Bayesian Maximum Margin Clustering

Bo Dai, Baogang Hu
NLPR/LIAMA
Institute of Automation, Chinese Academy of Sciences
Beijing, P.R.China
$\{$*bdai, hubg*$\}$*@nlpr.ia.ac.cn*

Gang Niu
Department of Computer Science
Tokyo Institute of Technology
Tokyo, Japan
gang@sg.cs.titech.ac.jp

***Abstract*—Most well-known discriminative clustering models, such as *spectral clustering* (SC) and *maximum margin clustering* (MMC), are non-Bayesian. Moreover, they merely considered to embed domain-dependent prior knowledge into data-specific kernels, while other forms of prior knowledge were seldom considered in these models. In this paper, we propose a *Bayesian maximum margin clustering* model (BMMC) based on the *low-density separation* assumption, which unifies the merits of both Bayesian and discriminative approaches. In addition to stating prior distribution on functions explicitly as traditional Gaussian processes, special prior knowledge can be embedded into BMMC implicitly via the *Universum* set easily. Furthermore, it is much easier to solve a BMMC than an MMC since the integer variables in the optimization are eliminated. Experimental results show that the BMMC achieves comparable or even better performance than state-of-the-art clustering methods and solving BMMC is more efficiently.**

***Keywords*-Clustering; Bayesian; Maximum Margin Principle; Universum;**

I. Introduction

Clustering has been studied extensively in machine learning and data mining. Generally speaking, the target of clustering is to find out how data are organized. *Generative* approaches seem natural for clustering. Given $\mathbf{x}$ representing the instance and y denoting the cluster assignment of an instance, generative methods model the class-conditional densities $p(\mathbf{x}|y)$ and $p(y)$ explicitly, e.g., mixture model [1]. The contribution of unlabeled data is obtained by improving the fit of $p(\mathbf{x})$. Prediction is given by posterior probability which is obtained through the Bayes theorem. Obviously, Bayesian approaches are convenient to be integrated in generative models. However, generative models add restrict assumptions on $p(\mathbf{x}|y)$ and $p(y)$, the clustering results may be unconvincing when these assumptions are violated.

Recently, *discriminative* clustering methods have attracted more and more renewed attentions, e.g., spectral clustering (SC) [2][3] and maximum margin clustering (MMC) [4][5][6][7]. Both of them are learning a discriminative model from unlabeled data by connecting the posterior probability $P(y|\mathbf{x})$ and $p(\mathbf{x})$ directly through the *low-density separation* assumption. To express low-density separation preference, all of these algorithms learn in a non-Bayesian way that employ loss functions as the implementation of the assumption. Recent research in computer vision [8][6] demonstrates the power of discriminative clustering methods. Besides its usefulness in practice, these methods are also theoretically appealing [9][10].

Although non-Bayesian approaches provide additional flexibility and convenience in designing loss functions, they neither capture the uncertainty in predictions, nor have the ability to learn the hyperparameters. These limitations can be skirted via Bayesian approaches. However, how to extend Bayesian approaches to implement the low-density separation assumption in the unsupervised setting has not been considered yet.

To address the dilemma mentioned above, we propose a Bayesian low-density separation clustering method in this paper, named *Bayesian maximum margin clustering* (BMMC), which combines the advantages of both the Bayesian learning and the discriminative model. More specifically, BMMC is based on the *Bayesian support vector machine* (Bayesian SVM) [11]. However, the same as the other discriminative models, the Bayesian SVM is incapable in the unsupervised setting. Following the way in [12] for semi-supervised problems, we adopt the augmented model to extend the learning ability of Bayesian SVM to unlabeled data and to capture the spirit of the low-density separation. The proposed BMMC is the counterpart of the existing MMC in the Bayesian framework. Compared with the existing MMC, our model eliminates the integer variables, and thus makes the optimization much easier to solve.

On the other hand, prior knowledge plays an important role in machine learning, especially in clustering problems. From the regularization perspective, the low-density separation assumption can be viewed as a general priori for clustering. Both of SC and MMC are learning under the priori where domain-dependent prior knowledge can be embedded into kernel, and it seems that these algorithms are flexible enough to different problems in practice. Nevertheless, choosing a suitable kernel for a special problem is equivalent to selecting a prior distribution of functions [13] and is indeed a job for oracle. Although several kernel learning methods have been proposed recently, they are still limited in the learning environment in which side information is sufficient. In contrast, the proposed model allows the users to encode prior knowledge via a set of data, which is called *Universum* [14]. Universum, introduced by Vapnik firstly through *maximal contradiction on Universum principle* [15][16] and applied to semi-supervised learning [17][18], has not been utilized in unsupervised problems yet. Defining a Universum set is approximately equivalent to specifying a kernel, but much easier. The Universum can be integrated into our model as a natural extension. It is notable that our model can also handle the classical clustering problems in which no Universum is available.

The remainder of this paper is organized as follows. After introducing the clustering problem with the Universum, MMC and Bayesian SVM in the next section, we present the probabilistic model and the inference method in Section 3. The relationship between MMC and the log-posteriori of BMMC is discussed in Section 4. Two different perspectives of Universum are also explained. To validate our model, experimental results are reported in Section 5. The last section gives the conclusion.

A. *Contributions*

- A low-density separation clustering method is proposed as the counterpart of MMC in the Bayesian framework.
- Universum in which prior knowledge embeds can be utilized in our model more naturally.
- The relationship between MMC and BMMC is studied. Our model eliminates the integer variables; thus much easier to solve.

II. Preliminaries

A. *Clustering with the Universum*

Since our model can utilize the *Universum*, the problem which we could solve is more general than the classical clustering algorithm. We first define the problem as well as the notations used in this paper formally.

