Multi-Class Support Vector Machine via Maximizing Multi-Class Margins

Jie Xu[1,3], **Xianglong Liu**[2], **Zhouyuan Huo**[3], **Cheng Deng**[1], **Feiping Nie**[4,3], **Heng Huang**[3,1]
[1]Xidian University, Xi'an 710071, China
[2]School of Computer Science and Engineering, Beihang University, China
[3]University of Texas at Arlington, USA
[4]Northwestern Polytechnical University, China

Abstract

Support Vector Machine (SVM) is originally proposed as a binary classification model with achieving great success in many applications. In reality, it is more often to solve a problem which has more than two classes. So, it is natural to extend SVM to a multi-class classifier. There have been many works proposed to construct a multi-class classifier based on binary SVM, such as one versus rest strategy (OvsR), one versus one strategy (OvsO) and Weston's multi-class SVM. The first two split the multi-class problem to multiple binary classification subproblems, and we need to train multiple binary classifiers. Weston's multi-class SVM is formed by ensuring risk constraints and imposing a specific regularization, like Frobenius norm. It is not derived by maximizing the margin between hyperplane and training data which is the motivation in SVM. In this paper, we propose a multi-class SVM model from the perspective of maximizing margin between training points and hyperplane, and analyze the relation between our model and other related methods. In the experiment, it shows that our model can get better or compared results when comparing with other related methods.

1 Introduction

Support Vector Machine (SVM) is originally proposed as a binary classifier [Cortes and Vapnik, 1995] which has achieved great success in many different applications, such as handwritten digit recognition [Lauer *et al.*, 2007; Maji and Malik, 2009], speaker identification [Kamruzzaman *et al.*, 2010; Campbell *et al.*, 2006], event recognition [Chang *et al.*, 2016a; 2016b], feature selection [Cai *et al.*, 2011], action recognition [Yang *et al.*, 2017], and text categorization [Pilászy, 2005; Joachims, 1999; Nie *et al.*, 2014]. To apply SVM model to practical multi-class classification problems, many researchers tend to extend SVM to be a multi-classification classifier.

Existing multi-class SVM models can be mainly divided into two types [Hsu and Lin, 2002]. The first type is splitting the multi-class classification problem into multiple binary classification subproblems, like OvsR multi-class SVM and OvsO multi-class SVM. OvsR splits a c class classification problem to c binary classification subproblems, and OvsO strategy splits it to $\frac{c(c-1)}{2}$ binary classification subproblems. The second type is to solve multi-class classification problem in a single optimization model, like Weston's multi-class SVM [Weston and Watkins, 1998], Crammer multi-class SVM [Crammer and Singer, 2002], and regression-like formulation [Nie *et al.*, 2017]. However, none of them follow the motivation in SVM which is maximizing the margin between training points and hyperplane.

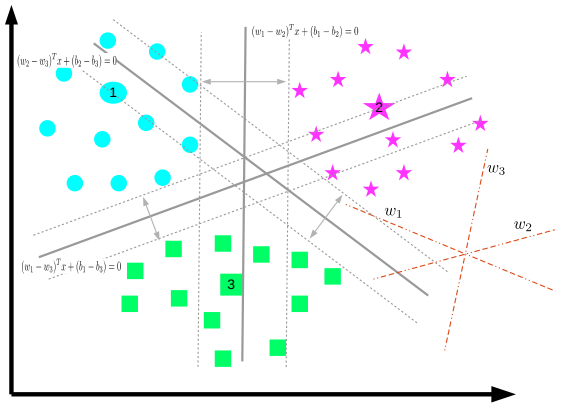


Figure 1: We train a multi-class Support Vector Machine model by maximize the margin between every two classes pair. In this three classes case, we need three parameters $\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3$, and use $\mathbf{w}_1$ and $\mathbf{w}_2$ to form a maximum-margin hyperplane for class 1 and 2.

In this paper, we will propose a novel multi-class Support Vector Machine model. Figure 1 presents the motivation of our model. It is derived by maximizing the margin between each two classes pair and the relationships between this model and other related multi-class SVM methods are also analyzed in this paper. We also extend our model to solve semi-supervised problems. Stochastic gradient descent with variance reduction algorithm (SVRG) is used to optimize this model, and it is proved to have faster convergence rate and reach better local optimum. Experiments on benchmark datasets show that our model can get equal or better results than other related methods.

2 Multi-Class Support Vector Machine

Although SVM model is a binary classifier, researchers works to extend it to solve multi-class classification problems. The earliest attempt is one versus all (one versus rest) strategy. Suppose there are n training data in the form of $(\mathbf{x}_i, y_i)$, and c classes in total, we need to build c binary SVM models. When we train j_{th} SVM model, we define class j as positive and the rest classes as negative. If the number of training data in each class is balanced, this subproblem is an unbalanced binary classification problem, and can be represented as,

$$\begin{aligned} \min_{\mathbf{w}_j, b_j} \quad & \frac{1}{2}\|\mathbf{w}_j\|_2^2 + C\sum_{i=1}^{n}\xi_i^j \\ s.t. \quad & \mathbf{w}_j^T\mathbf{x}_i + b_j \geq 1 - \xi_i^j, \text{ if } y_i = j \\ & \mathbf{w}_j^T\mathbf{x}_i + b_j \leq -1 + \xi_i^j, \text{ if } y_i \neq j \\ & \xi_i^j \geq 0 \end{aligned} \tag{1}$$

A new sample $\mathbf{x}_i$ is belonged to the class j which has the largest decision function value,

$$\tilde{y}_i = arg\max_j \mathbf{w}_j^T\mathbf{x} + b_j \tag{2}$$

As we mentioned, the main defect of this strategy is that each binary classification is unbalanced. This property may affect the performance of one versus rest strategy on multi-class classification problem.

