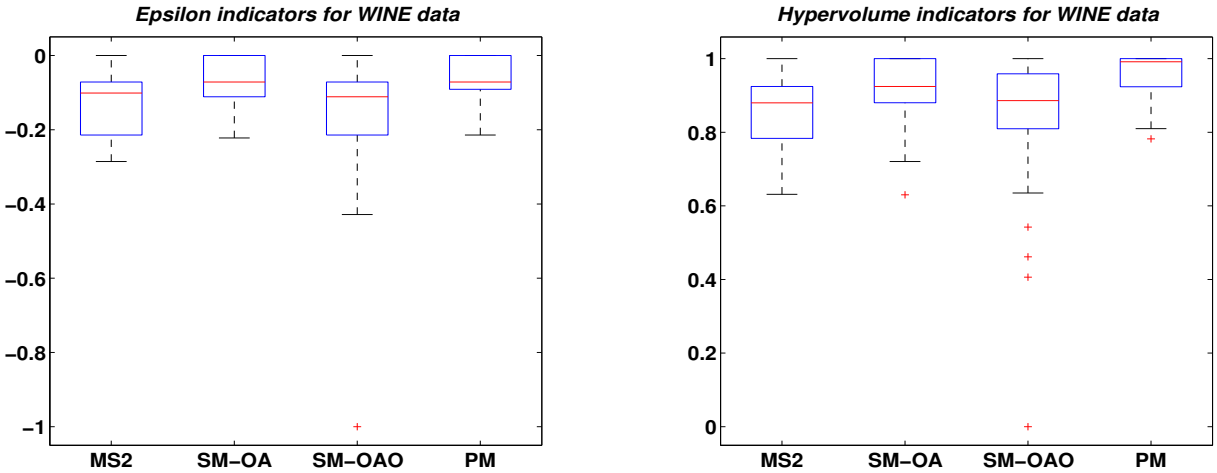

Table 4. Statistic information of epsilon and hyper volume indicators for WINE data

Epsilon indicators for WINE data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	-0.1290	0.0059	-0.2857	-0.2143	-0.1010	-0.0714	0	100	121.3078
SM-OA	-0.0755	0.0035	-0.2222	-0.1111	-0.0714	0	0	100	135.6761
SM-OAO	-0.1590	0.0255	-1	-0.2143	-0.1111	-0.0714	0	90.62	145.7128
PM	**-0.0576**	0.0034	-0.2143	-0.0909	-0.0714	0	0	**100**	**1.5107**
Hypervolume indicators for WINE data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	0.8661	0.0076	0.6314	0.7836	0.8800	0.9245	1	100	121.3078
SM-OA	0.9120	0.0070	0.6301	0.8802	0.9245	1	1	100	135.6761
SM-OAO	0.8415	0.0342	0	0.8096	0.8861	0.9589	1	90.62	145.7128
PM	**0.9544**	0.0038	0.7820	0.9238	0.9917	1	1	**100**	**1.5107**