Assume we are given a data set $\mathcal{D}$ consisting of n instances $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ drawn *i.i.d* from a certain distribution $\mathcal{S}$. Here $\mathbf{x}_i \in \mathbb{R}^d$ $(i = 1, 2, \ldots, n)$ is the input feature vector. In addition to these instances, a collection of examples, denoting as $\{\mathbf{x}_1^*, \mathbf{x}_2^*, \ldots, \mathbf{x}_{|\mathfrak{U}|}^*\}$, known of belonging to the same domain as the problem of interest but not belonging to either class may be also available. The set $\mathfrak{U}$ is called *Universum set* in which meaningful prior information about the task at hand can be embedded. Without loss of generality, we assume the label of instance $y_i \in \{-1, +1\}$ as MMC [4]. Two different approaches for multi-class clustering extension are discussed later. Our task is to assign each instance in $\mathcal{D}$ a label y or an uncertainty prediction. In the absence of the Universum, this problem is the same as classical clustering and can also be solved by BMMC.

B. *Maximum Margin Clustering*

Large margin methods, e.g., support vector machine (SVM) and adaboost, have been applied in many supervised tasks successfully. Given the training instances and corresponding labels, the goal of SVM is to find a discriminant function $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$ by solving the following optimization:

$$\begin{aligned} \min_{\mathbf{w},\xi} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{s.t.} \quad & y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, n \\ & \xi_i \geq 0 \end{aligned}$$

where ξ is the slack variable and C balances the regularization and the loss function.

The dual of the above optimization is:

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \mathbf{e}^T\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}^T(K \circ \mathbf{y}\mathbf{y}^T)\boldsymbol{\alpha} \\ \text{s.t.} \quad & \boldsymbol{\alpha}^T\mathbf{y} = 0, \quad 0 \leq \boldsymbol{\alpha} \leq C \end{aligned} \tag{1}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_n]$, $\mathbf{e} = [1, 1, \ldots, 1]$ and $\circ$ denotes the elementwise product between matrics. By kernel trick, the linear function can be easily extended to non-linear form.

Maximum margin clustering, which is proposed by [4], can be viewed as an unsupervised extension of support vector machine. The key idea of maximum margin clustering is to find a labelling so that the obtained margin would be maximal over all candidate labellings. Based on this intuition, the optimization is written as

$$\begin{aligned} \max_{\boldsymbol{\alpha},\mathbf{y}} \quad & \mathbf{e}^T\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}^T(K \circ \mathbf{y}\mathbf{y}^T)\boldsymbol{\alpha} \\ \text{s.t.} \quad & \boldsymbol{\alpha}^T\mathbf{y} = 0, \quad 0 \leq \boldsymbol{\alpha} \leq C \\ & y_i \in \{-1, +1\} \end{aligned} \tag{2}$$

To prevent the meaningless solution that assigns all the instances into the same class, the class-balance constraint, $-\ell \leq \mathbf{e}^T\mathbf{y} \leq \ell$, had been introduced by [4].

The optimization (2) is difficult to solve because it contains integer variables and is non-convex. To make the optimization tractable, [4] relaxed it as a semi-definite program (SDP). Valizadegan and Jin [5] proposed the generalized maximum margin clustering that reduce the n^2 optimization variables to n, thus made a significant reduction of computational cost. To make the MMC more practical, many optimization methods have been proposed [6][7][19]. However, all of these methods are limited in the non-Bayesian domain. Moreover, they relaxed the integer optimization technically. In contrast, our model avoids the integer optimization from Bayesian approach naturally.

C. *Bayesian Support Vector Machine*

The Bayesian support vector machine (Bayesian SVM) [11] provides a probabilistic interpretation of the margin concept in supervised setting. Our model focuses on dividing instances through low-density region without labels by extending the Bayesian SVM. Thus, it is worth establishing the background of the Bayesian SVM.

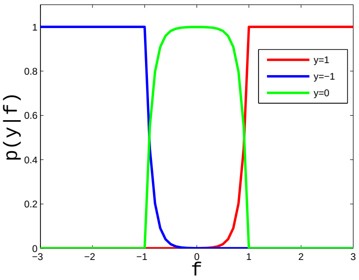


Figure 1. Bayesian Support Vector Machine noise model

The Bayesian SVM can be viewed as a special case of the Gaussian process classifier. The only difference between the Bayesian SVM and the traditional Gaussian process classifier is their noise model. Traditional Gaussian process classifiers adopt logistics or probit function while the Bayesian SVM defines a distribution which has some special properties, e.g., sparsity and facility in introducing the Universum.

More formally, the Bayesian SVM adopts the probability of obtaining output y given $\mathbf{x}$ as

$$p(y=\pm 1|f(\mathbf{x}))=Z(C)\exp(-Cl(y,f(\mathbf{x})))$$

where $l(y,f(\mathbf{x}))$ is the hinge loss, $Z(C)=\frac{1}{1+exp(-2C)}$ is chosen to make the probabilities less than 1. It is still necessary to introduce a *null category* (labeled by $y=0$) to make the noise model consistent,

$$p(y=0|f(\mathbf{x}))=1-\sum_{y=\pm 1}p(y|f(\mathbf{x}))$$

The null category can be considered as a probabilistic interpretation of the 'margin' concept in standard SVM. Noticing the Gaussian process priori over function with covariance $K(x,x')$, the log-posteriori of the model formulates as

$$\begin{aligned}&\ln p(f|\mathcal{D})\propto\ln p(f)+\ln p(y|f(\mathbf{x}))\\ =\ &-\frac{1}{2}\sum_{i,j}^{n}f(\mathbf{x}_i)K^{-1}(\mathbf{x}_i,\mathbf{x}_j)f(\mathbf{x}_j)-C\sum_{i}^{n}l(y_i,f(\mathbf{x}_i))\end{aligned}$$

Since the maximum achieves at $f^*(\mathbf{x})=\sum_i\alpha_i y_i K(\mathbf{x},\mathbf{x}_i)$, the maximum a posteriori (MAP) is identical to standard SVM.

III. Bayesian Maximum Margin Clustering

As studied in [20], the phenomenon that the Bayesian SVM cannot distinguish instances in the unsupervised setting is caused by the independence between f and unlabeled data. This means that the knowledge of instances $\mathbf{X}$ cannot affect the posterior function distribution when labels are unobserved. In this section, we begin with specifying the augmenting probabilistic model of Bayesian SVM following the way studied in [12] to handle unsupervised problems in the spirit of low-density separation assumption. Prediction and multi-class clustering extension methods are introduced lastly.

