Applied Computing and Informatics 15 (2019) 172–181

Contents lists available at ScienceDirect

Applied Computing and Informatics

journal homepage: www.sciencedirect.com

# Semi-supervised soft margin consistency based multi-view maximum entropy discrimination

Changming Zhu [a],*, Zhe Wang [b]

[a] College of Information Engineering, Shanghai Maritime University, Shanghai 201306, PR China
[b] College of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237, PR China

ARTICLE INFO

*Article history:*
Received 16 August 2017
Revised 6 October 2017
Accepted 20 October 2017
Available online 14 August 2018

*Keywords:*
Semi-supervised multi-view learning
Multi-view maximum entropy discrimination
Soft margin consistency

ABSTRACT

Multi-view maximum entropy discrimination (MVMED) and alternative MVMED (AMVMED) are proposed as extensions of maximum entropy discrimination (MED). In MVMED and AMVMED, they use hard margin consistency principle that the decision of margin parameter is related to classifier parameter directly. While the decision always be indirectly in practice, thus soft margin consistency based multi-view maximum entropy discrimination (SMVMED) has been proposed. But it is found that SMVMED is only adaptive to supervised problems. In this paper, we extend the model of SMVMED to the semi-supervised problems and develop a semi-supervised SMVMED (SSMVMED). Related experiments on multi-view data sets from different aspects have validated the effectiveness of SSMVMED theoretically and empirically. From the experiments, it is found that (1) compared with SMVMED, the average test accuracy of SSMVMED has a 2% enhancement; (2) SSMVMED costs more training time than SMVMED and the extra time is not more than 10%; (3) in terms of the generation of additional unlabeled instances, 'mid' strategy has a better test accuracy than 'self' and taking all instances to get the center brings a better test accuracy as well; (4) with SSMVMED, the applications to estimation problem and regression problem will be more feasible.

## 1. Introduction

### 1.1. Background

Multi-view data set is composed of instances with multiple views and each view consists of multiple features. A corresponding feature group is made up of these features. Take a video data set as an example, suppose this data set consists of multiple videos and each video appears in multiple different forms including visual, audio, and text. Then we treat each form as a view of a video. Moreover, each view has several features, for example, text view can be described by text color, text size, text content and color, size, content form a feature group of text view. In order to process those multi-view data sets, many multi-view learning machines are proposed as below.

(1) pre-fusion methods: multiple kernel learning (MKL) [1], centered alignment-based MKL algorithms (CABMKL) [2], simple MKL method (SMKL) [3], group Lasso-based MKL method (GLMKL) [4], localized MKL (LMKL) [5].
(2) late-fusion methods: robust late fusion method (RLF) [6].
(3) subspace approaches: multi-view linear discriminant analysis (MV-LDA) [7], multi-view canonical correlation analysis (MV-CCA) [8], multi-view locality preserving projections (MV-LPP) [9].
(4) disagreement-based methods: co-training [10], confident co-training with data editing (CoTrade) [11], co-regularized Laplacian SVM (Co-Lap) [12].

But these learning machines always neglect to consider uncertainties over model parameters and then maximum entropy discrimination (MED) [13] has been developed to consider this issue and learn a discriminative classifier. In MED, it learns a distribution $p(\Theta)$ over classifier parameter $\Theta$ and this is contrast to the traditional learning machines. In terms of traditional learning machines,

* Corresponding author.
*E-mail addresses:* cmzhu@shmtu.edu.cn (C. Zhu), wangzhe@ecust.edu.cn (Z. Wang).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.aci.2017.10.004
2210-8327/© 2018 The Authors. Production and hosting by Elsevier B.V. on behalf of King Saud University.

they only find a single classifier parameter $p(\Theta)$ of the discriminant function $L(\Theta)$ (e.g., $L(X_t|\Theta) = \theta^T X_t + b, \Theta = \{\theta, b\}$). While MED obtains a joint distribution $p(\Theta, \gamma)$ over $\Theta$ and margin parameters $\gamma$ by minimizing its relative entropy with respect to some prior target distribution $p_0(\Theta, \gamma)$ under certain large margin constraints, MED marginalizes out $\gamma$ to obtain $p(\Theta)$ [14].

Although MED considers the uncertainties over model parameters, its applicable scope limits to single-view problem. Thus multi-view maximum entropy discrimination (MVMED) [15] and alternative MVMED (AMVMED) [16] are developed to extend the model of MED to multi-view problems. MVMED and AMVMED exploit multiple views in a different style called margin consistency and enforce the margins from two views to be identical. In other words, they adopt the hard margin consistency while have no ability to process large data sets. So soft margin consistency based multi-view maximum entropy discrimination (SMVMED) [17] has been proposed. Different from MVMED and AMVMED, SMVMED achieves 'soft' margin consistency by utilizing the sum of two KL divergences $KL(p(\gamma)||q(\gamma))$ and $KL(q(\gamma)||p(\gamma))$ in the objective function, where $p(\gamma)$ and $q(\gamma)$ are the posteriors of two view margins, respectively. By balancing all involved terms in the objective function, SMVMED is more flexible.

While SMVMED is only feasible for supervised problems, i.e., the used instances are labeled. Indeed, most real-world multi-view data sets consist of labeled and unlabeled instances and they are named semi-supervised data sets. In order to process semi-supervised data sets, semi-supervised learning machines have been developed and introduced in many applications [18–24]. In terms of the applications of MED, there also exist some related learning machines for semi-supervised problems, for example, semi-supervised multi-sensor classification via consensus-based multi-view maximum entropy discrimination [25], semi-supervised learning via generalized maximum entropy [26], and semi-supervised multi-task learning via self-training and maximum entropy discrimination [27]. But these learning machines do not consider soft margin and we find that soft margin has its special physical meaning. As [17] said, if one view corresponds to one special sub-learning machine, its parameters include classifier parameter $\Theta$ and margin parameter $\gamma$. If the decision of $\gamma$ is related to $\Theta$ directly, we define it as hard margin consistency while if the decision is not directly, we call it as soft margin consistency. For soft margin consistency, [17] has also validated that the decision of $\gamma$ is more flexible and the performance and the conduction of a learning machine is much better and fast.

### 1.2. Proposal and trouble

Thus, in this paper, we still adopt soft margin consistency and apply it to semi-supervised problems and then propose the semi-supervised soft margin consistency based multi-view maximum entropy discrimination (SSMVMED). But during the process of SSMVMED, there is a potential trouble when the data sets consist of few labeled instances and many unlabeled ones. As we know, compared with unlabeled instances, labeled ones can provide more useful discriminant information while in real-world applications, most data sets consist of few labeled instances and many unlabeled ones and labeling instances is a high-cost task. Thus, for traditional semi-supervised problems, the performances of learning machines are sensitive to the data sets. In order to enhance the performance of a learning machine, a widely used and feasible method is generating additional unlabeled instances with the original labeled or unlabeled ones and combining all instances together. These additional unlabeled instances will possess some discriminant information derived from the original labeled and unlabeled instances. Here, Universum learning [28] is such a kind of method. For Universum learning, it aims to create additional unlabeled instances (also called Universum instances or Universum set) and incorporates priori knowledge which is introduced in the form of additional unlabeled instances into the learning process [29]. By Universum learning, the performance of a traditional learning machine can be boosted. Now Universum learning has been gradually spread into different applications [30–34] and some related methods are also developed including Universum support vector machine (U-SVM) [35] and self-Universum support vector machine (SUSVM) [36]. So for our SSMVMED, we also adopt Universum learning to add useful discriminant information.

But for these Universum-based learning machines, there still exists two key problems. First one is that when generating Universum set, the weights of views and features which play different discriminant roles are always neglected. Second one is that when generating the additional unlabeled instances, traditional Universum-based learning machines only adopt labeled or unlabeled instances for generation. In order to solve the first problem, we adopt weighted multi-view clustering (WMVC) [37] which is a multi-view clustering method and can find the optimal cluster assignment. With WMVC, the weights of views and features can be gotten. For the second problem, we try to design some schemes and adopt both labeled and unlabeled instances to generate the additional unlabeled instances.

### 1.3. Novelty and practical applications

The novelty of the proposed SSMVMED is given below.

First, compared with some semi-supervised learning machines with MED, SSMVMED adopts the soft margin and inherits the advantages of soft margin consistency which makes the decision of margin parameters be more flexible and the performance and the conduction of a learning machine be much better and faster.

Second, SSMVMED extends the model of SMVMED to semi-supervised problems and this enlarges the applicable scope of MED.

Third, during the procedure of SSMVMED, it improves the Universum learning methods, and when generating the additional unlabeled instances, the weights of views and features will be considered and both labeled and unlabeled instances are used for the generation.

The practical applications of SSMVMED can include estimation problems, regression problem, classification problems and so on. In our work, we will adopt some related experiments to show the effectiveness of our proposed SSMVMED in these applications.

### 1.4. Framework

The rest of this paper is organized as below. Related work about MED is given in Section 2. Description of SSMVMED is given in Section 3. Experiments are given in Section 4. The conclusions are given in Section 5.