One versus one strategy solves this problem by training more binary SVM models. In this strategy, we will train one binary SVM model for each two classes, so there are $\frac{c(c-1)}{2}$ models in total. For class j and k, the maximum-margin hyperplane between them is $\mathbf{w}_{jk}\mathbf{x}_i + b_{jk} = 0$, and it can be learned through the problem as follows,

$$\begin{aligned} \min_{\mathbf{w}_{jk}, b_{jk}} \quad & \frac{1}{2}\|\mathbf{w}_{jk}\|_2^2 + C\sum_{i=1}^{n}\xi_i^{jk} \\ s.t. \quad & \mathbf{w}_{jk}^T\mathbf{x}_i + b_{jk} \geq 1 - \xi_i^{jk}, \text{ if } y_i = j \\ & \mathbf{w}_{jk}^T\mathbf{x}_i + b_{jk} \leq -1 + \xi_i^{jk}, \text{ if } y_i = k \\ & \xi_i^j \geq 0 \end{aligned} \tag{3}$$

Voting strategy is used in testing, if $\text{sign}(\mathbf{w}_{jk}\mathbf{x}_i + b_{jk})$ says $\mathbf{x}_i$ is in class j, then the vote for class j is added by one, otherwise vote for class k is added by one. Final prediction class is the class which has the largest vote.

Instead of handing multi-class classification by solving multiple subproblems, Weston proposed to use one single objective function [Weston and Watkins, 1998],

$$\begin{aligned} \min_{W\in\mathbb{R}^{d\times c}, \mathbf{b}\in\mathbb{R}^c} \quad & \frac{1}{2}\sum_{j=1}^{c}\|\mathbf{w}_j\|_2^2 + C\sum_{i=1}^{n}\sum_{j\neq y_i}\xi_i^j \\ s.t. \quad & \mathbf{w}_{y_i}^T\mathbf{x}_i + b_{y_i} \geq \mathbf{w}_j^T\mathbf{x}_i + b_j + 2 - \xi_i^j \\ & \xi_i^j \geq 0, \forall i, j \in \{1, ..., c\} \setminus y_i \end{aligned} \tag{4}$$

Predicting class for a new sample is same as one versus rest method, and it is classified to be the class which has the largest value of decision function. Moreover, Crammer *et al*. proposed a new model as follows [Crammer and Singer, 2002],

$$\begin{aligned} \min_{W\in\mathbb{R}^{d\times c}, \mathbf{b}\in\mathbb{R}^c} \quad & \frac{1}{2}\sum_{j=1}^{c}\|\mathbf{w}_j\|_2^2 + C\sum_{i=1}^{n}\xi_i \\ s.t \quad & \mathbf{w}_{y_i}^T\mathbf{x}_i + \delta_{y_i,j} - \mathbf{w}_j^T\mathbf{x}_i \geq 1 - \xi_i, \forall i, j \end{aligned} \tag{5}$$

where $\delta_{y_i,j} = \begin{cases} 1 & \text{if} \quad y_i \neq j \\ 0 & \text{if} \quad y_i = j \end{cases}$.

Different from problem (4), it just considers one slack variable ξ_i for each sample $\mathbf{x}_i$. Guermeur developed a theoretical framework for multi-class SVMs [Guermeur, 2002] and proposed a model,

$$\begin{aligned} \min_{W\in\mathbb{R}^{d\times c}, \mathbf{b}\in\mathbb{R}^c} \quad & \frac{1}{2}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\|\mathbf{w}_j - \mathbf{w}_k\|_2^2 + C\sum_{i=1}^{n}\sum_{j\neq y_i}\xi_{ij} \\ s.t \quad & \mathbf{w}_{y_i}^T\mathbf{x}_i + b_{y_i} \geq \mathbf{w}_j^T\mathbf{x}_i + b_j + 1 - \xi_{ij} \\ & \xi_{ij} \geq 0, \forall i, j \in \{1, ..., c\} \setminus y_i \\ & \sum_{j=1}^{c}\mathbf{w}_j = \mathbf{0}, \quad \sum_{j=1}^{c} b_j = 0 \end{aligned} \tag{6}$$

In that paper, Guermeur also pointed out that its formulation is equal to the model proposed in [Bredensteiner and Bennett, 1999],

$$\begin{aligned} \min_{W\in\mathbb{R}^{d\times c}, \mathbf{b}\in\mathbb{R}^c} \quad & \frac{1}{2}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\|\mathbf{w}_j - \mathbf{w}_k\|_2^2 + \sum_{j=1}^{c}\|\mathbf{w}_j\|_2^2 \\ & + C\sum_{i=1}^{n}\sum_{j\neq y_i}\xi_{ij} \\ s.t \quad & \mathbf{w}_{y_i}^T\mathbf{x}_i + b_{y_i} \geq \mathbf{w}_j^T\mathbf{x}_i + b_j + 1 - \xi_{ij} \\ & \xi_{ij} \geq 0, \forall i, j \in \{1, ..., c\} \setminus y_i \end{aligned} \tag{7}$$

3 Maximizing Multi-Class Margins for SVM

The forthright methods to solve multi-class problems are one versus rest and one versus one strategy. However, one versus rest strategy has a drawback that the training data of each subproblem is unbalanced, which will affect the performance of each binary classifier. While for one versus one strategy, space and time consumption are two big problems, since we have to train $\frac{c(c-1)}{2}$ binary SVM model for c classes.