Figure 3. Boxplots of epsilon and hypervolume indicators for SEEDS data

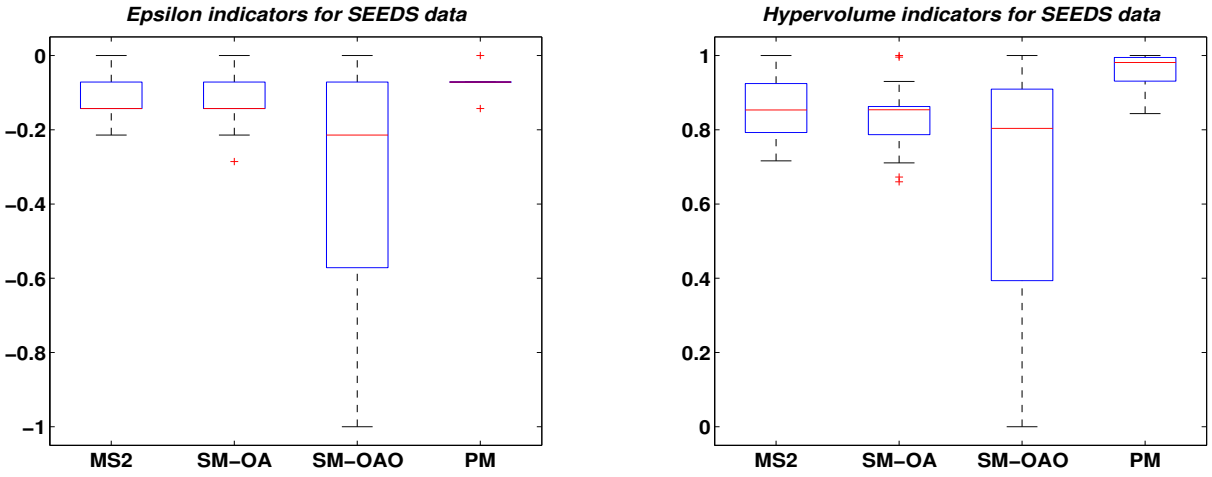

Table 5. Statistic information of epsilon and hypervolume indicators for SEEDS data

Epsilon indicators for SEEDS data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	-0.1171	0.0031	-0.2143	-0.1429	-0.1429	-0.0714	0	97.66	185.2025
SM-OA	-0.1243	0.0041	-0.2857	-0.1429	-0.1429	-0.0714	0	100	128.2825
SM-OAO	-0.3586	0.1248	-1	-0.5714	-0.2143	-0.0714	0	96.46	156.314
PM	**-0.0771**	0.0016	-0.1429	-0.0714	-0.0714	-0.0714	0	**100**	**1.7399**
Hypervolume indicators for SEEDS data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	0.8440	0.0057	0.7162	0.7927	0.8533	0.9244	1	97.66	185.2025
SM-OA	0.8313	0.0070	0.6599	0.7868	0.8540	0.8625	1	100	128.2825
SM-OAO	0.6421	0.1295	0	0.3937	0.8038	0.9096	1	96.46	156.314
PM	**0.9647**	0.0017	0.8436	0.9309	0.9813	0.9948	1	**100**	**1.7399**

Figure 4. Boxplots of epsilon and hypervolume indicators for VEHICLE data

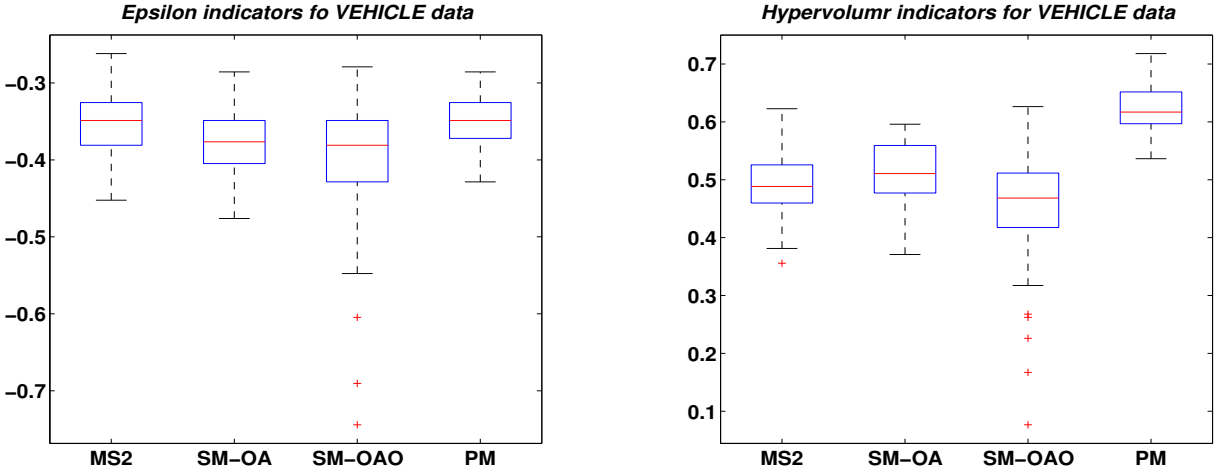

Table 6. Statistic information of epsilon and hypervolume indicators for VEHICLE data