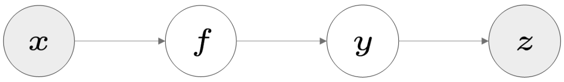


Figure 2. Graphical representation of Bayeisian maximum margin clustering model

A. Probabilistic Model

To make unlabeled data have an effect on the posterior distribution of f, we could restore the dependence by augmenting the model following [12]. Introducing an additional variable z which is the child of the label variable and always observed, the traditional Gaussian process transforms as shown in Figure 2. The shaded nodes can always be observed while the unshaded nodes are latent variables. Assuming the variable z is an indicator identifying whether the instance belongs to the Universum set or is an instance that needs to be assigned label, e.g., z takes the value 0 when the instance belongs to the Universum set, and vice versa, the dependence between f and unlabeled data has been established. Based on the definition of variable z and the problem defined in Section 2, we have $p(z_i=0|y_i=0)=1$ meaning that any samples from the Universum set must be observed. Meanwhile, we assign the probabilities of unlabeled data by

$$p(z_i=1|y_i)=\begin{cases}\gamma_+, & y_i=1,\\ \gamma_-, & y_i=-1,\\ 0, & y_i=0.\end{cases}$$

where $\gamma_+ + \gamma_-$ equals to 1 implicitly. Through modifying the value of γ_+ and γ_-, we can control the size of the two clusters similarly to the effect of class-balance constraint [4]. Thus, if instances belong to the Universum set, the posterior process is updated by $p(y|f(\mathbf{x}))$; otherwise, the instances need to be assigned labels and the process is updated by $p(z|f(\mathbf{x}))$ which can be computed as

$$p(z=1|f(\mathbf{x}))=\sum_{y=\{-1,1\}}p(z=1|y)p(y|f(\mathbf{x}))$$

Recall the special noise model defined in the Bayesian SVM:

$$p(y|f(\mathbf{x}))=\begin{cases}Z(C)\exp(-Cl(y,f(\mathbf{x}))), & y=\pm 1,\\ 1-\sum_{y=\pm 1}p(y|f(\mathbf{x})), & y=0.\end{cases}\qquad(3)$$

we obtain the effective likelihood function $\mathcal{L}'(f)$ by combining (3) into the augmenting model,

$$\begin{cases}\sum_{y=\pm 1}\gamma_y Z(C)\exp(-Cl(y,f(\mathbf{x}))), & z=1, \quad (4a)\\ 1-\sum_{y=\pm 1}p(y|f(\mathbf{x})), & y,z=0. (4b)\end{cases}$$

(4a) encourages f at each instance far away from margin leading to low-density separation and (4b) reflects the effect of prior knowledge incorporated in the Universa which helps locating margin. The derivations of Universum from two different perspectives will also be discussed in Section 4.

B. Prediction

To provide the confidence besides the labelling, we take a fully Bayesian approach that needs to compute the posterior distribution over the latent variables. However, the posteriori cannot be solved in an analytic way. We thus adopt Laplace's method to find an approximation. Doing a second order Taylor expansion of $\log p(\mathbf{f}|\mathcal{D}, \mathbf{Z}, \mathbf{X}^*, \mathbf{Y}^*)$ around the maximum, a Gaussian approximation of posteriori will be obtained. The log-posteriori of BMMC is non-convex. It is because of the intrinsic difficulty of clustering problem, i.e., the exchangeability of labels. For clustering, an assignment is equal to another one that exchanges labels with counterpart in the sense of distinction. Thus, one local minimum is enough and it is reasonable that we employ Laplace's method to inference approximately.

Based on the effective likelihood function (4a) ,(4b) and the priori over $\mathbf{f}$ specified through Gaussian process, the log-posteriori of latent variable $\mathbf{f}$ is formulated as

$$\begin{aligned}\mathcal{L} = &\sum_{i=1}^{n} \log \left(\sum_{y_i=-1,1} \gamma_{y_i} \mathrm{exp}(-Cl(y_i, f_i)) \right) \\ &+ \sum_{k=1}^{|\mathfrak{U}|} \log \left(1 - \sum_{y_k=\pm 1} p(y_k|f_k) \right) - \frac{1}{2} \sum_{i,j=1}^{n+|\mathfrak{U}|} f_i K^{-1}(\mathbf{x}_i, \mathbf{x}_j) f_j \end{aligned} \tag{5}$$

We will prove that the optimization in MMC proposed by [4] is an approximation of (5) without considering the Universum later. However, the existing MMC methods add labels of instances, which are integer variables, into optimization while our model eliminates them making the optimization much easier to solve.

To find a nontrivial maximum of (5), we utilize Newton's method with special initial solution inspired by recent development of deep learning which claimed that the 2^{nd}-order optimization method can handle highly non-linear criteria [21]. We also implement an EM style algorithm to optimize this criterion. We neglect the EM algorithm because of the page limitation. The performances of these two optimize algorithms are similar in classical clustering setting. Differentiating (5) w.r.t. $\mathbf{f}$, we obtain $\nabla \mathcal{L} = \mathbf{g} - K^{-1}\mathbf{f}$ and $\nabla\nabla\mathcal{L} = -W - K^{-1}$, where $\mathbf{g} = [g_i]$, $i = 1, 2, \ldots, n + |\mathfrak{U}|$

$$g_i = \begin{cases} \frac{\sum_{y_i=\pm 1} \gamma_{y_i} \varphi(y_i, f_i) t(y_i)}{\sum_{y_i=\pm 1} \gamma_{y_i} \varphi(y_i, f_i)}, & z_i = 1, \\ \frac{-\sum_{y=\pm 1} \varphi(y_i, f_i) t(y_i)}{(1 - \sum_{y=\pm 1} \varphi(y_i, f_i))}, & y_i = 0. \end{cases} \tag{6}$$

and $W = diag(w_i)$, $i = 1, 2, \ldots, n + |\mathfrak{U}|$, w_i is computed by

$$\begin{cases} \frac{-\prod_{y_i=\pm 1} \gamma_{y_i} \varphi(y_i, f_i)(t(1) - t(-1))^2}{(\sum_{y_i=\pm 1} \gamma_{y_1} \varphi(y_i, f_i))^2}, & z_i = 1, \\ \frac{\sum_{y_i=\pm 1} \varphi(y_i, f_i) t^2(y_i) - \prod_{y_i=\pm 1} \varphi(y_i, f_i)(t(1) - t(-1))^2}{(1 - \sum_{y_i=\pm 1} \varphi(y_i, f_i))^2}, & y_i = 0 \end{cases} \tag{7}$$

$\varphi(y_i, f_i)$ denotes $\exp(-Cl(y_i, f_i))$, and $t(y_i) = Cy_i$ when $y_i f_i \leq 1$, otherwise, $t(y_i) = 0$.