To address this issue, we propose a novel multi-class SVM model. In our model, we build a classifier for every two classes, however, different from one versus one strategy which stores $\frac{c(c-1)}{2}$ models $\mathbf{w}_{jk}$, we propose to use c vectors to simulate all these binary classifiers, for example a classifier between class j and k, $\mathbf{w}_{jk} = \mathbf{w}_j - \mathbf{w}_k$. In this way, our space consumption equals to the space consumption of one versus rest strategy, and avoid unbalanced training data subproblem at the same time. In the following section, we also prove that we just need to compute c decision functions in testing, and do not need to use vote strategy like one versus one strategy.

To solve the binary classification subproblem between class j and k, and maximize the margin between their data, soft margin objective function for this problem is,

$$\begin{aligned}\min_{\mathbf{w}_j,\mathbf{w}_k\in\mathbb{R}^d} \quad & \frac{1}{2}\left\|\mathbf{w}_j-\mathbf{w}_k\right\|_2^2+C\sum_{y_i\in\{j,k\}}\xi_i^{jk}\\ s.t \quad & y_i^{jk}f_{jk}(\mathbf{x}_i)\geq 1-\xi_i^{jk},\ \ \forall y_i\in\{j,k\}\\ & \xi_i^{jk}\geq 0\end{aligned} \tag{8}$$

where $\mathbf{x}_i\in\mathbb{R}^d$, non-negative slack variable ξ_i^{jk} is introduced to handle the data that are not linearly separable. $\sum_{y_i\in\{j,k\}}\xi_i^{jk}$ is the penalty term, and is able to reduce the number of training errors. Parameter C is a balance between training error and regularization term $\left\|\mathbf{w}_j-\mathbf{w}_k\right\|_2^2$. $f_{jk}(\mathbf{x}_i)=(\mathbf{w}_j-\mathbf{w}_k)^T\mathbf{x}_i+(b_j-b_k)$ is the decision function, and $y_i^{jk}=\begin{cases}1 & \text{if} \quad y_i=j\\ -1 & \text{if} \quad y_i=k\end{cases}$.

There are $\frac{c(c-1)}{2}$ binary classifiers in total, and all of them are essentially correlated. Then we build a multi-class SVM model as follows,

$$\begin{aligned}\min_{W\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} \quad & \frac{1}{2}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\left\|\mathbf{w}_j-\mathbf{w}_k\right\|_2^2+C\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\sum_{y_i\in\{j,k\}}\xi_i^{jk}\\ s.t \quad & y_i^{jk}f_{jk}(\mathbf{x}_i)\geq 1-\xi_i^{jk},\ \ \forall y_i\in\{j,k\}\\ & \xi_i^{jk}\geq 0\end{aligned} \tag{9}$$

There should be only one optimal solution. As we can see, if $W\in\mathbb{R}^{d\times c}$ and $\mathbf{b}\in\mathbb{R}^d$ is the optimal solution, let $\tilde{W}=W+\mathbf{1}\mathbf{1}^T$ and $\tilde{\mathbf{b}}=\mathbf{b}+\mathbf{1}$, then it is easy to know that $\mathbf{w}_j-\mathbf{w}_k=\tilde{\mathbf{w}}_j-\tilde{\mathbf{w}}_k$ and $b_j-b_k=\tilde{b}_j-\tilde{b}_k$, their objective function values are the same. There are multiple optimal solutions, and this is not what we want. Therefore, two more constraints should be imposed on problem (9), and the final objective function is as follows,

$$\begin{aligned}\min_{W\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} \quad & \frac{1}{2}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\left\|\mathbf{w}_j-\mathbf{w}_k\right\|_2^2+\frac{1}{2}\sum_{j=1}^{c}\left\|\mathbf{w}_j\right\|_2^2\\ & +\frac{1}{2}\sum_{j=1}^{c}b_j^2+C\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\sum_{y_i\in\{j,k\}}\xi_i^{jk}\\ s.t \quad & y_i^{jk}f_{jk}(\mathbf{x}_i)\geq 1-\xi_i^{jk},\ \ \forall y_i\in\{j,k\}\\ & \xi_i^{jk}\geq 0\end{aligned} \tag{10}$$

where decision function $f_{jk}(\mathbf{x}_i)=(\mathbf{w}_j-\mathbf{w}_k)^T\mathbf{x}_i+(b_j-b_k)$ and $y_i^{jk}=\begin{cases}1 & \text{if} \quad y_i=j\\ -1 & \text{if} \quad y_i=k\end{cases}$.