## 2. Related work

Since our SSMVMED is the extension of SMVMED and SMVMED is different from MVMED and AMVMED, thus we will review MVMED, AMVMED, and SMVMED here.

### 2.1. MVMED

MVMED was proposed as an extension of MED to the multi-view learning setting and it considers a joint distribution $p(\Theta_1, \Theta_2)$ over the view 1 classifier parameter $\Theta_1$ and view 2 classifier parameter $\Theta_2$. Using the augmented joint distribution

$p(\Theta_1, \Theta_2, \gamma)$, the model of MVMED is given in Eq. (1) [15] where $1 \leqslant t \leqslant N$. In this model, $L_1(X_t^1|\Theta_1)$ and $L_2(X_t^2|\Theta_2)$ are discriminant functions from two views, respectively. $y_t$ is the class label of $t$th labeled instance $X_t$ and its margin parameter is $\gamma_t$. $X_t^1$ and $X_t^2$ are the representations of $X_t$ in the first and second views, respectively. With MVMED, multi-view feature selection, multi-view multi-task learning, multi-view structure learning, and some other problems can be solved well.

$$min_{p(\Theta_1,\Theta_2,\gamma)} KL(p(\Theta_1,\Theta_2,\gamma)||p_0(\Theta_1,\Theta_2,\gamma)) \tag{1}$$
$$s.t.\begin{cases} \int p(\Theta_1,\Theta_2,\gamma)[y_t L_1(X_t^1|\Theta_1) - \gamma_t] d\Theta_1 d\Theta_2 d\gamma \geqslant 0 \\ \int p(\Theta_1,\Theta_2,\gamma)[y_t L_2(X_t^2|\Theta_2) - \gamma_t] d\Theta_1 d\Theta_2 d\gamma \geqslant 0 \end{cases}$$

### 2.2. AMVMED

Different from MVMED, AMVMED considers two separate distributions $p(\Theta_1)$ over $\Theta_1$ and $p(\Theta_2)$ over $\Theta_2$ and balances KL divergences of their augmented distributions with respect to the corresponding prior distributions. The model of AMVMED is given in Eq. (2) [16] and $1 \leqslant t \leqslant N, \rho$ is a coefficient.

$$\begin{aligned} min_{p_1(\Theta_1,\gamma),p_2(\Theta_2,\gamma)} & \rho KL(p_1(\Theta_1,\gamma)||p_0(\Theta_1,\gamma)) \\ & + (1-\rho) KL(p_2(\Theta_2,\gamma)||p_0(\Theta_2,\gamma)) \end{aligned} \tag{2}$$
$$s.t.\begin{cases} \int p(\Theta_1,\gamma)[y_t L_1(X_t^1|\Theta_1) - \gamma_t] d\Theta_1 d\gamma \geqslant 0 \\ \int p(\Theta_2,\gamma)[y_t L_2(X_t^2|\Theta_2) - \gamma_t] d\Theta_2 d\gamma \geqslant 0 \\ \int p(\Theta_1,\gamma) d\Theta_1 = \int p(\Theta_2,\gamma) d\Theta_2 \end{cases}$$

As [17] said, for both MVMED and AMVMED, they exploit the multiple views in a different style called margin consistency which indicate the margins from two views are enforced to be identical. Moreover, the margins are related to the parameters $\Theta_1$ and $\Theta_2$ directly and those margins are named as hard margins. Although MVMED and AMVMED have provided state-of-the-art multi-view learning performance, hard margin requirement is somewhat too strong to fulfill in many cases. For example, all positive margins can lead to the same label prediction in binary classifications.

### 2.3. SMVMED

SMVMED is different from MVMED and AMVMED due to the margin of SMVMED is soft. SMVMED achieves margin consistency by minimizing the KL-divergence between the posteriors of margin parameters from two views. Then a trade-off parameter balancing large margin and margin consistency is also introduced to make the model more flexible. The model of SMVMED is given below. Here, $p(\star)$ or $q(\star)$ is a distribution over $\star$ and $p(\star_1, \star_2)$ is a joint distribution over $\star_1$ and $\star_2$. The parameter $\alpha$ is a parameter playing the trade-off role of balancing large margin and soft margin consistency. Compared with MVMED and AMVMED, SMVMED is more flexible and the performance and the conduction of a learning machine with SMVMED is much better and faster.

$$\begin{aligned} min_{p(\Theta_1,\gamma),q(\Theta_2,\gamma)} & KL(p(\Theta_1)||p_0(\Theta_1)) + KL(q(\Theta_2)||q_0(\Theta_2)) \\ & + (1-\alpha) KL(p(\gamma)||p_0(\gamma)) + (1-\alpha) KL(q(\gamma)||q_0(\gamma)) \\ & + \alpha KL(p(\gamma)||q(\gamma)) + \alpha KL(q(\gamma)||p(\gamma)) \end{aligned} \tag{3}$$
$$s.t.\begin{cases} \int p(\Theta_1,\gamma)[y_t L_1(X_t^1|\Theta_1) - \gamma_t] d\Theta_1 d\gamma \geqslant 0 \\ \int q(\Theta_2,\gamma)[y_t L_2(X_t^2|\Theta_2) - \gamma_t] d\Theta_2 d\gamma \geqslant 0 \end{cases}$$

## 3. Semi-supervised Soft Margin Consistency based Multi-view Maximum Entropy Discrimination (SSMVMED)

Our proposed SSMVMED consists of three steps. First, compute the weights of views and features by WMVC [37]. Second, generate the additional unlabeled instances. Third, apply the original labeled, unlabeled, and the generated additional unlabeled instances into the model of SSMVMED.

### 3.1. Obtain the weights of views and features

In order to obtain the weights of views and features, we adopt WMVC for help. Suppose there is a multi-view data set consisting of $N$ instances represented by $V$ views, i.e., $\chi = \{X_1^1, X_1^2, \ldots, X_1^V, \ldots, X_N^1, X_N^2, \ldots, X_N^V\}$ where $X_i^v \in \mathbb{R}^{d^v}$ is the representation of $i$th instance $X_i$ in the $v$th view and $d^v$ is the dimension of $v$th view. Here, we let $X^v = \{X_1^v, X_2^v, \ldots, X_N^v\}$ represent the $v$th view and $\chi$ can be represented as $\chi = \{X^1, X^2, \ldots, X^v, \ldots, X^{V-1}, X^V\}$. Then according to the notion of WMVC, we try to obtain the weights of views and the weights of features for each view. Let weight of each view be $\omega_v$ where $v = 1, 2, \ldots, V$ and weight for the $l$th feature of $v$th view is $\tau_l^v$ where $l = 1, 2, \ldots, d^v$. Here, each weight should not be less than zero. Furthermore, $\sum_{v=1}^V \omega_v = 1$ and for each view $X^v, \sum_{l=1}^{d^v} \tau_l^v = 1$. Then according to the notion of WMVC, the whole multi-view data set should be divided into several clusters. Let the number of clusters be $M, k$ denote the index of clusters, and $\delta_{ik}$ denote the belonging of the instance $X_i$, if instance $X_i$ belongs to $k$th cluster, then $\delta_{ik} = 1$, otherwise, $\delta_{ik} = 0$. For any instance $X_i, \sum_{k=1}^M \delta_{ik} = 1$. The objective function of WMVC is given in Eq. (4).

$$min_{\{\delta_{ik}\}_{k=1}^M, \{\omega_v\}_{v=1}^V, \{\tau^v\}_{v=1}^V} \quad \varepsilon_H \tag{4}$$
$$s.t.\begin{cases} \sum_{k=1}^M \delta_{ik} = 1, \quad \forall i, \quad \delta_{ik} \in \{0, 1\} \\ \sum_{v=1}^V \omega_v = 1, \quad \omega_v \geqslant 0 \\ \sum_{l=1}^{d^v} \tau_l^v = 1, \quad \forall v, \quad \tau_l^v \geqslant 0 \end{cases}$$

In this function, $\varepsilon_H = \sum_{v=1}^V (\omega_v)^p \sum_{i=1}^N \sum_{k=1}^M \delta_{ik} ||diag(\tau^v)(X_i^v - m_k^v)||^2 + \beta \sum_{v=1}^V ||\tau^v||^2, \tau^v = \{\tau_1^v, \tau_2^v, \ldots, \tau_{d^v}^v\}$, and $diag(\tau^v)$ represents the diagonal matrix where other elements in this matrix are zeros. Moreover, $m_k^v = \frac{\sum_{i=1}^N \delta_{ik} X_i^v}{\sum_{i=1}^N \delta_{ik}}$ is the cluster center of $k$th cluster in the $v$th view. Furthermore, $\beta \sum_{v=1}^V ||\tau^v||^2$ is used to control the sparsity of the feature weight vectors $\tau^v, \forall v$ so as to avoid the situation that only a few features are selected in getting a very small but meaningless objective value. The parameters $p$ and $\beta$ are the exponential and balancing parameters, which are selected according to the priori knowledge of data so as to help controlling the sparsity of the view weight vector $\omega = \{\omega_1, \omega_2, \ldots, \omega_V\}$ and the feature weight vectors $\tau^v, \forall v = 1, 2, \ldots, V$ respectively.