Epsilon indicators for VEHICLE data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	-0.3522	0.002	-0.4524	-0.3810	-0.3488	-0.3256	-0.2619	95.72	244.0788
SM-OA	-0.3782	0.0017	-0.4762	-0.4048	-0.3765	-0.3488	-0.2857	70.18	290.3144
SM-OAO	-0.4086	0.0089	-0.7442	-0.4286	-0.3810	-0.3488	-0.2791	24.08	221.8891
PM	**-0.3497**	0.0014	-0.4286	-0.3721	-0.3488	-0.3256	-0.2857	**100**	**4.6410**
Hypervolume indicators for VEHICLE data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	0.4914	0.0026	0.3558	0.4599	0.4886	0.5257	0.6228	95.72	244.0788
SM-OA	0.5106	0.0030	0.3708	0.4771	0.5107	0.5594	0.5960	70.18	290.3144
SM-OAO	0.4490	0.0117	0.0766	0.4176	0.4685	0.5116	0.6264	24.08	221.8891
PM	**0.6220**	0.0018	0.5365	0.5970	0.6169	0.6518	0.7181	**100**	**4.6410**

Figure 5. Boxplots of epsilon and hypervolume indicators for CAR data

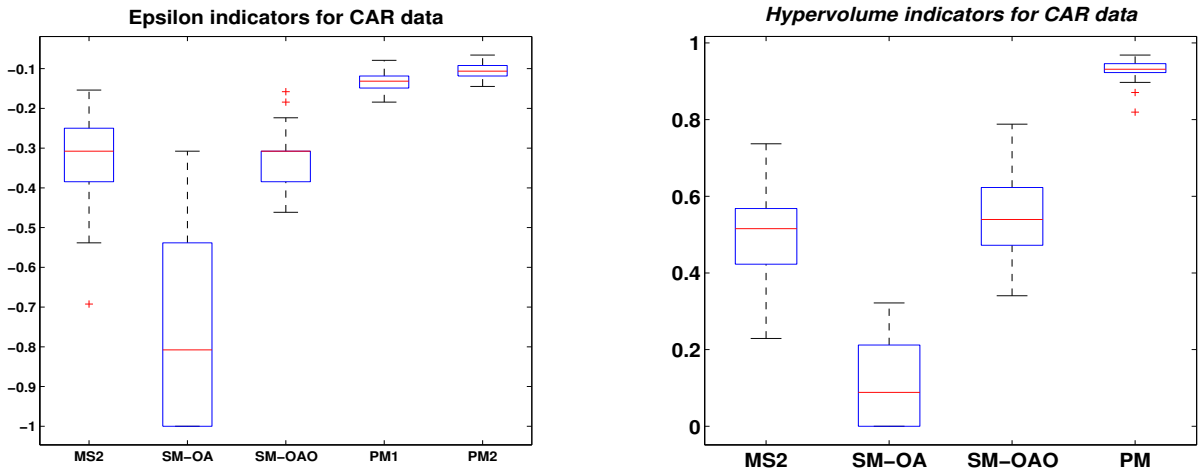

Table 7. Statistic information of epsilon and hypervolume indicators for CAR data

Epsilon indicators for CAR data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	-0.3215	0.0103	-0.6923	-0.3846	-0.3077	-0.25	-0.1538	72.44	340.689
SM-OA	-0.8154	0.0362	-1	-1	-0.8462	-0.6154	-0.4615	92.68	392.7254
SM-OAO	-0.2985	0.0080	-0.5385	-0.3846	-0.3077	-0.2308	-0.1447	34.84	521.4667
PM	**-0.1307**	0.0005	-0.1842	-0.1488	-0.1316	-0.1184	-0.0789	**100**	**2.7537**
Hypervolume indicators for CAR data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	0.5040	0.0109	0.2292	0.4229	0.5154	0.5679	0.7368	72.44	340.689
SM-OA	0.1051	0.0120	0	0	0.0883	0.2119	0.3217	92.68	392.7254
SM-OAO	0.5442	0.0107	0.3404	0.4722	0.5395	0.6230	0.7881	34.84	521.4667
PM	**0.9309**	0.0006	0.8196	0.9226	0.9314	0.9459	0.9684	**100**	**2.7537**