Algorithm 1: Bayesian Maximum Margin Clustering

```
Input: $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ (clustering dataset),
       $\mathfrak{U} = \{\mathbf{x}_1^*, \mathbf{x}_2^*, \ldots, \mathbf{x}_{|\mathfrak{U}|}^*\}$ (Universum set),
       $K$ (kernel matrix)
Output: $\mathbf{y}$ or $p(\mathbf{y}|\mathcal{D}, \mathfrak{U})$

begin
    Initialize $\mathrm{f}_0$ as Remark 2
    repeat
        compute $\mathbf{g}$ as (6)
        compute $W$ as (7)
        $\mathbf{b} = W\mathbf{f} + \mathbf{g}$
        $L = Cholsky(I + W^{\frac{1}{2}} K W^{\frac{1}{2}})$
        $\mathbf{f}^{new} = K(\mathbf{b} - W^{\frac{1}{2}} L^T \setminus (L \setminus W^{\frac{1}{2}} K\mathbf{b}))$
    until convergence;
    compute $\mathbf{y} = sgn(\mathbf{f})$ or
    $p(\mathbf{y}|\mathcal{D}, \mathfrak{U}) = \int p(y_i|f_i, z_i{=}1) q(f_i|\mathcal{D}, \mathbf{Z}, \mathbf{X}^*, \mathbf{Y}^*) df$
end
return $\mathbf{y} = sgn(\mathbf{f})$ or $p(\mathbf{y}|\mathcal{D}, \mathfrak{U})$
```

In each iteration, we update $\mathbf{f}$ by (8) until a convergence is reached,

$$\mathbf{f}^{new} = (K^{-1} + W)^{-1}(W\mathbf{f} + \mathbf{g}) \tag{8}$$

After finding the maximum $\hat{\mathbf{f}}$, we specify the Laplace approximation of the posteriori

$$q(\mathbf{f}|\mathcal{D}, \mathbf{Z}, \mathbf{X}^*, \mathbf{Y}^*) \sim \mathcal{N}(\hat{\mathbf{f}}, (K^{-1} + W)^{-1}) \tag{9}$$

The uncertain prediction is making by computing

$$\begin{aligned} &p(y_i|\mathcal{D}, \mathbf{Z}, \mathbf{X}^*, \mathbf{Y}^*, x_i, z_1 = 1) \\ \simeq &\int p(y_i|f_i, z_i = 1) q(f_i|\mathcal{D}, \mathbf{Z}, \mathbf{X}^*, \mathbf{Y}^*) df \end{aligned} \tag{10}$$

***Remark** 1:* In (8), computing $(K^{-1} + W)^{-1}$ costs too much and may cause numerical instability. As specified in [13] that

$$(K^{-1} + W)^{-1} = K - KW^{\frac{1}{2}} B^{-1} W^{\frac{1}{2}} K$$

it is trivial to get

$$\mathbf{f}^{new} = K(\mathbf{b} - W^{\frac{1}{2}} L^T \setminus (L \setminus W^{\frac{1}{2}} K\mathbf{b}))$$

where $\mathbf{b} = W\mathbf{f} + g$, $B = (I + W^{\frac{1}{2}} K W^{\frac{1}{2}})$, and L is the Cholesky decomposition of B. By this trick, we can achieve numerical stability rather than updating by (8) directly.

***Remark** 2:* To find a better minimal, a good initial solution is needed. In our implementation, the initial solution $\mathbf{f}_0 = [sgn(\mathbf{v}_2 - \frac{1}{n}\mathbf{v}_2^T\mathbf{1}), 0, \ldots, 0]$ is used, $\mathbf{v}_2$ being the second smallest eigenvector of $K_{n\times n}^{-1}$. However, computing $K_{n\times n}^{-1}$ is costly and not stable. We adopt the Laplacian matrix $\mathscr{L}$ as the pseudo-inverse of $K_{n\times n}$ [5]. This initialization is inspired by the pre-training phase in deep learning that finds a point near the maximum of hidden layer greedily.

***Remark** 3:* When we just focus on finding the labels and do not need the confidences, we may compare $l(1, \hat{f}_i)$ and $l(-1, \hat{f}_i)$ to get a guess directly. Compared with MMC and GMMC whose computational complexities are $\mathcal{O}(n^{6.5})$ and $\mathcal{O}(n^{4.5})$, our method costs $\mathcal{O}(Tn^3)$ in such situation, where T is the number of iterations. If we use the pseudo-inverse of K specified above and subgradient descent method to optimize the MAP (5) directly, the computational complexity could reduce to $\mathcal{O}(Tn^2)$.

***Remark** 4:* An important property of BMMC is that it can provide both uncertain predictions and labels. For different clustering tasks targets, different output of the BMMC could be utilized. Moreover, the uncertain predictions open doors to apply BMMC to active learning by querying the most uncertain instances.

C. Multi-Class Clustering Extension

Although the discussion about the Bayesian maximum margin clustering focused on the two-class clustering setting above, the algorithm can be extended to multi-class clustering problems easily. In this part, we propose two different methods.

Firstly, a heuristic method [8][6] can be adopted to extend the BMMC to the multi-class clustering setting. We can execute the two-class clustering method recursively. By this approach, a hierarchical clustering algorithm is formed.