Then with WMVC, $\omega_v$ can be updated by Eq. (5) and $\tau^v$ can be updated by Eq. (6) until the computations of $\omega_v$ and $\tau^v$ be convergent or the iteration times is up to a maximum number. In these equations, $D_v = \sum_{i=1}^N \sum_{k=1}^M \delta_{ik} \left|\left|diag(\tau^v)(X_i^v - m_k^v)\right|\right|^2$ and $B_l^v = \beta + (\omega_v)^p \sum_{i=1}^N \sum_{k=1}^M \delta_{ik} (X_i^v - m_k^v)_l^2$ where $(\star)_l$ represents the $l$th element of $(\star)$.

$$\omega_v = \frac{1}{\sum_{u=1}^V \left(\frac{D_v}{D_u}\right)^{1/(p-1)}} \quad p > 1 \tag{5}$$
$$\omega_v = \begin{cases} 1, v = argmin_u D_u \\ 0, \quad otherwise \end{cases} \qquad p = 1$$

$$\tau_l^v = \frac{1}{\sum_{m=1}^{d^v} \frac{B_l^v}{B_m^v}} \quad \forall l \tag{6}$$

Finally, we can get the optimal or final $\omega$ and $\tau^v$ and the weights of views and features are also gotten.

### 3.2. Approaches of generating additional unlabeled instances

After we get the weights of views and features, we will adopt Universum learning to generate the additional unlabeled instances (i.e., Universum instances or Universum set). In our paper, the generation approaches used are given in Table 1. From this table, it is found that each approach has a code with the form 'A-B-C'. 'A' has three choices, 'all', 'unlabeled', 'labeled'. 'B' has three choices, 'all', 'near', 'far'. 'C' has two choices, 'mid' and 'self'. For 'A', 'all' ('unlabeled', 'labeled') represents that one computes the midpoint of all (unlabeled, labeled) instances as a center. For 'B', 'all', 'near', and 'far' represent that one uses all instances, $K$ instances which locates nearest from the center, and $K$ instances which locates farthest from the center as selected instances respectively. For 'C', 'mid' represents one takes the midpoint of a selected instance and the center to construct Universum set while 'self' represents Universum set consists of the selected instances themselves. We take $\dot{U}_{1-2}$ as the example. In this approach, we first to compute the mean of all instances as a center, then we select $K$ instances which locates nearest from this center, finally take the midpoint of a selected instance and the center to construct Universum set.

For all approaches used here, when we compute midpoint or distance, the weights of views and features should be used. Concretely speaking, if there are two instances, $X_1 = \{X_1^1, X_1^2, \ldots, X_1^V\}$ and $X_2 = \{X_2^1, X_2^2, \ldots, X_2^V\}$. The weights of views are $\omega_1, \omega_2, \ldots, \omega_V$ and the weights of features are $\tau^v$ where $v = 1, 2, \ldots, V$ and $\tau^v = \{\tau_1^v, \tau_2^v, \ldots, \tau_{d^v}^v\}$. $d^v$ is the dimension of $v$th view and $\tau_l^v$ represents weight of the $l$th feature of $v$th view. Then the midpoint of $X_1$ and $X_2$ is $\frac{\sum_{v=1}^{V}\omega_v\sum_{l=1}^{d^v}(\tau_l^v(X_1^v+X_2^v)_l)}{2}$ and the distance between $X_1$ and $X_2$ is $\sum_{v=1}^{V}\omega_v\sum_{l=1}^{d^v}(\tau_l^v(X_1^v - X_2^v)_l^2)$ where $(\star)_l$ represents the $l$th element of $(\star)$.

### 3.3. Solution of SSMVMED

Once we generate additional unlabeled instances, we will apply them along with the original labeled and unlabeled instances into the model of SSMVMED. For convenience, we adopt a binary-view data set for example. For a data set with more than two views, we can divide it into several binary-view problems, and for each binary-view problem, a sub model of SSMVMED is gotten and they can be integrated together and get the final model for multi-view data sets.

Suppose for a binary-view data sets, there are $N$ (here $N$ is different from the $N$ in Section 3.1 which denotes the number of all instances) labeled instances $\{X_t^1, X_t^2, y_t\}$ and $L$ unlabeled instances (including the generated additional ones) $\{U_i^1, U_i^2\}$. Here, $X_t^1$ $(X_t^2)$ represents the first (or second) view of the $t$th labeled instance $X_t$ and $y_t$ is its class label. For $U_i^1$ $(U_i^2)$, it represents the first (or second) view of $i$th unlabeled instance $U_i$. Different from Eq. (3), the model of SSMVMED adds the constraint of unlabeled instances and the parameter $\gamma = \{\gamma_j\}$ where $j = 1, 2, \ldots, N, N+1, \ldots, N+L$. Namely, SSMVMED considers the margin constraint of each labeled or unlabeled instance. Eq. (7) shows the model of SSMVMED. In this model, $1 \leqslant t \leqslant N, 1 \leqslant i \leqslant L, p(\star)$ $(q(\star))$ is a distribution over $\star$ and $p(\star_1, \star_2)$ $(q(\star_1, \star_2))$ is a joint distribution over $\star_1$ and $\star_2$. $\alpha$ is still a parameter playing the trade-off role of balancing large margin and soft margin consistency. $\Theta_1$ $(\Theta_2)$ is the view 1 (2) classifier parameter. $\gamma$ is the margin parameter.

$$\begin{aligned} & min_{p(\Theta_1,\gamma),q(\Theta_2,\gamma)} KL(p(\Theta_1)||p_0(\Theta_1)) + KL(q(\Theta_2)||q_0(\Theta_2)) \\ & \quad + (1-\alpha)KL(p(\gamma)||p_0(\gamma)) + (1-\alpha)KL(q(\gamma)||q_0(\gamma)) \\ & \quad + \alpha KL(p(\gamma)||q(\gamma)) + \alpha KL(q(\gamma)||p(\gamma)) \\ & s.t. \begin{cases} \int p(\Theta_1,\gamma)[y_t L_1(X_t^1|\Theta_1) - \gamma_t]d\Theta_1 d\gamma \geqslant 0 \\ \int q(\Theta_2,\gamma)[y_t L_2(X_t^2|\Theta_2) - \gamma_t]d\Theta_2 d\gamma \geqslant 0 \\ \int p(\Theta_1,\gamma)[L_1(U_i^1|\Theta_1) - \gamma_i]d\Theta_1 d\gamma \geqslant 0 \\ \int q(\Theta_2,\gamma)[L_2(U_i^2|\Theta_2) - \gamma_i]d\Theta_2 d\gamma \geqslant 0 \end{cases} \end{aligned} \tag{7}$$

In order to optimize Eq. (7), we use an iterative scheme for finding a solution to Eq. (7) which is similar with the one given in [17]. In the $m$th iteration, we successively update $p^{(m)}(\Theta_1, \gamma)$ and $q^{(m)}(\Theta_2, \gamma)$ by solving the following two problems (Eqs. (8) and (9)) and before the solution, we choose some initial value for $q^{(0)}(\Theta_2, \gamma)$ with $q_0(\Theta_2, \gamma)$ and this makes Eq. (8) be a standard MED problem.

$$\begin{aligned} & p^{(m)}(\Theta_1,\gamma) = argmin_{p^{(m)}(\Theta_1,\gamma)} KL(p^{(m)}(\Theta_1)||p_0(\Theta_1)) \\ & \quad + (1-\alpha)KL(p^{(m)}(\gamma)||p_0(\gamma)) + \alpha KL(p^{(m)}(\gamma)||q^{(m-1)}(\gamma)) \\ & s.t. \begin{cases} \int p^{(m)}(\Theta_1,\gamma)[y_t L_1(X_t^1|\Theta_1) - \gamma_t]d\Theta_1 d\gamma \geqslant 0 \\ \int p^{(m)}(\Theta_1,\gamma)[L_1(U_i^1|\Theta_1) - \gamma_i]d\Theta_1 d\gamma \geqslant 0 \end{cases} \end{aligned} \tag{8}$$

$$\begin{aligned} & q^{(m)}(\Theta_2,\gamma) = argmin_{q^{(m)}(\Theta_2,\gamma)} KL(q^{(m)}(\Theta_2)||q_0(\Theta_2)) \\ & \quad + (1-\alpha)KL(q^{(m)}(\gamma)||q_0(\gamma)) + \alpha KL(q^{(m)}(\gamma)||p^{(m)}(\gamma)) \\ & s.t. \begin{cases} \int q^{(m)}(\Theta_2,\gamma)[y_t L_2(X_t^2|\Theta_2) - \gamma_t]d\Theta_2 d\gamma \geqslant 0 \\ \int q^{(m)}(\Theta_2,\gamma)[L_2(U_i^2|\Theta_2) - \gamma_i]d\Theta_2 d\gamma \geqslant 0 \end{cases} \end{aligned} \tag{9}$$