Figure 6. Boxplots of epsilon and hypervolume indicators for GLASS data

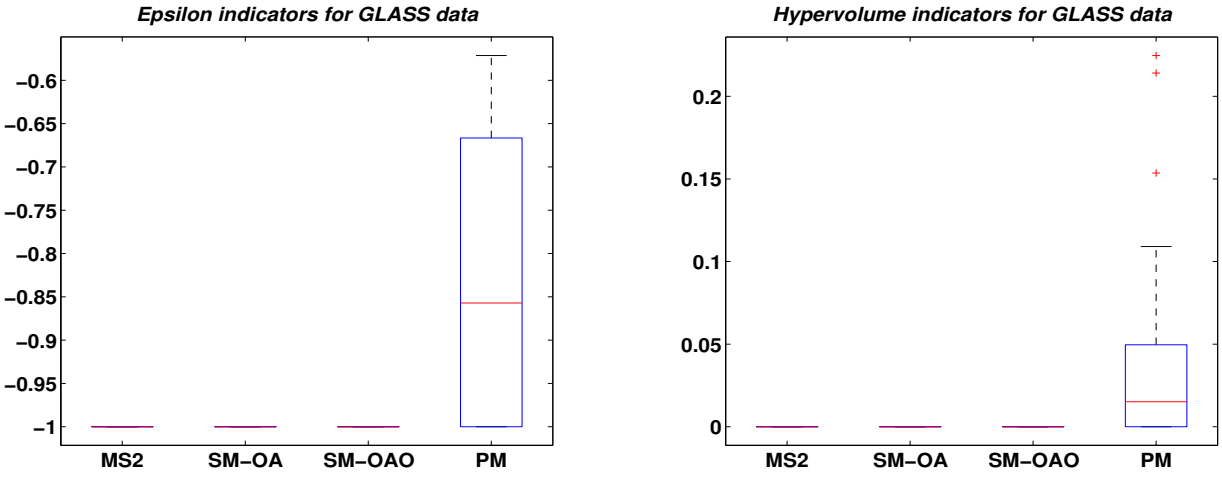

Table 8. Statistic information of epsilon and hypervolume indicators for GLASS data

Epsilon indicators for GLASS data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	-1	0	-1	-1	-1	-1	-1	73.22	139.3754
SM-OA	-1	0	-1	-1	-1	-1	-1	95.48	246.4926
SM-OAO	-1	0	-1	-1	-1	-1	-1	80.78	248.0824
PM	**-0.8344**	0.0245	-1	-1	-0.8571	-0.6667	-0.5714	**100**	**3.3367**
Hypervolume indicators for GLASS data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	0	0	0	0	0	0	0	73.22	139.3754
SM-OA	0	0	0	0	0	0	0	95.48	246.4926
SM-OAO	0	0	0	0	0	0	0	80.78	248.0824
PM	**0.0354**	0.0027	0	0	0.0151	0.0496	0.2247	**100**	**3.3367**