When we use this model to classify a new sample, we use the same voting strategy as in one versus one strategy multi-class SVM. If sign $\left((\mathbf{w}_j-\mathbf{w}_k)^T\mathbf{x}_i+b_j-b_k\right)=1$, then vote for class j is added by one. It is easy to find out that the more k that satisfies $\mathbf{w}_j^T\mathbf{x}_i+b_j>\mathbf{w}_k^T\mathbf{x}_i+b_k$, the more votes j will get. It is easy to know that the predicted class should have the largest decision function value,

$$\tilde{y}_i=arg\max_j \mathbf{w}_j^T\mathbf{x}_i+b_i. \tag{11}$$

In this way, we do not need to use vote strategy and compute decision function $\frac{c(c-1)}{2}$ times, we just need to compute decision function c times, and find the largest value.

4 Connections to Other SVM algorithms

In this section, we analyze the relations between our model and other multi-class SVM models. According to problem (10), we use hinge loss to replace all slack variable ξ_i^{jk}, and our problem becomes,

$$\begin{aligned}\min_{W\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} \quad & \frac{1}{2}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\left\|\mathbf{w}_j-\mathbf{w}_k\right\|_2^2+\frac{1}{2}\sum_{j=1}^{c}\left\|\mathbf{w}_j\right\|_2^2\\ & +\frac{1}{2}\sum_{j=1}^{c}b_j^2+C\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\sum_{y_i\in\{j,k\}}\left[1-y_i^{jk}f_{jk}(\mathbf{x}_i)\right]_+\end{aligned} \tag{12}$$

where decision function $f_{jk}(\mathbf{x}_i)=(\mathbf{w}_j-\mathbf{w}_k)^T\mathbf{x}_i+(b_j-b_k)$ and $y_i^{jk}=\begin{cases}1 & y_i=j\\ -1 & y_i=k\end{cases}$.

Proposition 4.1. *Our model can be transformed to the objective function (15) proposed in [Bredensteiner and Bennett, 1999] excluding term* $\frac{1}{2}\sum_{j=1}^{c}b_j^2$.

Proof. First, we have

$$\begin{aligned}&\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\sum_{y_i\in j,k}\left[1-y_i^{jk}f_{jk}(\mathbf{x}_i)\right]_+\\ =&\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}(\sum_{y_i\in j}\left[1-f_{jk}(\mathbf{x}_i)\right]_+\!+\sum_{y_i\in k}\left[1-f_{kj}(\mathbf{x}_i)\right]_+)\\ =&\sum_{j=1}^{c-1}\sum_{y_i\in j}\sum_{k=j+1}^{c}\left[1-((\mathbf{w}_j-\mathbf{w}_k)^T\mathbf{x}_i+(b_j-b_k))\right]_+\\ =&\sum_{j=1}^{c}\sum_{y_i\in j}(\sum_{k=j+1}^{c}\left[1-(f_{jk}(\mathbf{x}_i)\right]_+\!+\sum_{k=1}^{j-1}\left[1-f_{jk}(\mathbf{x}_i)\right]_+)\\ =&\sum_{i=1}^{n}\sum_{k\neq y_i}\left[1-((\mathbf{w}_{y_i}-\mathbf{w}_k)^T\mathbf{x}_i+(b_{y_i}-b_k))\right]_+\end{aligned} \tag{13}$$

Then, our objective function (12) can be transformed to an equivalent formation as follows,

$$\begin{aligned}\min_{W\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^d} \quad & \frac{1}{2}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\left\|\mathbf{w}_j-\mathbf{w}_k\right\|_2^2+\frac{1}{2}\sum_{j=1}^{c}\left\|\mathbf{w}_j\right\|_2^2\\ & +\frac{1}{2}\sum_{j=1}^{c}b_j^2+C\sum_{i=1}^{n}\sum_{k\neq y_i}\left[1-f_{y_ik}(\mathbf{x}_i)\right]_+\end{aligned} \tag{14}$$

We introduce a slack variable ξ in the function above, and problem (12) can be represented as,

$$\begin{aligned}\min_{W\in\mathbb{R}^{d\times c},\mathbf{b}\in\mathbb{R}^c} & \frac{1}{2}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\|\mathbf{w}_j-\mathbf{w}_k\|_2^2+\sum_{j=1}^{c}\|\mathbf{w}_j\|_2^2 \\ & +\frac{1}{2}\sum_{j=1}^{c}b_j^2+C\sum_{i=1}^{n}\sum_{j\neq y_i}\xi_{ij} \\ s.t \quad & \mathbf{w}_{y_i}^T\mathbf{x}_i+b_{y_i}\geq\mathbf{w}_j^T\mathbf{x}_i+b_j+1-\xi_{ij} \\ & \xi_{ij}\geq 0,\forall i,j\in\{1,...,c\}\setminus y_i\end{aligned} \tag{15}$$

Therefore, our model can be transformed to be the objective function proposed in [Bredensteiner and Bennett, 1999].