The Lagrangian of Eq. (8) can be written as

$$\begin{aligned} L = & \int p^{(m)}(\Theta_1) log \frac{p^{(m)}(\Theta_1)}{p_0(\Theta_1)} d\Theta_1 \\ & + (1-\alpha)\int p^{(m)}(\gamma) log \frac{p^{(m)}(\gamma)}{p_0(\gamma)} d\gamma + \alpha \int p^{(m)}(\gamma) log \frac{p^{(m)}(\gamma)}{q^{(m-1)}(\gamma)} d\gamma \\ & - \sum_{t=1}^{N} \int p^{(m)}(\Theta_1,\gamma)\lambda_{1,t}^{(m)}[y_t L_1(X_t^1|\Theta_1) - \gamma_t]d\Theta_1 d\gamma \\ & - \sum_{i=1}^{L} \int p^{(m)}(\Theta_1,\gamma)\phi_{1,i}^{(m)}[L_1(U_i^1|\Theta_1) - \gamma_i]d\Theta_1 d\gamma \\ = & \int p^{(m)}(\Theta_1,\gamma) log \frac{p^{(m)}(\Theta_1,\gamma)}{p_0(\Theta_1)[p_0(\gamma)]^{1-\alpha}[q^{(m-1)}(\gamma)]^{\alpha}} \\ & - \sum_{t=1}^{N} \int p^{(m)}(\Theta_1,\gamma)\lambda_{1,t}^{(m)}[y_t L_1(X_t^1|\Theta_1) - \gamma_t]d\Theta_1 d\gamma \\ & - \sum_{i=1}^{L} \int p^{(m)}(\Theta_1,\gamma)\phi_{1,i}^{(m)}[L_1(U_i^1|\Theta_1) - \gamma_i]d\Theta_1 d\gamma \end{aligned} \tag{10}$$

**Table 1**
The codes of used Universum set construction ways.

| Code | Way | Code | Way | Code | Way |
|---|---|---|---|---|---|
| $\dot{U}_{1-1}$ | all-all-mid | $\dot{U}_{2-1}$ | unlabeled-all-mid | $\dot{U}_{3-1}$ | labeled-all-mid |
| $\dot{U}_{1-2}$ | all-near-mid | $\dot{U}_{2-2}$ | unlabeled-near-mid | $\dot{U}_{3-2}$ | labeled-near-mid |
| $\dot{U}_{1-3}$ | all-far-mid | $\dot{U}_{2-3}$ | unlabeled-far-mid | $\dot{U}_{3-3}$ | labeled-far-mid |
| $\dot{U}_{1-4}$ | all-near-self | $\dot{U}_{2-4}$ | unlabeled-near-self | $\dot{U}_{3-4}$ | labeled-near-self |
| $\dot{U}_{1-5}$ | all-far-self | $\dot{U}_{2-5}$ | unlabeled-far-self | $\dot{U}_{3-5}$ | labeled-far-self |

where $\lambda_1^{(m)} = \{\lambda_{1,t}^{(m)}\}$ and $\phi_1^{(m)} = \{\phi_{1,i}^{(m)}\}$ are sets of nonnegative Lagrange multipliers, one for each classification constraint. Then we take the partial derivative of Eq. (10) with respect to $p^{(m)}(\Theta_1, \gamma)$, set it to be zero and get the solution of $p^{(m)}(\Theta_1, \gamma)$ as below.

$$p^{(m)}(\Theta_1, \gamma) = \frac{1}{Z_1^{(m)}(\lambda_1^{(m)})} p_0(\Theta_1)[p_0(\gamma)]^{1-\alpha}[q^{m-1}(\gamma)]^{\alpha} \tag{11}$$

$$e^{\sum_{t=1}^{N}\lambda_{1,t}^{(m)}[y_t L_1(X_t^1|\Theta_1)-\gamma_t]+\sum_{i=1}^{L}\phi_{1,i}^{(m)}[L_1(U_i^1|\Theta_1)-\gamma_i]}$$

where $Z_1^{(m)}(\lambda_1^{(m)})$ is the normalization constant and $e$ is the exponential operation. According to [17] said, $\lambda_1^{(m)}$ is set by finding the unique maximum of the following concave objective function.

$$J_1^{(m)}(\lambda_1^{(m)}) = -logZ_1^{(m)}(\lambda_1^{(m)}) \tag{12}$$

Then for Eq. (9), we adopt the same analysis and obtain the solution of $q^{(m)}(\Theta_2, \gamma)$ as below.

$$q^{(m)}(\Theta_2, \gamma) = \frac{1}{Z_2^{(m)}(\lambda_2^{(m)})} q_0(\Theta_2)[q_0(\gamma)]^{1-\alpha}[p^{m}(\gamma)]^{\alpha} \tag{13}$$

$$e^{\sum_{t=1}^{N}\lambda_{2,t}^{(m)}[y_t L_2(X_t^2|\Theta_2)-\gamma_t]+\sum_{i=1}^{L}\phi_{2,i}^{(m)}[L_2(U_i^2|\Theta_2)-\gamma_i]}$$

where $\lambda_2^{(m)} = \{\lambda_{2,t}^{(m)}\}$ and $\phi_2^{(m)} = \{\phi_{2,i}^{(m)}\}$ are another sets of nonnegative Lagrange multipliers. Like $\lambda_1^{(m)}, \lambda_2^{(m)}$ is set by finding the unique maximum of the following concave objective function.

$$J_2^{(m)}(\lambda_2^{(m)}) = -logZ_2^{(m)}(\lambda_2^{(m)}) \tag{14}$$

As [17] said, after each iteration, we calculate the relative error between values of Eq. (12) from two successively iterations and that of Eq. (14), respectively, and utilize them for determining convergence. When the relative errors

$$\frac{J_1^{(m)}(\lambda_1^{(m)}) - J_1^{(m-1)}(\lambda_1^{(m-1)})}{J_1^{(m-1)}(\lambda_1^{(m-1)})} \tag{15}$$

and

$$\frac{J_2^{(m)}(\lambda_2^{(m)}) - J_2^{(m-1)}(\lambda_2^{(m-1)})}{J_2^{(m-1)}(\lambda_2^{(m-1)})} \tag{16}$$

are both less than some tolerance $\epsilon$, the iteration ends and finally we can get the optimal $p(\Theta_1)$ and $q(\Theta_2)$. Then for a test instance $X_r = \{X_r^1, X_r^2\}$, we can use $\hat{y_1} = sign\left(\int p(\Theta_1)L_1(X_r^1|\Theta_1)d\Theta_1\right)$ to get the class label of $X_r$ from the first view while $\hat{y_2} = sign\left(\int p(\Theta_2)L_2(X_r^2|\Theta_2)d\Theta_2\right)$ is used to compute the class label of $X_r$ from the second view. Finally, we can use $\hat{y} = sign\left(\omega_1 \int p(\Theta_1)L_1(X_r^1|\Theta_1)d\Theta_1 + \omega_2 \int p(\Theta_2)L_2(X_r^2|\Theta_2)d\Theta_2\right)$ to get the class label of $X_r$ in the whole sample space without considering the view spaces where $\omega_1$ and $\omega_2$ are the weights of views respectively.

## 4. Experiments

In order to validate that SSMVMED has a better performance, we conduct our experiments on five parts. They are (1) comparison for test accuracy, (2) comparison for training time and computational complexity, (3) comparison between different additional unlabeled instances generation approaches in terms of test accuracy and training time, (4) application to estimation problem, and (5) application to regression problem.