Figure 7. Boxplots of epsilon and hypervolume indicators for SCC data

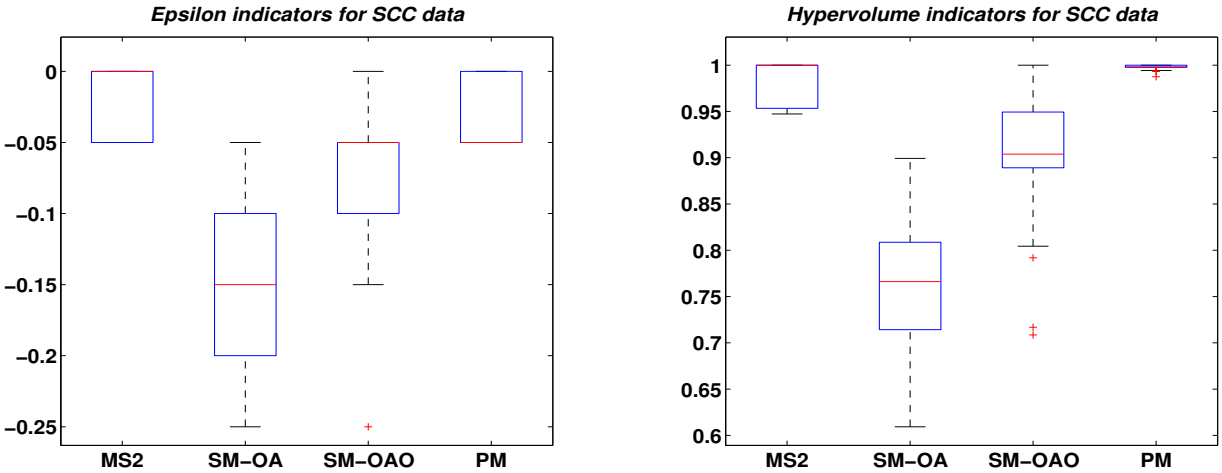

Table 9. Statistic information of epsilon and hyper volume indicators for SCC data

Epsilon indicators for SCC data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	**-0.018**	0.0006	-0.05	-0.05	0	0	0	74.14	285.3247
SM-OA	-0.136	0.0025	-0.25	-0.2	-0.15	-0.1	-0.05	96.66	182.1686
SM-OAO	-0.07	0.0018	-0.25	-0.1	-0.05	-0.05	0	16.76	395.1692
PM	*-0.032*	0.0006	-0.05	-0.05	-0.05	0	0	**100**	**2.1981**
Hypervolume indicators for SCC data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	0.9845	0.0005	0.9473	0.9534	1	1	1	74.14	285.3247
SM-OA	0.7570	0.0063	0.6093	0.7142	0.7662	0.8087	0.8993	96.66	182.1686
SM-OAO	0.9078	0.0043	0.7085	0.8891	0.9039	0.9494	1	16.76	395.1692
PM	**0.9981**	0.0000	0.9875	0.9977	0.9982	1	1	**100**	**2.1981**

Figure 8. Boxplots of epsilon and hypervolume indicators for CTG data

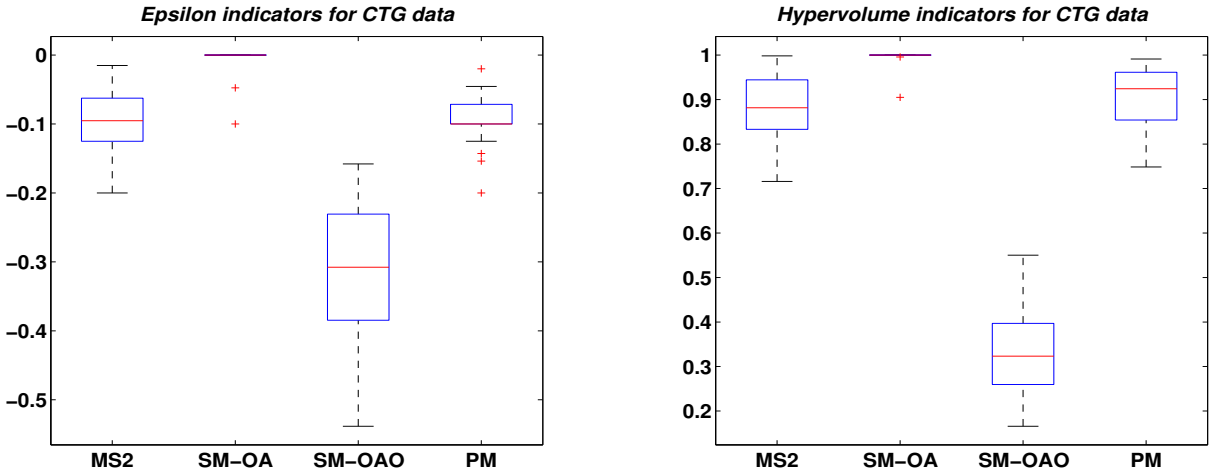

Table 10. Statistic information of epsilon and hypervolume indicators for CTG data