□

Proposition 4.2. *Our objective function (10) has the same optimal solution as problem (6) proposed in [Guermeur, 2002].*

Proof. To handle the constraints in objective function, we transform the primal problem (10) to a lagrangian dual problem,

$$\begin{aligned}L(W,\mathbf{b},\gamma,\lambda)= & \frac{1}{2}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\|\mathbf{w}_j-\mathbf{w}_k\|_2^2+\frac{1}{2}\sum_{j=1}^{c}\|\mathbf{w}_j\|_2^2 \\ & +\frac{1}{2}\sum_{j=1}^{c}b_j^2+C\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\sum_{y_i\in\{j,k\}}\xi_i^{jk} \\ & +\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\sum_{y_i\in\{j,k\}}\alpha_i^{jk}\left(1-\xi_i^{jk}-y_i^{jk}f_{jk}(\mathbf{x}_i)\right) \\ & -\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\sum_{y_i\in\{j,k\}}\beta_i^{jk}\xi_i^{jk}\end{aligned} \tag{16}$$

where $\alpha_i^{jk}\geq 0$ and $\beta_i^{jk}\geq 0$. Then we take derivative of $L(W,\mathbf{b},\gamma,\lambda)$ with respect to each $\mathbf{w}_j$ and b_j respectively, where $\mathbf{w}_j$ is column j of matrix W, and b_j is entry j in vector $\mathbf{b}$, and set them to zero. We obtain the following two equations,

$$\begin{aligned}\frac{\partial L}{\partial\mathbf{w}_j}= & \sum_{k\neq j}(\mathbf{w}_j-\mathbf{w}_k)+\mathbf{w}_j+\sum_{k<j}\sum_{y_i\in\{k,j\}}\alpha_i^{kj}y_i^{kj}\mathbf{x}_i \\ & -\sum_{j<k}\sum_{y_i\in\{j,k\}}\alpha_i^{jk}y_i^{jk}\mathbf{x}_i \\ & =\mathbf{0}\end{aligned} \tag{17}$$

where $\mathbf{0}$ is a vector in $\mathbb{R}^d$.

$$\begin{aligned}\frac{\partial L}{\partial b_j}= & b_j+\sum_{k<j}\sum_{y_i\in\{k,j\}}\alpha_i^{kj}y_i^{kj}-\sum_{j<k}\sum_{y_i\in\{j,k\}}\alpha_i^{jk}y_i^{jk} \\ & =0\end{aligned} \tag{18}$$

These two equations hold when we get the optimal solution, and then we sum up all the derivatives on $\mathbf{w}_j$, and get the following equation,

$$\begin{aligned}\sum_{j=1}^{c}\frac{\partial L(W,\mathbf{b},\gamma,\lambda)}{\partial\mathbf{w}_j}= & \sum_{j=1}^{c}\mathbf{w}_j+\sum_{j=1}^{c}(\sum_{k<j}\sum_{y_i\in\{k,j\}}\alpha_i^{kj}y_i^{kj}\mathbf{x}_i \\ & -\sum_{j<k}\sum_{y_i\in\{j,k\}}\alpha_i^{jk}y_i^{jk}\mathbf{x}_i) \\ & =\mathbf{0}\end{aligned} \tag{19}$$

In second term of the function above, whenever there is $\alpha_i^{kj}y_i^{kj}\mathbf{x}_i$ term, it will always be balanced out by another term $-\alpha_i^{kj}y_i^{kj}\mathbf{x}_i$, therefore it is easy to know that,

$$\sum_{j=1}^{c}\left(\sum_{k<j}\sum_{y_i\in\{k,j\}}\alpha_i^{kj}y_i^{kj}\mathbf{x}_i-\sum_{j<k}\sum_{y_i\in\{j,k\}}\alpha_i^{jk}y_i^{jk}\mathbf{x}_i\right)=\mathbf{0} \tag{20}$$

Therefore, equation holds that,

$$\sum_{j=1}^{c}\mathbf{w}_j=\mathbf{0} \tag{21}$$

Same as the derivation about parameter $\mathbf{w}_j$ above, when we sum up all derivatives on b_s, we get an equation

$$\sum_{j=1}^{c}b_j=0 \tag{22}$$

Above all, two constraints on W and $\mathbf{b}$ in problem (6) do not affect the optimal solution, and they share same formulation now. So, it is true that these two models have the same optimal solution.

□

Proposition 4.3. *Our objective function (10) has the same optimal solution as problem (4) proposed in [Weston and Watkins, 1998].*

Proof. As per the proof in Proposition 4.2, we have the equation that,

$$\sum_{j}^{c}\mathbf{w}_j=\mathbf{0} \tag{23}$$

where $\mathbf{w}_j$ are optimal solutions of the original problem. It also implies that,

$$\begin{aligned}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\|\mathbf{w}_j-\mathbf{w}_k\|_2^2 & \\ =\ \frac{1}{2}\sum_{j=1}^{c}\sum_{k=1}^{c}\|\mathbf{w}_j-\mathbf{w}_k\|_2^2 & \\ =\ \frac{1}{2}\sum_{j=1}^{c}\sum_{k=1}^{c}\left(\mathbf{w}_j^T\mathbf{w}_j+\mathbf{w}_k^T\mathbf{w}_k-2\mathbf{w}_j^T\mathbf{w}_k\right) & \\ =\ c\sum_{j=1}^{c}\|\mathbf{w}_j\|_2^2-\sum_{j=1}^{c}\mathbf{w}_j\sum_{k=1}^{c}\mathbf{w}_k & \\ =\ c\sum_{j=1}^{c}\|\mathbf{w}_j\|_2^2 & \end{aligned} \tag{24}$$

Above all, our objective function (10) has the same optimal solution as problem (4).