The used data sets and compared learning machines are given in related experimental contents and we will show the common experimental settings and the setting for our SSMVMED. Concretely speaking, (a) for each data set, we run the compared learning machine for 10 times and 70% instances for each data set are chosen in random for training and the remaining are chosen for test. In the training set, we randomly choose 30% as the labeled instances while the left 70% are treated as unlabeled instances; (b) for SSMVMED, (b-1) when we compute the weights of views and features, the setting can be referred to [37], i.e., exponential parameter $p$ is selected from the set $\{1, 2, \ldots, 30\}$ and balancing parameter $\beta$ is initialized to be 0.1, the cluster number equals to be the class number, the maximum iteration times is 300, $\omega_v = \frac{1}{V}$, and $\tau_l^v = \frac{1}{d^v}, \forall l = 1, 2, \ldots, d^v, \forall v = 1, 2, \ldots, V$; (b-2) when we generate the additional unlabeled instances, the used approaches can refer to Table 1. In terms of the number of instances which locates farthest or nearest from the center $K$ is selected from the set $\{1, .., N_{e-max}\}$ where $N_{e-max} = N_t - N_{max}$. $N_t$ is the total number of training instances and $N_{max}$ is the number of instances from largest training class. For example, a training data set consists of three classes, one has 100 instances, another has 120 instances, and the third has 140 instances, then $N_{e-max} = 220$; (b-3) since we have given the solution of $p(\Theta_1)$ and $q(\Theta_2)$ and furthermore, we also have given the way to predict a test instance with the optimal $p(\Theta_1)$ and $q(\Theta_2)$, thus it is found that the key of solution is the expressions of $L_1(X_t^1|\Theta_1)$ and $L_2(X_t^2|\Theta_2)$. In our practical experiments, we adopt linear classifier assumptions, i.e., $L_1(X_t^1|\Theta_1) = \theta_1^T X_t^1 + b_1$ and $L_2(X_t^2|\Theta_2) = \theta_2^T X_t^2 + b_2$. Furthermore, in our experiments, we refer to [17] and suppose that $p_0(\Theta_1, \gamma) = p_0(\Theta_1)p_0(\gamma) = p_0(\theta_1)p_0(b_1)p_0(\gamma)$ and $q_0(\Theta_2, \gamma) = q_0(\Theta_2)q_0(\gamma) = q_0(\theta_2)q_0(b_2)q_0(\gamma)$ where $p_0(\theta_1)$ and $q_0(\theta_2)$ are satisfied with Gaussian distributions with mean 0 and standard deviation $I, p_0(b_1)$ and $q_0(b_2)$ are set to non-informative Gaussian distributions, and $p_0(\gamma) = \prod_{t=1}^{N} p_0(\gamma_t) \prod_{i=1}^{L} p_0(\gamma_i)$ and $q_0(\gamma) = \prod_{t=1}^{N} q_0(\gamma_t) \prod_{i=1}^{L} q_0(\gamma_i)$. Here, $p_0(\gamma_t) = q_0(\gamma_t) = \frac{c}{\sqrt{2\pi}} e^{-\frac{c^2}{2}(1-\gamma_t)^2}$ and $p_0(\gamma_i) = q_0(\gamma_i) = \frac{c}{\sqrt{2\pi}} e^{-\frac{c^2}{2}(1-\gamma_i)^2}$ which are Gaussian priors with mean 1 that encourages large margins where $c$ is selected from the set $\{2^1, 2^2, \ldots, 2^{15}\}$. Finally, for SSMVMED, the tolerance $\epsilon$ is initialized to be0.001.

### 4.1. Comparison for test accuracy

First, we will show the effectiveness of the proposed SSMVMED on test accuracy. Since related experiments have been validated that SMVMED outperforms MED, MVMED, and AMVMED [17] and the effectiveness of MED-related learning machines have also been validated compared with the traditional multi-view learning machines due to MED-related learning machines consider the uncertainties over model parameters [13–16], thus the used compared learning machine is SMVMED here. Experimental setting of SMVMED can be referred to the one of SSMVMED (see (b-3)) since the parameters of SSMVMED include the ones of SMVMED. Moreover, the used data sets are six multi-view data sets Course, Citeseer, Cora, WebKB, NewsGroup, and Reuters. Information of them is summarized in Table 2 where $C$ represents the class number. (1) Course data set [10] is used to describe web pages and we want to predict whether the given web page is a course page or not; (2) Citeseer and Cora data sets both consist of 4 views and we choose view content and cites here [38]; (3) WebKB data set consists of web pages collected from four universities: Cornell, Texas, Wisconsin and Washington which have 5 categories, i.e., student, project, course, stuff and faculty. Data in WebKB are described with two views: content and citation. We treat WebKB in four separate data sets grouped by universities [39]; (4) NewsGroup data set [40] is of six groups extracted from the 20-Newsgroup dataset, i.e., M2, M5,

**Table 2**
Brief data set description.

| Data set | $C$ | $N$ | $V$ | $d^v$ ($v = 1, 2, \ldots, V$) |
|---|---|---|---|---|
| Course | 2 | 1051 | 2 | 66, 5 |
| Citeseer | 6 | 3264 | 2 | 3703, 3264 |
| Cora | 7 | 2708 | 2 | 1433, 2708 |
| Cornell | 5 | 195 | 2 | 1703, 195 |
| Texas | 5 | 185 | 2 | 1703, 185 |
| Washington | 5 | 217 | 2 | 1703, 217 |
| Wisconsin | 5 | 262 | 2 | 1703, 262 |
| News-M2 | 2 | 1200 | 3 | 2000, 2000, 2000 |
| News-M5 | 5 | 500 | 3 | 2000, 2000, 2000 |
| News-M10 | 10 | 500 | 3 | 2000, 2000, 2000 |
| News-NG1 | 2 | 500 | 3 | 2000, 2000, 2000 |
| News-NG2 | 5 | 400 | 3 | 2000, 2000, 2000 |
| News-NG3 | 8 | 1000 | 3 | 2000, 2000, 2000 |
| Reuters | 6 | 1600 | 5 | 2000, 2000, 2000, 2000, 2000 |

M10, NG1, NG2, NG3. Every group contains 10 data sets, and we choose the first set for all six groups in our experiments (see Table 2). For each data set, there are three views, Partitioning Around Methods, Supervised Mutual Information, and Unsupervised Mutual Information; (5) In terms of Reuters, it is the abbreviation of Reuters RCV1/RCV2 Multilingual and this data set consists of machine translated documents which are written in five different languages [41,42]. These five languages are English (EN), French (FR), German (GR), Italian (IT), and Spanish (SP). Each language is treated as a view of this Reuters data set and each document can be translated from one language to another language. For this data set, the documents are also categorized into six different topics, i.e., six classes. They are C15, CCAT, E21, ECAT, GCAT, M11.

Table 3 shows the related experimental results about the comparison for test accuracy between SSMVMED and SMVMED. From this table, it is found that the proposed SSMVMED outperforms SMVMED in average since on 12 data sets, SSMVMED has a better performance. The average accuracy of SSMVMED has a 2% enhancement. Moreover, from the standard deviation values, it is found that the performance of SSMVMED is more stable than SMVMED.

**Table 3**
Test accuracies (average value ± std.) compared with SMVMED. Last row list the win/tie/lose counts of SSMVMED on all data sets with t-test against SMVMED at significance level 95%. The best performance on each data set is in bold. (★) indicates the sig-value with paired t-test.

| Data set | SSMVMED | SMVMED |
|---|---|---|
| Course | **0.957 ± 0.008** | 0.943 ± 0.011 (0.032) |
| Citeseer | **0.737 ± 0.008** | 0.714 ± 0.008 (0.041) |
| Cora | **0.827 ± 0.002** | 0.815 ± 0.010 (0.047) |
| Cornell | **0.788 ± 0.021** | 0.760 ± 0.041 (0.023) |
| Texas | **0.797 ± 0.014** | 0.785 ± 0.037 (0.031) |
| Washington | **0.826 ± 0.019** | 0.816 ± 0.039 (0.026) |
| Wisconsin | **0.884 ± 0.011** | 0.868 ± 0.023 (0.043) |
| News-M2 | 0.967 ± 0.003 | **0.972 ± 0.010** (0.057) |
| News-M5 | **0.989 ± 0.011** | 0.965 ± 0.015 (0.037) |
| News-M10 | **0.855 ± 0.003** | 0.829 ± 0.017 (0.033) |
| News-NG1 | 0.929 ± 0.001 | **0.931 ± 0.025** (0.049) |
| News-NG2 | **0.953 ± 0.006** | 0.912 ± 0.009 (0.012) |
| News-NG3 | **0.915 ± 0.006** | 0.901 ± 0.009 (0.016) |
| Reuters | **0.783 ± 0.001** | 0.753 ± 0.016 (0.033) |
| Average | **0.872 ± 0.008** | 0.855 ± 0.019 |
| W/T/L | SSMVMED vs. SMVMED | 13/1/0 |

Furthermore, in order to validate the difference between SSMVMED and SMVMED is significant, we adopt paired t-test [45] (paired t-test is different from t-test) and Nemenyi statistical test [46] for quantitative evaluation analysis. Paired t-test is used to analyze if the differences between two compared learning machines on one data set are significant or not. Then for Nemenyi statistical test, it is used to analyze if the differences between two compared learning machines on multiple data sets are significant or not. Nemenyi is different from another famous test, i.e., Friedman statistical test which is used to analyze if the differences between all compared learning machines on multiple data sets are significant or not. Since the number of compared learning machines here is two, thus we adopt Nemenyi statistical test rather than Friedman statistical test. In generally, the differences always indicate the ones in test accuracy. Thus, here we conduct quantitative evaluation analysis in terms of test accuracy.

(A) For paired t-test [45], we use sig-value to represent the significant differences of test accuracy. When sig-value is less than 0.05, it indicates that the compared two learning machines have a significant difference in the test accuracy on one data set. Furthermore, the difference is more significant when the sig-value is smaller. According to Table 3, we use (★) indicates the sig-value with the comparison between SSMVMED and SMVMED. From the table, we find that the difference between SSMVMED and SMVMED is significant in average.

(B) For Nemenyi statistical test, the performance of two learning machines on all data sets is significantly different if the corresponding average ranks differ by at least the critical difference

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \tag{17}$$

where critical value $q_\alpha$ is given in Table 4, $N$ is the number of data sets, $k$ is the number of learning machines.