The other method is modifying the noise model as [22]. Assume that there is k Gaussian processes, $\mathbf{f}_l$, each one corresponding to a class and $\mathbf{y}_l$ is the class indicator, the noise model $p(\mathbf{y}_l|\mathbf{f}_l(\mathbf{x}))$ is

$$\begin{cases} \delta(f_{lk} > f_{li} + \epsilon, \forall i \neq k)\mathcal{N}(\mathbf{f}_l, \mathbf{I}_k), & y_{lk} = 1, \\ 1 - \sum_k \delta(f_{lk} > f_{li} + \epsilon, \forall i \neq k)\mathcal{N}(\mathbf{f}_l, \mathbf{I}_k), & \sum_k y_{lk} = 0. \end{cases}$$

Thus, the $\mathcal{L}'(f)$ is

$$\begin{cases} \sum_k \gamma_k \delta(f_{lk} > f_{li} + \epsilon, \forall i \neq k)\mathcal{N}(\mathbf{f}_l, \mathbf{I}_k), & z_l = 1, \\ 1 - \sum_k \delta(f_{lk} > f_{li} + \epsilon, \forall i \neq k)\mathcal{N}(\mathbf{f}_l, \mathbf{I}_k), & \mathbf{y}_l, z_l = 0. \end{cases}$$

In this paper, we mainly focus on the first extension method and the results presented in experiment are all based on hierarchical clustering. The second extension method is our future work.

IV. Justification

It is natural to investigate the relationship between maximum margin clustering [4] and the proposed model. As the MAP of Bayesian SVM is equivalent to SVM [11], we will prove that MMC is an approximation of the log-posteriori of the proposed model in this section. In addition, two different derivations of the Universum will be discussed here to clarify its role in learning process.

A. Relationship between MMC and BMMC

***Theorem** 1:* Maximum margin clustering is an approximation of (5) without considering the Universum.

Proof: Without considering the Universum, the effective likelihood is defined as

$$\mathcal{L}'(f) = \sum_{y=\pm 1} \gamma_y Z(C) \exp(-Cl(y, f(\mathbf{x})))$$

Thus, the log-posteriori transforms as

$$\begin{aligned} \mathcal{L} &= \sum_{i=1}^{n} \log \left(\sum_{y_i=-1,1} \gamma_{y_i} \exp(-Cl(y_i, f(\mathbf{x}_i))) \right) \\ &- \frac{1}{2} \sum_{i,j=1}^{n} f(\mathbf{x}_i) K^{-1}(\mathbf{x}_i, \mathbf{x}_j) f(\mathbf{x}_j) \end{aligned} \tag{11}$$

For convenience, we denote $l(1, f(\mathbf{x}_i))$ as ξ_{i+} and $l(-1, f(\mathbf{x}_i))$ as ξ_{i-}. Noticing that the first term on the right hand of (11) is the *soft maximum function* which is an approximation of the max-function, we have

$$\begin{aligned} \arg\max_f \mathcal{L} &\approx \arg\min_f \frac{1}{2} \sum_{i,j=1}^{n} f(\mathbf{x}_i) K^{-1}(\mathbf{x}_i, \mathbf{x}_j) f(\mathbf{x}_j) \\ &+ C \sum_{i}^{n} \min(\xi_{i+} - \log \gamma_+, \xi_{i-} - \log \gamma_-) \end{aligned}$$

which is equivalent to

$$\begin{aligned} \min_{f,\mathbf{y}} \quad & \frac{1}{2} \sum_{i,j=1}^{n} f(\mathbf{x}_i) K^{-1}(\mathbf{x}_i, \mathbf{x}_j) f(\mathbf{x}_j) + \\ & C \sum_{y_i=1} (\xi_i - \log \gamma_+) + C \sum_{y_i=-1} (\xi_i - \log \gamma_-) \\ \text{s.t.} \quad & y_i f(\mathbf{x}_i) \geq 1 - \xi_i, i = 1, 2, \ldots, n \\ & \mathbf{y} \in \{-1, +1\}^n \end{aligned} \tag{12}$$

Set $\gamma_+ = \gamma_-$ and realize $f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i)$, (12) transforms as

$$\begin{aligned} \min_{\vec{\alpha},\mathbf{y}} \quad & \frac{1}{2} \sum_{i,j=1}^{n} y_i y_j \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) + C \sum_{y_i} \xi_i \\ \text{s.t.} \quad & y_i f(\mathbf{x}_i) \geq 1 - \xi_i, i = 1, 2, \ldots, n \\ & \mathbf{y} \in \{-1, +1\}^n \end{aligned} \tag{13}$$

Recall the primary form of MMC,

$$\begin{aligned} \min_{\mathbf{w},\mathbf{y} \in \{-1,+1\}^n} \quad & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{y_i} \xi_i \\ \text{s.t.} \quad & y_i f(\mathbf{x}_i) \geq 1 - \xi_i, i = 1, 2, \ldots, n \end{aligned} \tag{14}$$

Substituting *KKT* condition $\mathbf{w} = \sum_{i}^{n} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i)$ into the optimization, we obtain (13). ■

***Corollary** 2:* From regularization perspective, the second term on the right hand of (5) can be viewed as another loss function which implements the *maximal contradiction* *on Universum principle*, different from [14] which adopts ϵ-insensitive loss. Thus, (5) is an implementation of *Maximum Margin Clustering with Maximal Contradiction on Universum principle*.

B. Influence of the Universum

In this section, we discuss the details about the influence of the Universum in clustering problems to clarify its role in learning process. Although the Universum derived from two different perspectives, the same conclusion about the effect of Universum is achieved: incorporating prior knowledge about the discriminant function.

Inference with the Universum could be derived from *maximal contradiction on Universum principle* [14]. This principle, introduced by Vapnik [15][16] firstly, is a structural risk minimization (SRM) principle which is parallel to the well-known *maximal margin principle* but more flexible. The principle prefers the equivalence function class that makes more different predictions on Universum set. Given a Universum set, the corresponding prior distribution about functions has been specified. Defining a prior function distribution is a direct way to construct a structure on the set of admissible functions. However, finding a suitable domain-dependent function distribution is very hard. Embedding priori into the Universum set makes the priori choosing problem much easier.

Rather than being introduced by a novel SRM principle, Universum in the proposed model is just a natural extension of the null category. As presented in [11][12], the null category is introduced to make the probability $p(y|f)$ consistent. Based on the interpretation of null category, i.e., the range of margin, we can claim the influence of the Universum: providing the prior knowledge about the range of decision boundary location. From this perspective, a Universum selection criterion is clear as *Remark 5*. A similar selection criterion is also obtained by rigorous mathematical analysis in [23].