□

5 Optimization Algorithm

In this section, we are going to use stochastic gradient descent with variance reduction to solve this nonconvex and nonsmooth problem. Problem (10) can be rewritten as:

$$\min_{W,\mathbf{b}} \quad \frac{1}{2}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\|\mathbf{w}_j-\mathbf{w}_k\|_2^2+\frac{1}{2}\sum_{j=1}^{c}\|\mathbf{w}_j\|_2^2+\frac{1}{2}\sum_{j=1}^{c}b_j^2 + C\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\sum_{\tilde{y}_i\in\{j,k\},i\in U}\Big[1-\tilde{y}_i^{jk}f_{jk}(\mathbf{x}_i)\Big]_+ \tag{25}$$

where $W \in \mathbb{R}^{d\times c}$, $\mathbf{b} \in \mathbb{R}^c$, decision function $f_{jk}(\mathbf{x}_i) = (\mathbf{w}_j - \mathbf{w}_k)^T\mathbf{x}_i + (b_j - b_k)$. $\tilde{y}_i^{jk}$ is the predicted class for unlabeled data, and $\tilde{y}_i = arg\max_j \mathbf{w}_j^T\mathbf{x}_i + b_i$. In this function, L denotes the samples which are labeled. C is the weight of unlabeled training data. Usually, $C = \frac{1}{|U|}$.

In each iteration, we select a $\mathbf{x}_i \in \mathbb{R}^d$sample in training data randomly, and the objective function with respect to this sample can be represented as:

$$\min_{W,b} \quad \frac{1}{2}\sum_{j=1}^{c-1}\sum_{k=j+1}^{c}\|\mathbf{w}_j-\mathbf{w}_k\|_2^2+\frac{1}{2}\sum_{j=1}^{c}\|\mathbf{w}_j\|_2^2+\frac{1}{2}\sum_{j=1}^{c}b_j^2 + C\left(\sum_{j=\tilde{y}_i+1}^{c}[1-f_{\tilde{y}_i j}(\mathbf{x}_i)]_+ + \sum_{j=1}^{\tilde{y}_i-1}[1+f_{j\tilde{y}_i}(\mathbf{x}_i)]_+\right) \tag{26}$$

When sample is from U, $\tilde{y}_i$ is computed through function (11). Then we take derivative of the above function with respect to $\mathbf{w}_j$, and there are two different cases: when $j = y_i$, its subgradient is:

$$\frac{\partial l_i}{\partial \mathbf{w}_{y_i}} = -\sum_{k\neq y_i}(\mathbf{w}_k - \mathbf{w}_{y_i}) + \mathbf{w}_{y_i} + \sum_{k\neq y_i}\begin{cases}-C\mathbf{x}_i & \text{if} \quad 1-f_{y_i k}(\mathbf{x}_i) > 0\\ 0 & \text{if} \quad 1-f_{y_i k}(\mathbf{x}_i)\le 0\end{cases}$$

the other case is when $j \neq y_i$, its subgradient is as follows:

$$\frac{\partial l_i}{\partial \mathbf{w}_{j,j\neq y_i}} = -\sum_{k\neq j}(\mathbf{w}_k - \mathbf{w}_j) + \mathbf{w}_j + \begin{cases}C\mathbf{x}_i & \text{if} \quad 1-f_{jk}(\mathbf{x}_i) > 0\\ 0 & \text{if} \quad 1-f_{jk}(\mathbf{x}_i)\le 0\end{cases}$$

We also need to take derivative with respect to b_j and get the subgradient, there are two cases $j = y_i$ and $j \neq y_i$

$$\frac{\partial l_i}{\partial b_{y_i}} = b_{y_i} + \sum_{k\neq y_i}\begin{cases}-C & \text{if} \quad 1-f_{y_i k}(\mathbf{x}_i) > 0\\ 0 & \text{if} \quad 1-f_{y_i k}(\mathbf{x}_i)\le 0\end{cases} \tag{27}$$

$$\frac{\partial l_i}{\partial b_{k,k\neq y_i}} = b_k + \begin{cases}C & \text{if} \quad 1-f_{jk}(\mathbf{x}_i) > 0\\ 0 & \text{if} \quad 1-f_{jk}(\mathbf{x}_i)\le 0\end{cases} \tag{28}$$

The optimization algorithm to solve problem (25) is summarized in Algorithm 1.

Algorithm 1 SVRG to solve problem (25)