Epsilon indicators for CTG data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	-0.0934	0.0018	-0.2	-0.125	-0.0952	-0.0625	-0.0152	17.5641	9710.4
SM-OA	**-0.005**	0.0004	-0.1	0	0	0	0	82.9	1105.2
SM-OAO	-0.3131	0.0095	-0.5385	-0.3846	-0.3077	-0.2308	-0.1579	6.46	2496
PM	*-0.0936*	0.0013	-0.2	-0.1	-0.1	-0.0714	-0.02	**100**	**19.1136**
Hypervolume indicators for CTG data set									
Method	mean	variance	min	25%	median	75%	max	set size	time(s)
MS2	0.8783	0.005	0.7159	0.8332	0.8816	0.9443	0.9982	17.5641	9710.4
SM-OA	**0.9961**	0.0004	0.9050	1	1	1	1	82.9	1105.2
SM-OAO	0.3318	0.01	0.1657	0.2596	0.3234	0.3969	0.5502	6.46	2496
PM	*0.9030*	0.0048	0.7484	0.8540	0.9244	0.9614	0.9912	**100**	**19.1136**

We can see from these experimental results that **PM** outperforms the other multi-objective methods. The values of the indicators obtained by **PM** are consistently among the best for the multi-objective SVMs mentioned in this paper. For the IRIS, WINE, SEEDS, VEHICLE, CAR and GLASS data sets, **PM** has the largest mean values for both the hypervolume and epsilon indicators. For the SCC and CTG datasets, **PM** shows comparable performance with respect to the other three multi-objective approaches considered in this paper. Indeed for SCC and CTG, **PM** has the second largest values of the indicators among these multi-objective methods, and these values are very close to the largest ones.

Evaluating the performances in terms of the training times and the numbers of Pareto-optimal solutions computed within those times, we find that **PM** always outperforms the other three multi-objective methods. Specifically, **PM** is at least 60 times quicker than MS2, SM-OA and SM-OAO. Moreover, **PM** always obtains the largest approximation sets. As a summary, **PM** gives us more options in a shorter time compared with MS2, SM-OA and SM-OAO.

5. Conclusions

We have proposed a new multi-objective method (**PM**) for multi-class classification. This method is an extension of the bi-objective SVM method described in Carrizosa and Martin-Barragan (2006). Our numerical results show that the performance of **PM** can be advantageously compared to that of the other multi-class SVMs mentioned in this paper. Specifically, **PM** provides the highest classification accuracy with least training time among the multi-objective methods, and its performance is also comparable to that of the single-objective methods. **PM** provides us with a good approximation of the Pareto frontier while the single-objective methods need to conduct a large number of computations to offer us a similar number of options to choose from. Moreover, from the values of the indicators, we can also see that **PM** outperforms the other three multi-objective methods (MS2, SM-OA and SM-OAO), as it always gives us Pareto frontiers with high out-of-sample quality compared to these other multi-objective methods.

From a theoretical point of view, **PM** is also an efficient and effective method. From Corollary 3.4, **PM** provides us with Pareto-optimal solutions, while the other multi-objective approaches are only able to offer us weakly Pareto-optimal solutions. Furthermore, the Pareto-optimal solutions obtained from **PM** can be computed by solving one quadratic problem Eq. (1). From Property 3.1, Eq. (1) can be decomposed into several binary problems Eq. (3.1). This property significantly reduces the computational cost when the problems of interest have a large number of classes.

In summary, both from a theoretical and from a computational point of view, **PM** is an efficient method compared with MS2, SM-OA and SM-OAO. Besides, **PM**'s performance is comparable to that of the single-objective SVMs, while being able to provide not just one Pareto-optimal solution, but a very detailed approximation of the set of all the Pareto-optimal solutions.

References

Bredensteiner, E., Bennett, K., 1999. Multicategory classification by support vector machines. Computational Optimization and Applications 12 (1), 53–79.

Brown, M. P., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C. W., Furey, T. S., Ares, M., Haussler, D., 2000. Knowledge-based analysis of microarray gene expression data by using support vector machines. Proceedings of the National Academy of Sciences 97 (1), 262–267.

Carrizosa, E., Martin-Barragan, B., 2006. Two-group classification via a biobjective margin maximization model. European Journal of Operational Research 173 (3), 746–761.

Chinchuluun, A., Pardalos, P. M., 2007. A survey of recent developments in multiobjective optimization. Annals of Operations Research 154 (1), 29–50.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Machine learning 20 (3), 273–297.