***Remark** 5:* The nearer the mean of Universa locates to the margin, and the tighter Universa are, the more helpful they are.

Of course one can point out the Universa directly when has some knowledge about the boundary location. In this paper, we consider two ways to generate the Universum from the dataset which is needed to be separated for experiment following [14][18]:

- Create artificial instances by generating each feature according to the data set empirical distribution
- Create artificial instances nearby the mean of given dataset.

From the discussion above, the influence of the Universum is clear. The Universum provides a convenient way to impose the domain-dependent prior knowledge on the proposed model while most of the state-of-the-art algorithms have not considered yet. However, the Universum is not a requisite and the proposed model can also handle clustering problems without the Universum.

Although we verified the helpfulness of these two Universa generating methods on some special datasets in Section 5, these two Universum set generating methods are just general strategies. As [14][23] claimed, the Universum needs to be chosen quite carefully in order to be helpful. Thus, the Universa choice method is still an open problem for different tasks.

V. Experiments

To demonstrate that the Bayesian maximum margin clustering method (BMMC) is efficient and effective, we design a series of experiments on synthetic and real-world benchmark datasets. For the datasets that the instances with different labels form different clusters, evaluating clustering on such datasets is reasonable [24]. For comparison, the k-means algorithm, the generalized maximum margin clustering (GMMC) [5], the normalized spectral clustering (NC) [2] and the proposed method are implemented in Matlab. The CVX [25] package is employed to solve SDP involved in GMMC. In the second part, we demonstrate the effect of Universum in small-sample clustering problems. At last, time costs are listed.

A. Classical Clustering

Firstly, we examine our method on some synthetic data sets with RBF kernel. We extend the BMMC to multi-class clustering problems by top-down hierarchical approach [8][6]. The kernel width σ is set to 0.08 for all the synthetic data sets expect the last one which is set to 0.16, and the $\gamma_+ = \gamma_-$. Experimental results are illustrated in Figure 3. The results of the proposed BMMC on four classical synthetic two-class clustering datasets are illustrated in the first line. The second line are results of some multi-class clustering problems. It is obvious that although the proposed BMMC is focused on two-class problems originally, top-down hierarchical approach can extend our method to multi-class clustering problems efficiently.

Because the clustering ability of divisive hierarchical clustering methods for multi-class clustering problems is based on their discriminative ability on two-class clustering problems, we focus on comparing the performances in the two-class clustering setting on real-world datasets. Thus, we follow the evaluation methods of [4][5][6][19] that focused on two-class clustering problems. Besides the best performance comparison on benchmark following these literatures, we also verify the stability and convergence of BMMC by experiments which means that the solution of BMMC does not change much with respect to the sampling process and converges to a certain labelling [26].

Benchmarks of handwritten digits and text, `MNIST`, `USPS` and `20Newsgroups`, are taken for performance evaluation. It has been demonstrated that the instances with different

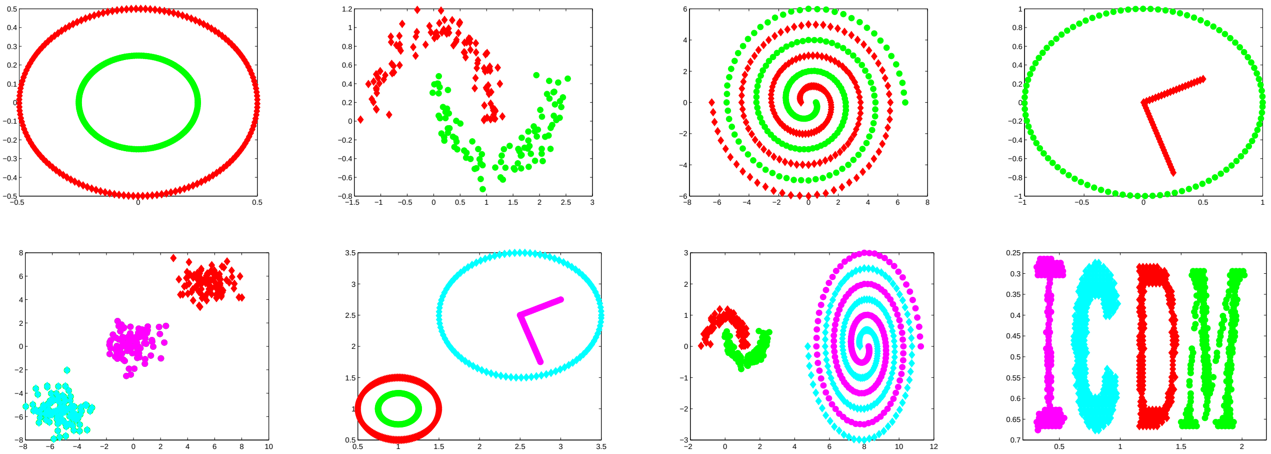


Figure 3. Illustration on some synthetic data sets. Different colors denote different clusters.

labels form different clusters in handwritten digits and text datasets [27]. Thus, they are suitable for evaluating the clustering methods.

Handwritten Digits In our experiments, we use 1 vs. 7, 3 vs. 8, 5 vs. 8, 8 vs. 9 on MNIST and USPS for difficult discrimination visually. Each digit in MNIST is represented by a 784 dimensional vector while in USPS is represented by a 256 dimensional vector. Among the experiments, we adopt RBF kernel for all the methods. For NC and GMMC, the kernel width is tuned beforehand from $\{10^{-2}, \ldots, 10^{3}\}$. For NC, the number of nearest neighbors k of adjacency matrix is tuned from 3 to 12. For GMMC, the C_δ is chosen from $\{10^{0}, \ldots, 10^{6}\}$. Although we can adopt type-II maximum likelihood to learn the parameters of BMMC, we tuned the parameters of BMMC in the same range as in NC for a fair comparison. We fix C_e in GMMC and C in BMMC. The best results and the average performance of 10-trials are reported in Figure 4 and 5 where the error is measured following [4][5][6][19]. Since the performance of GMMC is not stable, the average performance of GMMC is not reported. We do not perform GMMC when $n > 700$ because its computational complexity is too high.