**Table 4**
Critical values for the two-tailed Nemenyi test. Each critical value $q_\alpha$ is based on the studentized range statistic [46] divided by $\sqrt{2}$.

| No. learning machines | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| $q_{0.05}$ | 1.960 | 2.343 | 2.569 | 2.728 | 2.850 | 2.949 | 3.031 | 3.102 | 3.164 |
| $q_{0.10}$ | 1.645 | 2.052 | 2.291 | 2.459 | 2.589 | 2.693 | 2.780 | 2.855 | 2.920 |

According to Table 3, the average rank of SSMVMED 1.1429 is while the one of SMVMED is 1.8571. Then for Nemenyi statistical test, if $\alpha = 0.05$, the critical value $q_{0.05}$ is 1.960 (see Table 4) and the corresponding $CD$ is $1.960\sqrt{\frac{2\cdot(2+1)}{6\cdot 14}} = 0.5238$. Under such a case, since $1.1429 + 0.5238 = 1.6667 < 1.8571$, so the difference between the average rank of SSMVMED and the one of SMVMED is significant. When $\alpha = 0.10$, the critical value $q_{0.10}$ is 1.645 (see Table 4) and the corresponding $CD$ is $1.645\sqrt{\frac{2\cdot(2+1)}{6\cdot 14}} = 0.4396$. Under such a case, since $1.1429 + 0.4396 = 1.5825 < 1.8571$, so we get the similar conclusion.

In other words, with paired t-test and Nemenyi statistical test, we can validate the effectiveness of SSMVMED from the quantitative evaluation analysis aspect.

### 4.2. Comparison for training time and computational complexity

Here, we still adopt SMVMED for comparison about the training time and computational complexity. As we know, compared with SMVMED, our proposed SSMVMED has to generate additional unlabeled instances. Thus SSMVMED has to cost more training time theoretically and Table 5 validates that. From this table, we can find that SSMVMED costs more training time than SMVMED. But the extra time is not more than 10% which is acceptable for us. In other words, we can achieve better test accuracy with only a little extra time with SSMVMED adopted.

In order to validate the higher training time, we give the computational complexity of them theoretically. As we know, in SSMVMED, it consists of three steps. So here, we will discuss the computational complexities for different steps. For convenience, we let the number of labeled instances be $N$, the number of original unlabeled instances be $L$, the number of additional unlabeled instances be $U$.

For the first step, the computation focuses on $\omega_v$ and $\tau_l^v$. In terms of $\omega_v$, its computational complexity depends on $D_v$ and the computational complexity of $D_v$ is $O(M(N+L)((d^v)^2 + 2d^v))$. In terms of $\tau_l^v$, its computational complexity depends on $B_l^v$ which computational complexity is $O(M(N+L))$. Thus, for the first step, the computational complexity is $O(VM(N+L)((d^v)^2 + 2d^v)) + O(dM(N+L))$ where $d = \sum_{v=1}^{V} d^v, M$ is the number of clusters, and $V$ is the number of views. Here, for $d^v$ in $O(VM(N+L)((d^v)^2 + 2d^v)) + O(dM(N+L))$, we can select the largest $d^v$ and in fact, this selection won't influence too much.

For the second step, the computational complexity consists of three parts. For 'A', if we select 'all', the computational complexity is $O((N+L)dV)$; if we select 'unlabeled', the computational complexity is $O(LdV)$; if we select 'labeled', the computational complexity is $O(NdV)$. For 'B', if we select 'near' or 'far', the computational complexity is $O(N(N-1)dV/2)$; if we select 'all', we can omit the related computational complexity. For 'C', if we select 'mid', the computational complexity is $O(KdV)$ or $O((N+L)dV)$ which depends on the selection of 'B' and $K$ is the number of selected instances; if we select 'self', we can also omit the related computational complexity. As a result, the total computational complexity of the second step is $[min\{O(LdV),\ O(NdV)\},\ max\{O((N+L)dV) + O(N(N-1)dV/2) + O(KdV),\ 2O((N+L)dV)\}]$.

**Table 5**
Comparison about average training time (in seconds).

| Data set | SSMVMED | SMVMED |
|---|---|---|
| Course | 2.249 | 2.170 |
| Citeseer | 290.545 | 277.717 |
| Cora | 212.896 | 201.727 |
| Cornell | 91.870 | 87.120 |
| Texas | 87.832 | 84.980 |
| Washington | 2.933 | 2.677 |
| Wisconsin | 1.099 | 1.010 |
| News-M2 | 30.396 | 29.913 |
| News-M5 | 258.455 | 241.120 |
| News-M10 | 275.072 | 263.533 |
| News-NG1 | 199.226 | 183.350 |
| News-NG2 | 176.051 | 173.270 |
| News-NG3 | 259.336 | 259.153 |
| Reuters | 234.180 | 222.730 |

For the third step, the computational complexity is similar with the one of SMVMED. In this step, the computational complexity of SSMVMED is $O((N+U)^2)$ while the one of SMVMED is $O((N+L)^2)$.

Totally speaking, the computational complexity of SSMVMED is $O(VM(N+L)((d^v)^2 + 2d^v)) + O(dM(N+L)) + [min\{O(LdV), O(NdV)\},\ max\{O((N+L)dV) + O(N(N-1)dV/2) + O(KdV), 2O((N+L)dV)\}] + O((N+U)^2)$ and compared with SMVMED, the extra computational complexity is $O(VM(N+L)((d^v)^2 + 2d^v)) + O(dM(N+L)) + [min\{O(LdV),\ O(NdV)\}, max\{O((N+L)dV) + O(N(N-1)dV/2) + O(KdV),\ 2O((N+L)dV)\}] + O((N+U)^2) - O((N+L)^2)$. From this result, it looks like that our proposed SSMVMED seems to cost more training time, but compared with $O((N+U)^2)$ and $O((N+L)^2), O(VM(N+L)((d^v)^2 + 2d^v)) + O(dM(N+L)) + [min\{O(LdV), O(NdV)\}, max\{O((N+L)dV) + O(N(N-1)dV/2) + O(KdV), 2O((N+L)dV)\}]$ won't influence too much due to $N+U$ and $N+L$ is always much larger than $d, V, M, K$.

Now according to the theoretical analysis about computational complexity, we can also validate the conclusion derived from this experimental item that with SSMVMED adopted, we can achieve better test accuracy with only a little extra time.

### 4.3. Comparison between different additional unlabeled instances generation approaches in terms of test accuracy and training time

Here, we will discuss the difference between additional unlabeled instances generation approaches which are given in Table 1 in terms of test accuracy and training time. Figs. 1 and 2 show the test accuracy and training time of different approaches given in Table 1 on the used data sets. In these figures, orders in 'Approaches in Table 1' represent the approaches in Table 1, i.e., 1-$\dot{U}_{1-1}$, 2-$\dot{U}_{1-2}$, 3-$\dot{U}_{1-3}$, 4-$\dot{U}_{1-4}$, 5-$\dot{U}_{1-5}$, 6-$\dot{U}_{2-1}$, 7-$\dot{U}_{2-2}$, 8-$\dot{U}_{2-3}$, 9-$\dot{U}_{2-4}$, 10-$\dot{U}_{2-5}$, 11-$\dot{U}_{3-1}$, 12-$\dot{U}_{3-2}$, 13-$\dot{U}_{3-3}$, 14-$\dot{U}_{3-4}$, and 15-$\dot{U}_{3-5}$. Then from these figures, it is found that different additional unlabeled instances generation approaches bring different performances. We find that for $\dot{U}_{1-x}$ $(x = 1, 2, 3, 4, 5)$ approaches, they can get better test accuracies compared with the $\dot{U}_{y-x}$ approaches where $x = 1, 2, 3, 4, 5$ and $y = 2, 3$ in average. Moreover, we find that take 'mid' strategy for experiments has a better test accuracy than 'self'. In terms of training time, approaches with taking all instances as the center bring longer training time due to all instances rather than labeled or unlabeled instances are used to compute the center.