Crammer, K., Singer, Y., 2002. On the algorithmic implementation of multiclass kernel-based vector machines. The Journal of Machine Learning Research 2, 265–292.

Deb, K., 2001. Multi-objective optimization. Multi-objective optimization using evolutionary algorithms, 13–46.

Ehrgott, M., 2005. Multicriteria optimization. Vol. 491. Springer Verlag.

Fonseca, C. M., Knowles, J. D., Thiele, L., Zitzler, E., 2005. A tutorial on the performance assessment of stochastic multiobjective optimizers. In: Third International Conference on Evolutionary Multi-Criterion Optimization (EMO 2005). Vol. 216. p. 240.

Guyon, I., Weston, J., Barnhill, S., Vapnik, V., 2002. Gene selection for cancer classification using support vector machines. Machine learning 46 (1), 389–422.

Hsu, C., Lin, C., 2002. A comparison of methods for multiclass support vector machines. Neural Networks, IEEE Transactions on 13 (2), 415–425.

Ishida, S., Tatsumi, K., Tanino, T., 2012. A multiobjective multiclass support vector machine based on one-against-one method. In: Proceedings of the 5th International Conference on Optimization and Control with Applications. pp. 75–78.

Kreßel, U. H.-G., 1999. Pairwise classification and support vector machines. In: Advances in kernel methods. MIT Press, pp. 255–268.

Lin, Y., 2002. Support vector machines and the bayes rule in classification. Data Mining and Knowledge Discovery 6 (3), 259–275.

Platt, J. C., Cristianini, N., Shawe-Taylor, J., 1999. Large margin dags for multiclass classification. In: NIPS. Vol. 12. pp. 547–553.

Tatsumi, K., Hayashida, K., Higashi, H., Tanino, T., 2007a. Multi-objective multiclass support vector machine for pattern recognition. In: SICE, 2007 Annual Conference. IEEE, pp. 1095–1098.

Tatsumi, K., Hayashida, K., Tanino, T., 2007b. Multi-objective multiclass support vector machine maximizing exact margins. FRONTIERS SCIENCE SERIES 49, 381.

Tatsumi, K., Kawachi, R., Hayashida, K., Tanino, T., 2009. Multiobjective multiclass soft-margin support vector machine maximizing pair-wise interclass margins. Advances in Neuro-Information Processing, 970–977.

Tatsumi, K., Tai, M., Tanino, T., 2010. Multiobjective multiclass support vector machine based on the one-against-all method. In: Neural Networks (IJCNN), The 2010 International Joint Conference on. IEEE, pp. 1–7.

Tatsumi, K., Tai, M., Tanino, T., 2011. Nonlinear extension of multiobjective multiclass support vector machine based on the one-against-all method. In: Neural Networks (IJCNN), The 2011 International Joint Conference on. IEEE, pp. 1570–1576.

Tatsumi, K., Tanino, T., 2014. Support vector machines maximizing geometric margins for multi-class classification. TOP 22 (3), 815–840.

Tong, S., Koller, D., 2002. Support vector machine active learning with applications to text classification. The Journal of Machine Learning Research 2, 45–66.

Vapnik, V., 1998. Statistical learning theory. 1998.

Vapnik, V., 2000. The nature of statistical learning theory. Springer-Verlag New York Inc.

Weston, J., Watkins, C., 1999. Support vector machines for multi-class pattern recognition. In: Proceedings of the seventh European symposium on artificial neural networks. Vol. 4. pp. 219–224.

AppendixA. Proof of Lemma 3.2

Assume that (ω^*, b^*, ξ^*) is optimal to Eq. (4). Notice that the feasible region of Eq. (4) and the feasible region of **PM** are the same.