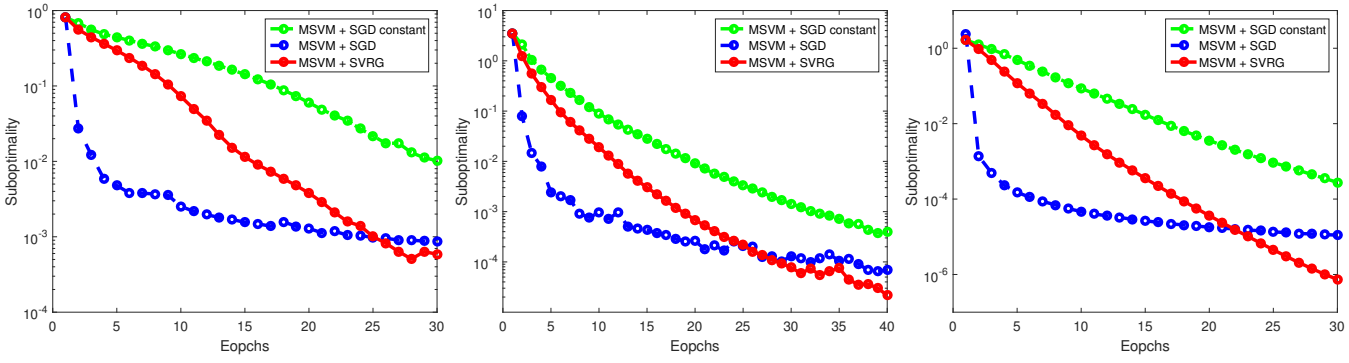
Initialize: $\tilde{W}^0 \in \mathbb{R}^{d\times c}$, $\tilde{\mathbf{b}}^0 \in \mathbb{R}^c$, learning rate η and $s = 0$.
while *not converge* **do**
  $s = s + 1$
  $\tilde{W} = \tilde{W}^{s-1}$, $\tilde{\mathbf{b}} = \tilde{\mathbf{b}}^{s-1}$
  for j=1,2...,c **do**
    $\tilde{\mathbf{u}}_j = \frac{1}{n}\sum_{i=1}^{n}\frac{\partial l}{\partial \tilde{\mathbf{w}}_j}$
    $\tilde{v}_j = \frac{1}{n}\sum_{i=1}^{n}\frac{\partial l}{\partial \tilde{b}_j}$
  end for
  $W^0 = \tilde{W}$
  for t=1,2,...,n **do**
    Randomly pick $i_t \in \{1, 2, ..., n\}$ and update parameters
    for j=1,2...,c **do**
      $\mathbf{w}_j^t = \mathbf{w}_j^{t-1} - \eta\left(\frac{\partial l_{i_t}}{\partial \mathbf{w}_j^{t-1}} - \frac{\partial l_{i_t}}{\partial \tilde{\mathbf{w}}_j} + \tilde{\mathbf{u}}_j\right)$
      $b_j^t = b_j^{t-1} - \eta\left(\frac{\partial l_{i_t}}{\partial b_j^{t-1}} - \frac{\partial l_{i_t}}{\partial \tilde{b}_j} + \tilde{v}_j\right)$
    end for
  end for
  Set $\tilde{W}^s = W^n$
end while

6 Experiments

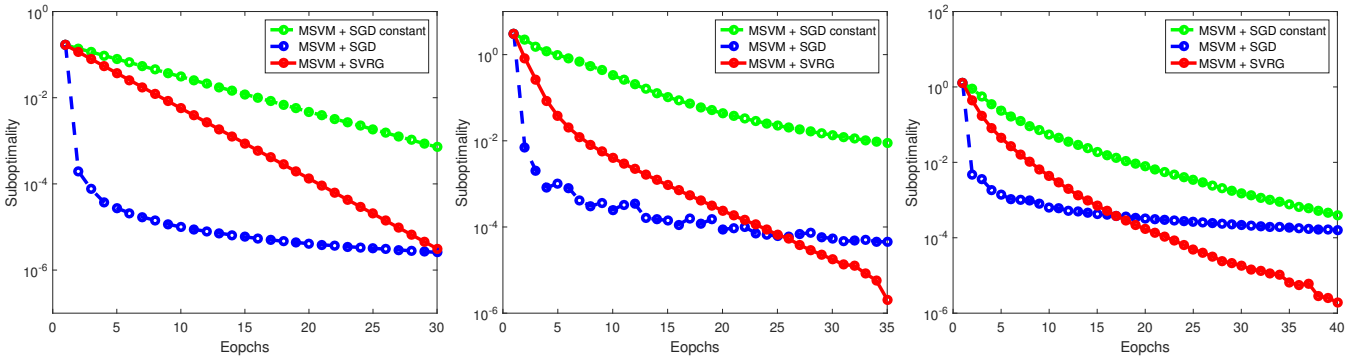
There are two main goals in our experiment, firstly, we will show that our model get the best or comparable performance compared with other related SVM models via computing classification accuracy; secondly, by looking at convergence figures, we will show that stochastic gradient descent method will get faster convergence rate on supervised learning task and even semi-supervised learning task which is a non-convex problem. Six multi-class classification datasets from UCI machine learning repository are used in our experiment [Lichman, 2013], main information are listed in Table 1.

In this experiment, we compared our method with 3 traditional multi-class support vector machine models, OvsR in (1), OvsO in (3) and Crammer in (5).

We use 5 times 5-fold cross validation and compute average accuracy for each method as final performance. In all experiments, we automatically tune the parameters by selecting among the values $\{10^r, r \in \{-5, ..., 5\}\}$. We compare the convergence rate of our method with stochastic gradient descent algorithm with constant learning rate and decreasing learning rate. We select the largest learning rate for



(a) iris dataset (b) dermatology dataset (c) satimage dataset



(d) svmguide2 dataset (e) usps5 dataset (f) waveform dataset

Figure 2: Suboptimality vs epochs. Suboptimality equals current objective function value minus optimal objective function value. The optimal objective function value is obtained by running our methods for a long time.

Table 1: Experiment datasets description.