Text clustering We present the experimental result on the 20Newsgroups data set, in which the instances have 26214 features. The classes which have most instances are selected. We choose kernel from the cosine kernel and RBF kernel to have better performance for all the methods. The parameters are tuned from the same sets for NC, GMMC and BMMC. The results are reported in Figure 6. The performances of BMMC and NC are similar on this dataset. Maybe the reason is that the NC achieves the global minimum and BMMC trapped in the same optimum.

Results From the comparison with NC and GMMC, both the best result and the average accuracy of the proposed method achieve comparable or even better performance in many situations on the image and text clustering problems. Moreover, with the number of instances increasing, the average performance of the proposed BMMC converges. The more instances, the better performance BMMC achieves. The stability of the BMMC could be verified by the variance of each 10-trials performance on random sampling subsets. The variances of BMMC on most subsets is comparable to NC which is known as a stable clustering algorithm [10]. Thus, we demonstrate that the convergence and stability of the proposed BMMC empirically.

B. *Universum in Small-Sample Clustering*

In this part, we evaluate the effect of Universum in small-sample clustering. The solution of BMMC here is got by EM algorithm. First, a synthetic data set and two different Universum sets are constructed. Considering clustering on two Gaussian distributions centered at $(-1.5, -1.5)$ and $(+1.5, +1.5)$ respectively, we generate two different Universum sets to verify *Remark 5*. One is around the origin and the other one is far from the boundary. The results are illustrated in Figure 7. Figure 7(a) is the ground truth of the instances. Figure 7(b) is the result of BMMC. Figure 7(c) and 7(d) are the results of BMMC with different Universum sets respectively. Obviously, the Universa around the mean of instances provides correct information about the range of margin and is much more helpful for the clustering task than Figure 7(d). Error prior knowledge contained in Universa in Figure 7(d) even debases the performance of BMMC.

Next, we examine the effect of Universum on real-world datasets. To examine the effect of prior knowledge closely, we reduce the data set scale. We select one-half of the features from MNIST and 20Newsgroups respectively. For each trial, 100 instances are chosen randomly from the most two difficult problems in MNIST and 20Newsgroups, and 20 Universa are generating randomly. We verify the two approaches for constructing Universum set mentioned above. The instances in $\mathfrak{U}_{mean}$ are generated nearby the mean of given dataset, while the features of the instances in $\mathfrak{U}_{gen}$ are generated according to the whole datasets empirical

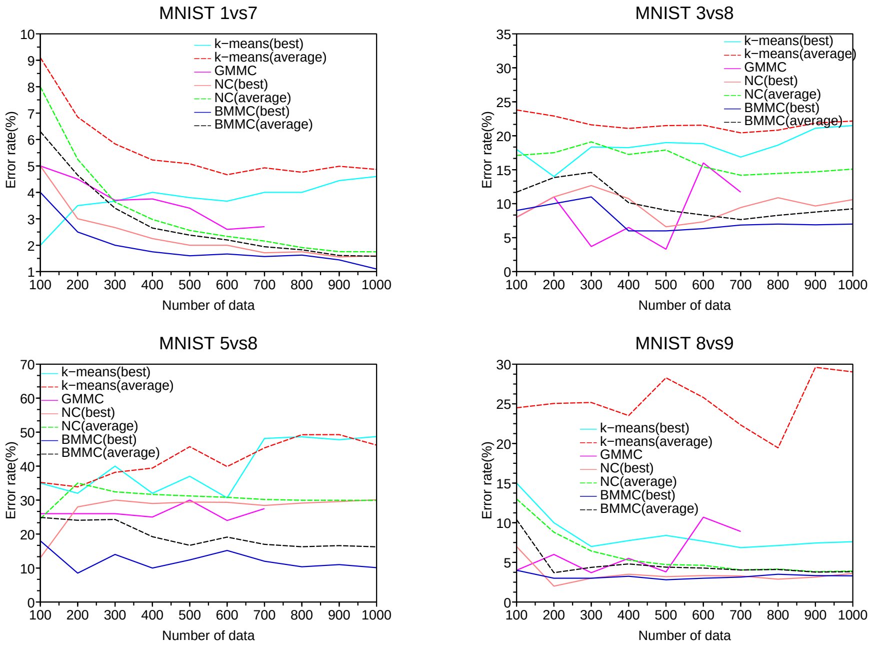


Figure 4. Empirical results on MNIST with the number of instances increasing.

distribution. The results are showed in Table I.

In the small-sample datasets, the performances by adding the Universum chosen by these two criteria are similar. We cannot assert which one is better. Like adding Universa under supervised and semi-supervised setting in [14][18], some progress are made in clustering problem, especially in handwritten digits dataset. However, the improvements on text clustering are not significant. The same phenomenon is observed in [14] on RCV1. Maybe it is caused by these two Universum choice strategies are just general criteria, and they cannot reflect the location of margin for different topic texts. Thus they are not suitable for text clustering problem. We also observed the phenomenon, described in [14], that when dataset is large, the instances provide enough information about the 'margin' themselves and the influence of the Universum diminishes.

For special tasks, the Universum choice is indeed a important work. Generating Universa carelessly may even hurt the performance. We will investigate this phenomenon in our future work.

C. Speed

Speed is another important issue besides accuracy. SDP based MMC solving methods are computationally expensive thus not practical. Because of the need of running GMMC, we examine the time cost of each algorithm on several medium datasets. Heart and Ionosphere are adopted, which contain 270 instances and 351 instances, respectively. We also record the running time on subset of MNIST and 20Newsgroups. We select 400 instances from the pair 3 vs. 8, 5 vs. 8 in MNIST randomly and 600 instances from the pair 6 vs.14 and 9 vs. 10 in 20Newsgroups randomly. The empirical running time is showed in **Table II**.