If (ω^*, b^*, ξ^*) is not weakly Pareto-optimal to **PM**, there will exist a feasible (ω_0, b_0, ξ_0) such that

$$\begin{aligned}&\varrho^{12}(\omega_0, \sqrt{C}\xi_0, b_0) > \varrho^{12}(\omega^*, \sqrt{C}\xi^*, b^*), \varrho^{21}(\omega_0, \sqrt{C}\xi_0, b_0) > \varrho^{21}(\omega^*, \sqrt{C}\xi^*, b^*), \ldots,\\ &\varrho^{m(m-1)}(\omega_0, \sqrt{C}\xi_0, b_0) > \varrho^{m(m-1)}(\omega^*, \sqrt{C}\xi^*, b^*).\end{aligned}$$

As $\theta^{pq} > 0$, we have:

$$\begin{aligned}&\varrho^{12}(\omega_0, \sqrt{C}\xi_0, b_0) > \varrho^{12}(\omega^*, \sqrt{C}\xi^*, b^*), \theta^{21}\varrho^{21}(\omega_0, \sqrt{C}\xi_0, b_0) > \theta^{21}\varrho^{21}(\omega^*, \sqrt{C}\xi^*, b^*), \ldots,\\ &\theta^{m(m-1)}\varrho^{m(m-1)}(\omega_0, \sqrt{C}\xi_0, b_0) > \theta^{m(m-1)}\varrho^{m(m-1)}(\omega^*, \sqrt{C}\xi^*, b^*).\end{aligned}$$

This contradicts our assumption that (ω^*, b^*, ξ^*) is optimal to Eq. (4). As a consequence, (ω^*, b^*, ξ^*) must be a weakly Pareto-optimal solution to **PM**.

Assume now that (ω^*, b^*, ξ^*) is a weakly Pareto-optimal solution to **PM**. Then, for any feasible (ω_0, b_0, ξ_0), there exists some $i \neq j,\ i, j \in G$ such that $\varrho^{ij}(\omega_0, \sqrt{C}\xi_0, b_0) \leq \varrho^{ij}(\omega^*, \sqrt{C}\xi^*, b^*)$. Let

$$\varrho_* = \max\left(\varrho_*^{12}, \varrho_*^{21}, \cdots, \varrho_*^{(m-1)m}, \varrho_*^{m(m-1)}\right),$$

where $\varrho_*^{pq} = \varrho^{pq}(\omega^*, \sqrt{C}\xi^*, b^*),\ p \neq q,\ p, q \in G$.

Formulate the following problem:

$$\begin{aligned}\max_{\omega,b,\xi}\min\quad &\left(\frac{\varrho_*}{\varrho_*^{12}}\varrho^{12}(\omega, \sqrt{C}\xi, b), \frac{\varrho_*}{\varrho_*^{21}}\varrho^{21}(\omega, \sqrt{C}\xi, b), \ldots, \frac{\varrho_*}{\varrho_*^{m(m-1)}}\varrho^{m(m-1)}(\omega, \sqrt{C}\xi, b),\right),\\ \text{s.t.}\quad &\omega^{pq}x + b^{pq} + \xi_x^{pq} > 0,\ x \in I_p,\ p < q,\ p, q \in G,\\ &-\omega^{pq}x - b^{pq} + \xi_x^{qp} > 0,\ x \in I_q,\ p < q,\ p, q \in G,\\ &\xi_x^{pq} \geq 0,\ p \neq q,\ p, q \in G.\end{aligned} \tag{A.1}$$

Note that as (ω^*, b^*, ξ^*) is optimal to **PM**, it is also optimal to Eq. (A.1). By dividing all the objectives in Eq. (A.1) by $\frac{\varrho_*}{\varrho_*^{12}}$, we get the equivalent optimization problem:

$$\begin{aligned}\max_{\omega,b,\xi}\min\quad &\left(\varrho^{12}(\omega, \sqrt{C}\xi, b), \frac{\varrho_*^{12}}{\varrho_*^{21}}\varrho^{21}(\omega, \sqrt{C}\xi, b), \ldots, \frac{\varrho_*^{12}}{\varrho_*^{m(m-1)}}\varrho^{m(m-1)}(\omega, \sqrt{C}\xi, b),\right),\\ \text{s.t.}\quad &\omega^{pq}x + b^{pq} + \xi_x^{pq} > 0,\ x \in I_p,\ p < q,\ p, q \in G,\\ &-\omega^{pq}x - b^{pq} + \xi_x^{qp} > 0,\ x \in I_q,\ p < q,\ p, q \in G.\end{aligned} \tag{A.2}$$

Thus, (ω^*, b^*, ξ^*) is also optimal to Eq. (A.2).