### 4.4. Application to estimation problem

Here, we apply our SSMVMED to estimation problem so as to validate its effectiveness. The estimation problem discussed here aims to forecast the demographic trends which has also been discussed in [43]. For the forecast, USs Census population data is used. In the experiments, we predict the demographic distribution in the year 2010 based on the historical data in years 2000 and 2006. The prediction is then compared with the actual Census data in the year 2010. This setting is same as the one in [43]. In [43], the authors conducted the estimation experiments with three stages

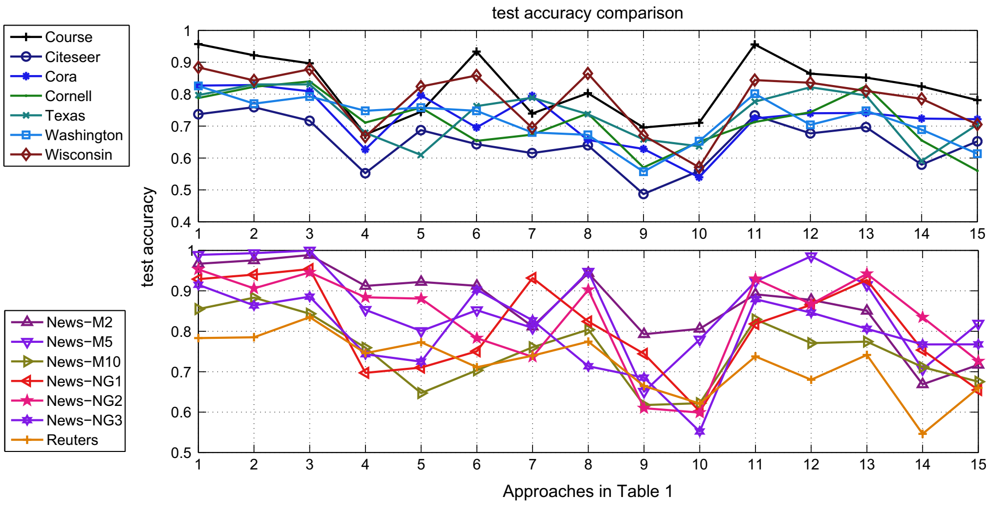

**Fig. 1.** Comparison between different additional unlabeled instances generation approaches in terms of test accuracy.

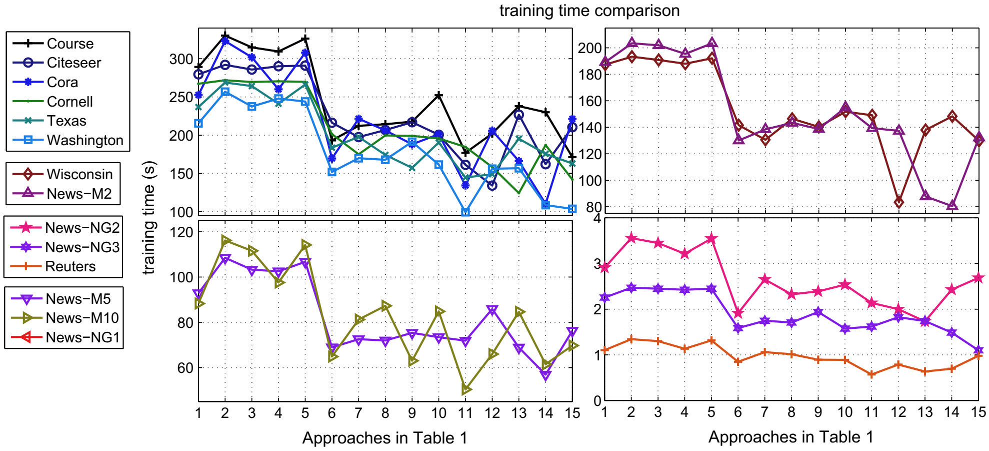

**Fig. 2.** Comparison between different additional unlabeled instances generation approaches in terms of training time.

and showed the experimental results with 12 figures, while since the limitation of paper length, we won't show all the results. For convenience, we will only show the average difference between the estimation and the real values of age in year 2010. SMVMED and the proposed method in [43] (we call it EDT) are used for comparison. Fig. 3 shows the related experiments. From this figure, it is found that compared with SMVMED and EDT, the predicted age distribution of our SSMVMED can accord with the real age distribution to a large extent.

### 4.5. Application to regression problem

Here, we apply SSMVMED to regression problem. The regression problem discussed here aims to estimate full joint distributions from incomplete information which has also been discussed in [44]. In [44], authors proposed generalized cross entropy model (GCEM) and estimated the distribution of Singapore household profile (http://www.singstat.gov.sg) with the joint probability distribution of the household dwelling type (HD), household size (HS) and home ownership (HO) measures. In those experiments, authors conducted the experiments on three different cases: (1) pure entropy with constraints, (2) minimum discrimination information without constraints, and (3) minimum discrimination information with constraints. Moreover, authors adopted (a) accuracy heat maps to compare the accuracy of the three cases, (b) KullbackCLeibler (KL) distance to compare the estimated joint distribution and the observed one, and (c) Linfoots measures to compare a wide range of spatiotemporal signals from brain waves to human dynamics. For accuracy heat maps, if the estimated heat map is close to the true heat map, we say the estimation is better. For KL distance, the smaller the KL distance between any two distributions is, the closer are their profiles. For Linfoots measures, it has three indexes, $C$ measures the relative structural content, $F$ looks at the fidelity or peak alignment, and $Q$ reflects the correlation quality [44]. If these three indexes are more closer to 1, then we say the estimated distribution is more closer to the true distribution. From those experiments, it was found that case 1 and case 2 had comparable performance and case 3 had a best performance.

In our experiments, for the limitation of paper length, we won't conduct all experiments given in [44]. We will only adopt GCEM and SSMVMED for comparison on the Singapore household profile when case 3 is considered. In order to show the results clearly, we combine the results of accuracy heat maps (here, the differences between estimated heat maps and the true ones are given), KL distance, and Linfoots measures together and only show that whether SSMVMED brings a better performance or not. Table 6 shows the results in a simple manner. From this table, it is found that with SSMVMED, the error of estimation is decreased and the difference

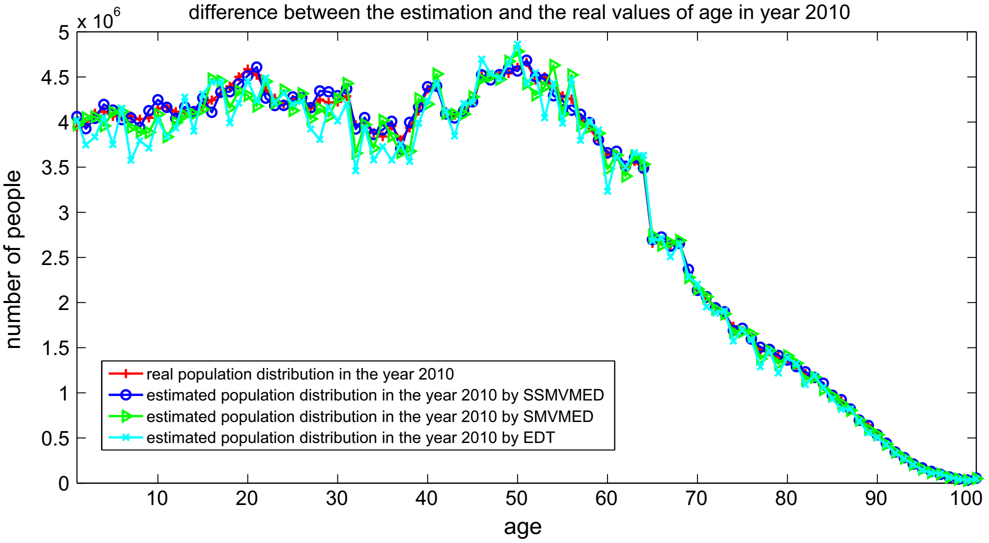

**Fig. 3.** The predicted age distribution of US population for the year 2010 based on the population data in the years 2000 and 2006 with SSMVMED, SMVMED, and EDT used.

**Table 6**
Comparison between GCEM and SSMVMED on Singapore household profile with minimum discrimination information with constraints in terms of the results of heat maps, KL distance, and Linfoots measures.

| Measures | | SSMVMED | SMVMED |
|---|---|---|---|
| Accuracy heat maps | | 0.017 | 0.012 |
| KL distance | | 0.0039 | 0.0028 |
| Linfoots measures | C | 1.016 | 1.003 |
| | Q | 0.9991 | 0.9994 |
| | F | 1.007 | 1.004 |

between the estimated distribution and the true distribution is more smaller. In other words, SSMVMED brings a better regression performance.

## 5. Conclusions

Traditional multi-view learning machines do not consider the uncertainties over model parameters. Thus maximum entropy discrimination (MED) and its extended versions multi-view maximum entropy discrimination (MVMED) and alternative MVMED (AMVMED) are developed for this issue. While for processing multi-view data sets, they only use the hard margin consistency principle that the decision of margin parameter $\gamma$ is related to classifier parameter $\Theta$ directly. As we know, the decision always be indirectly in practice. So soft margin consistency based multi-view maximum entropy discrimination (SMVMED) has been proposed. Although related experiments have validated the effectiveness of SMVMED, it is only adaptive to supervised problems. Indeed, in real-world, most data sets are semi-supervised, namely, the data sets consist of labeled instances and unlabeled instances. So this paper extends the model of SMVMED to the semi-supervised problems and develop a semi-supervised SMVMED (SSMVMED). Furthermore, in order to get more useful discriminant information, we propose some schemes to generate more additional unlabeled instances. Moreover, these generated additional unlabeled instances will also be used in the model of SSMVMED along with the original labeled and unlabeled instances so that the performance of a learning machine can be boosted. Related experiments on multi-view data sets from different aspects have validated the effectiveness of SSMVMED theoretically and empirically. From the experiments, it is found that (1) compared with SMVMED, the average test accuracy of SSMVMED has a 2% enhancement; (2) SSMVMED costs more training time than SMVMED and the extra time is not more than 10% which is acceptable for us; (3) in terms of the generation of additional unlabeled instances, 'mid' strategy has a better test accuracy than 'self' and taking all instances to get the center brings a better test accuracy as well; (4) with SSMVMED, the applications to estimation problem and regression problem will be more feasible.