Dataset	#Sample	#Attribute	#Class
iris	150	4	3
dermatology	366	34	6
satimage	4435	36	6
svmguide2	391	20	3
usps5	5427	256	5
waveform	2746	21	3

Table 2: Classification accuracy for all compared methods.

Dataset	OvsO	OvsR	Crammer	Proposed
iris	92.67	93.33	98.00	98.67
dermatology	96.71	96.99	95.07	96.99
satimage	82.64	87.24	85.68	86.02
svmguide2	80.00	82.56	81.03	82.31
usps5	97.75	97.95	97.55	97.80
waveform	86.56	86.12	86.60	87.27

each method and ensure that objective function value is decreasing during optimization. We plot a figure of suboptimality versus epoch, and suboptimality equals current value of objective function minus optimal value of objective function.

As shown in Table 2, we can see that our model can get always better or compared accuracy than other methods except OvsO multi-class SVM. It is normal because OvsO method has more parameters than our method. Figure 2 presents the convergence rate of three different stochastic gradient descent, constant learning rate, decreasing learning rate and stochastic gradient descent with variance reduction. It is clear that our method has much faster convergence rate after 10 epochs. Stochastic gradient descent method with constant learning rate and decreasing learning rate either has lower convergence rate or stop to converge because of decreasing learning rate.

7 Conclusion

In this paper, we follow the idea of maximizing margin between two classes and propose a novel multi-class SVM model. Analysis of the relation between our model and other related multi-class SVM models is also illustrated in the paper. Experiment results show that our model can get better or compared results than other related supervised and semi-supervised SVM models.

Acknowledgements

This work was partially supported by the National Natural Science Foundation of China (61572388 and 61402026), and U.S. NIH R01 AG049371, NSF IIS 1302675, IIS 1344152, DBI 1356628, IIS 1619308, IIS 1633753.

References

[Bredensteiner and Bennett, 1999] Erin J Bredensteiner and Kristin P Bennett. Multicategory classification by support vector machines. In *Computational Optimization*, pages 53–79. Springer, 1999.

[Cai *et al.*, 2011] Xiao Cai, Feiping Nie, Heng Huang, and Chris Ding. Multi-class l2,1-norm support vector machine. In *IEEE International Conference on Data Mining (ICDM 2011)*, pages 91–100, 2011.

[Campbell *et al.*, 2006] William M Campbell, Joseph P Campbell, Douglas A Reynolds, Elliot Singer, and Pedro A Torres-Carrasquillo. Support vector machines for speaker and language recognition. *Computer Speech & Language*, 20(2):210–229, 2006.

[Chang *et al.*, 2016a] Xiaojun Chang, Yao-Liang Yu, Yi Yang, and Eric P Xing. Semantic pooling for complex event analysis in untrimmed videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

[Chang *et al.*, 2016b] Xiaojun Chang, Yao-Liang Yu, Yi Yang, and Eric P Xing. They are not equally reliable: Semantic event search using differentiated concept classifiers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1884–1893, 2016.

[Cortes and Vapnik, 1995] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[Crammer and Singer, 2002] Koby Crammer and Yoram Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *The Journal of Machine Learning Research*, 2:265–292, 2002.

[Guermeur, 2002] Yann Guermeur. Combining discriminant models with new multi-class svms. *Pattern Analysis & Applications*, 5(2):168–179, 2002.

[Hsu and Lin, 2002] Chih-Wei Hsu and Chih-Jen Lin. A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2):415–425, 2002.

[Joachims, 1999] Thorsten Joachims. Transductive inference for text classification using support vector machines. In *ICML*, volume 99, pages 200–209, 1999.

[Kamruzzaman *et al.*, 2010] SM Kamruzzaman, ANM Karim, Md Islam, Md Haque, et al. Speaker identification using mfcc-domain support vector machine. *arXiv preprint arXiv:1009.4972*, 2010.

[Lauer *et al.*, 2007] Fabien Lauer, Ching Y Suen, and Gérard Bloch. A trainable feature extractor for handwritten digit recognition. *Pattern Recognition*, 40(6):1816–1824, 2007.

[Lichman, 2013] M. Lichman. UCI machine learning repository, 2013.

[Maji and Malik, 2009] Subhransu Maji and Jitendra Malik. Fast and accurate digit classification. *EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-159*, 2009.

[Nie *et al.*, 2014] Feiping Nie, Yizhen Huang, and Heng Huang. New primal svm solver with linear computational cost for big data classifications. In *The 31st International Conference on Machine Learning*, pages 505–513, 2014.

[Nie *et al.*, 2017] Feiping Nie, Xiaoqian Wang, and Heng Huang. Multiclass capped lp-norm svm for robust classifications. In *Thirty-First AAAI Conference on Artificial Intelligence*, pages 2415–2421, 2017.

[Pilászy, 2005] István Pilászy. Text categorization and support vector machines. In *The proceedings of the 6th international symposium of Hungarian researchers on computational intelligence*. Citeseer, 2005.

[Weston and Watkins, 1998] Jason Weston and Chris Watkins. Multi-class support vector machines. Technical report, Citeseer, 1998.

[Yang *et al.*, 2017] Yanhua Yang, Cheng Deng, Dapeng Tao, Shaoting Zhang, Wei Liu, and Xinbo Gao. Latent max-margin multitask learning with skelets for 3-d action recognition. *IEEE transactions on cybernetics*, 47(2):439–448, 2017.