## Acknowledgment

This work is supported by (1) Natural Science Foundation of Shanghai – China under Grant No. 16ZR1414500 (2) National Natural Science Foundation of China – China under Grant Nos. 61602296, 41701523, 51575336 and the authors would like to thank their supports.

## References

[1] R. Bach, G.R. Lanckriet, M.I. Jordan, Multiple kernel learning, conic duality, and the SMO algorithm, in: Proceedings of the 21st International Conference on Machine Learning, 2004, pp. 6–13.

[2] C. Cortes, M. Mohri, A. Rostamizadeh, Two-stage learning kernel algorithms, in: Proceedings of the 27th International Conference on Machine Learning, 2010, pp. 239–246.

[3] A. Rakotomamonjy, F. Bach, S. Canu, Y. Grandvalet, SimpleMKL, J. Mach. Learn. Res. 9 (2008) 2491–2521.

[4] M. Kloft, U. Brefeld, S. Sonnenburg, A. Zien, Non-sparse Regularization and Efficient Training with Multiple Kernels, 2010. Available from: arxiv preprint <arXiv:1003.0079>.

[5] M. Gönen, E. Alpaydin, Localized multiple kernel learning, in: Proceeding of the 25th International Conference on Machine Learning, 2008, pp. 352–359.

[6] G. Ye, D. Liu, I.H. Jhuo, S.F. Chang, Robust late fusion with rank minimization, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2012, pp. 3021–3028.

[7] A. Iosifidis, A. Tefas, N. Nikolaidis, I. Pitas, Multi-view human movement recognition based on fuzzy distances and linear discriminant analysis, Comput. Vis. Image Underst. 116 (3) (2012) 347–360.

[8] J. Rupnik, J. Shawe-Taylor, Multi-view canonical correlation analysis, in: Proceeding of Slovenian KDD Conference on Data Mining Data Warehouses, 2010, pp. 1–4.

[9] X. Yin, Q. Huang, X. Chen, Multiple view locality preserving projections with pairwise constraints, Commun. Syst. Inform. Technol. 100 (2011) 859–866.

[10] A. Blum, T. Mitchell, Combining labeled and unlabeled data with co-training, in: Eleventh Conference on Computational Learning Theory, 1998, pp. 92–100.

[11] M.L. Zhang, Z.H. Zhou, Cotrade: confident co-training with data editing, IEEE Trans. Syst., Man, Cybernet., Part B: Cybernet. 41 (6) (2011) 1612–1626.

[12] V. Sindhwani, P. Niyogi, M. Belkin, A co-regularization approach to semi-supervised learning with multiple views, in: Proceeding of ICML workshop on Learning With Multiple Views, 2005, pp. 74–79.

[13] T. Jaakkola, M. Meila, T. Jebara, Maximum entropy discrimination, Adv. Neural Inform. Process. Syst. 12 (2000) 470–476.

[14] T. Jebara, Machine Learning: Discriminative and Generative, Kluwer Academic, 2004.

[15] S.L. Sun, G.Q. Chao, Multi-view maximum entropy discrimination, in: Proceedings of the 23rd International Joint Conference on Artificial Intelligence, 2013, pp. 1706–1712.

[16] G.Q. Chao, S.L. Sun, Alternative multiview maximum entropy discrimination, IEEE Trans. Neural Networks Learn. Syst. 99 (2015) 1–12.

[17] L. Mao, S.L. Sun, Soft margin consistency based scalable multi-view maximum entropy discrimination, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, 2016, pp. 1839–1845.

[18] G. Tzortzis, A. Likas, Kernel-based weighted multi-view clustering, in: 2012 IEEE 12th International Conference on Data Mining, 2012, pp. 675–684.

[19] S.L. Sun, Q.J. Zhang, Multiple-view multiple-learner semi-supervised learning, Neural Process. Lett. 34 (2011) 229–240.

[20] M.Q. Deng, C. Wang, Q.F. Chen, Human gait recognition based on deterministic learning through multiple views fusion, Pattern Recogn. Lett. 78 (C) (2016) 56–63.

[21] F. Wu, X.Y. Jing, X.G. You, D. Yue, R.M. Hu, J.Y. Yang, Multi-view low-rank dictionary learning for image classification, Pattern Recogn. 50 (2016) 143–154.

[22] S.H. Zhu, X. Sun, D.L. jin, Multi-view semi-supervised learning for image classification, Neurocomputing 208 (2016) 136–142.

[23] H.Y. Wang, X. Wang, J. Zheng, J.R. Deller, H.Y. Peng, L.Q. Zhu, W.G. Chen, X.L. Li, R.J. Liu, H.J. Bao, Video object matching across multiple non-overlapping camera views based on multi-feature fusion and incremental learning, Pattern Recogn. 47 (12) (2014) 3841–3851.

[24] R. Sheikhpour, M.A. Sarram, S. Gharaghani, M.A.Z. Chahooki, A survey on semi-supervised feature selection methods, Pattern Recogn. 64 (2017) 141–158.

[25] T.P. Xie, N.M. Nasrabadi, I.A.O. Hero, Semi-supervised multi-sensor classification via consensus-based multi-view maximum entropy discrimination, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2015, pp. 1936–1940.

[26] A.N. Erkan, Y. Altun, Y.W. Teh, M. Titterington, Semi-supervised learning via generalized maximum entropy, in: International Conference on Artificial Intelligence and Statistics, 2010, pp. 209–216.

[27] G.Q. Chao, S.L. Sun, Semi-supervised multitask learning via self-training and maximum entropy discrimination, in: Neural Information Processing, Springer Berlin/Heidelberg, 2012.

[28] V. Vapnik, S. Kotz, Estimation of Dependences based on Empirical Data, Springer, 1982.

[29] V. Cherkassky, W.Y. Dai, Empirical study of the Universum SVM learning for high-dimensional data, Lect. Notes Comput. Sci. 5768 (2009) 932–941.

[30] D. Zhang, J. Wang, L. Si, Document clustering with Universum, in: International Conference on Research and Development in Information Retrieval, 2011, pp. 873–882.

[31] B. Peng, G. Qian, Y.Q. Ma, View-invariant pose recognition using multilinear analysis and the Universum, Adv. Visual Comput. 5359 (2008) 581–591.

[32] C. Shen, P. Wang, F. Shen, H. Wang, Uboost: boosting with the Universum, IEEE Trans. Pattern Anal. Mach. Intell. 34 (4) (2012) 825–832.

[33] X.H. Chen, S.C. Chen, H. Xue, Universum linear discriminant analysis, Electron. Lett. 48 (22) (2012) 1407–1409.

[34] Z. Wang, Y.J. Zhu, W.W. Liu, Z.H. Chen, D.Q. Gao, Multi-view learning with Universum, Knowl.-Based Syst. 70 (2014) 376–391.

[35] J. Weston, R. Collobert, F. Sinz, L. Bottou, V. Vapnik, Inference with the Universum, in: The 23rd International Conference on Machine Learning, 2006, pp. 1009–1016.

[36] D.L. Liu, Y.J. Tian, R.F. Bie, Y. Shi, Self-Universum support vector machine, Pers. Ubiquit. Comput. 18 (2014) 1813–1819.

[37] Y.M. Xu, C.D. Wang, J.H. Lai, Weighted multi-view clustering with feature selection, Pattern Recogn. 53 (2016) 25–35.

[38] P. Sen, G.M. Namata, M. Bilgic, L. Getoor, B. Gallagher, T. Eliassi-Rad, Collective classification in network data, AI Mag. 29 (3) (2008) 93–106.

[39] H.J. Ye, D.C. Zhan, Y. Miao, Y. Jiang, Z.H. Zhou, Rank consistency based multi-view learning: a privacy-preserving approach, in: ACM International on Conference on Information and Knowledge Management, 2015, pp. 991-1000.

[40] G. Bisson, C. Grimal, Co-clustering of multi-view datasets: a parallelizable approach, in: Proceedings of the IEEE 12th International Conference on Data Mining, 2012, pp. 828-833.

[41] M.R. Amini, N. Usunier, C.Goutte, Learning from multiple partially observed viewsan application to multilingual text categorization, in: Neural Information Processing Systems (NIPS), 2009, pp. 28–36.

[42] http://multilingreuters.iit.nrc.ca/ReutersMultiLingualMultiView.htm.

[43] G.Q. Li, D.X. Zhao, Y. Xu, S.H. Kuo, H.Y. Xu, N. Hu, G.S. Zhao, C. Monterola, Entropy based modelling for estimating demographic trends, Plos One 10.9 (2015) e0137324.

[44] H.Y. Xu, S.H. Kuo, G.Q. Li, E.F.T. Legara, D.X. Zhao, C.P. Monterola, Generalized Cross Entropy Method for estimating joint distribution from incomplete information, Physica A 453 (2016) 162–172.

[45] V.N. Vapnik, Statistical Learning Theory, Wiley Interscience Press, New York, 1998.

[46] J. Demsar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.