Table II
LIST OF RUNNING TIME AND ERROR RATE COMPARISON.

	dataset	k-means	NC	GMMC	BMMC
Time (sec.)	Heart(Stalog)	0.0037	0.3533	13.03	0.7485
	Ionosphere	0.0044	0.7991	26.84	1.504
	MNIST 5 vs. 8	0.2896	1.413	29.99	2.417
	MNIST 3 vs. 8	0.2392	1.438	23.84	2.386
	20-Newsgroups 6 vs. 14	5.383	5.278	79.65	9.047
	20-Newsgroups 9 vs. 10	5.626	5.181	82.78	9.212
Error Rate (%)	Heart(Stalog)	38.0	33.3	33.7	**32.8**
	Ionosphere	28.8	31.1	**25.3**	29.6
	MNIST 5 vs. 8	32	29	25	**10**
	MNIST 3 vs. 8	18.3	10.8	6.3	**6.0**
	20-Newsgroups 6 vs. 14	26	13.6	13.8	**8.8**
	20-Newsgroups 9 vs. 10	18.5	7.7	8.3	**7.2**

VI. CONCLUSION

In this paper, we proposed a probabilistic model which extends the maximum margin clustering method to the Bayesian framework. We proved that MMC is an approximation of the log-posteriori of BMMC without considering the Universum. Compared with the existing MMC, the proposed model can provide not only a guess but also the confidence when it is necessary. The existence of the integer variables in the optimization, i.e., labels of instances, is one of the intrinsic difficulties of the MMC. They are eliminated in our model making the optimization easier to solve. Moreover, our model can utilize Universum naturally which makes the

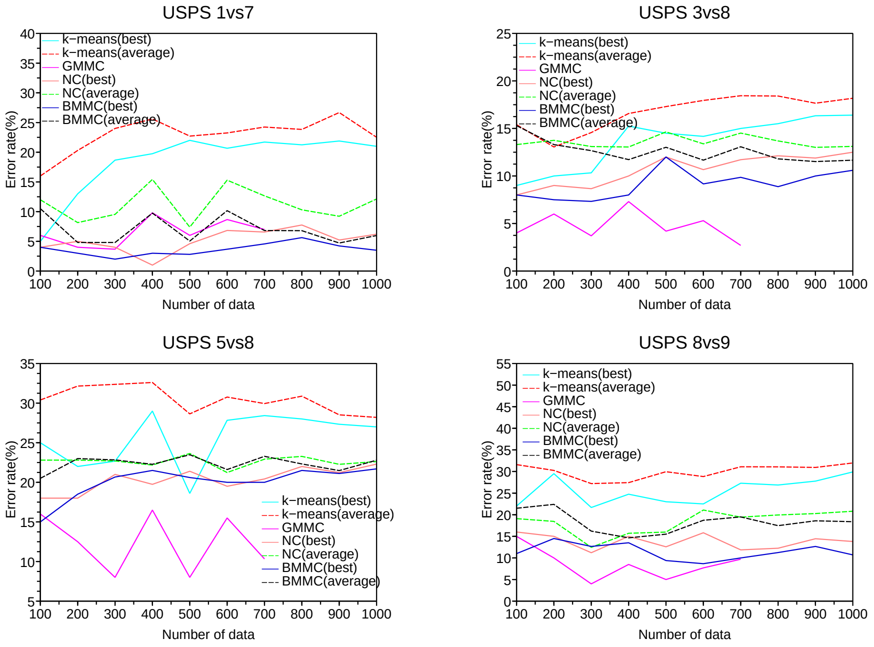


Figure 5. Empirical results on USPS with the number of instances increasing.

Table I
ERROR RATE (%) ON SMALL-SAMPLE CLUSTERING WITHOUT/WITH UNIVERSUM.

	NC best	NC average	BMMC best	BMMC average	BMMC $\mathcal{U}_{mean}$ beat	BMMC $\mathcal{U}_{mean}$ average	BMMC $\mathcal{U}_{gen}$ best	BMMC $\mathcal{U}_{gen}$ average
MNIST 5 vs. 8	10	21.5±6.93	10	20.2±5.85	8	18.2±4.76	12	17.6±4.50
MNIST 3 vs. 8	14	19.5±5.28	8	18.8±9.90	8	13±4.92	6	13.4±4.53
20Newsgroups 6 vs. 14	20	26.3±4.95	18	25±4.45	20	23.6±3.75	20	23.2±3.01
20Newsgroups 9 vs. 10	14	21.6±6.47	8	19±5.75	6	18.6±5.34	12	18.8±4.34

algorithm more flexible to different problems. Finally, empirical results show promising performance of the proposed model and verify the helpfulness of Universum in clustering, especially in the small-sample clustering problem.

For future work, a more efficient approximate inference method and the second multi-class clustering extension method mentioned in Section 3 are needed to be investigated. Moreover, the strategy for choosing the appropriate Universa still needs more research.

ACKNOWLEDGMENT

This work is supported by National Science Foundation of China (NSFC #60275025). We thank the reviewers for many helpful suggestions to improve this paper.

REFERENCES

[1] R. Redner and H. Walker, “Mixture densities, maximum likelihood and the em algorithm,” *SIAM Review*, vol. 26, no. 2, pp. 195–239, 1984.

[2] A. Ng, M. Jordan, and Y. Weiss, “On spectral clustering: Analysis and an algorithm,” in *NIPS*, 2001.

[3] U. von Luxburg, “A tutorial on spectral clustering,” *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[4] L. Xu, J. Neufeld, B. Larson, and D. Schuurmans, “Maximum margin clustering,” in *NIPS*, 2004.

[5] H. Valizadegan and R. Jin, “Generalized maximum margin clustering and unsupervised kernel learning,” in *NIPS*, 2006.

[6] K. Zhang, I. Tsang, and J. Kwok, “Maximum margin clustering made practical,” in *ICML*, 2007.

[7] B. Zhao, F. Wang, and C. Zhang, “Efficient maximum margin clustering via cutting plane algorithm,” in *SDM*, 2008.

[8] J. Shi and J. Malik, “Normalized cuts and image segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, 2000.

[9] S. Ben-David, T. Lu, D. Pál, and M. Sotáková, “Learning low-density separators,” in *AISTATS’09*, 2009.

[10] S. Ben-David and U. von Luxburg, “Relating clustering stability to properties of cluster boundaries,” in *COLT*, 2008.

[11] P. Sollich, “Probabilistic methods for support vector machines,” in *NIPS*, 2000.

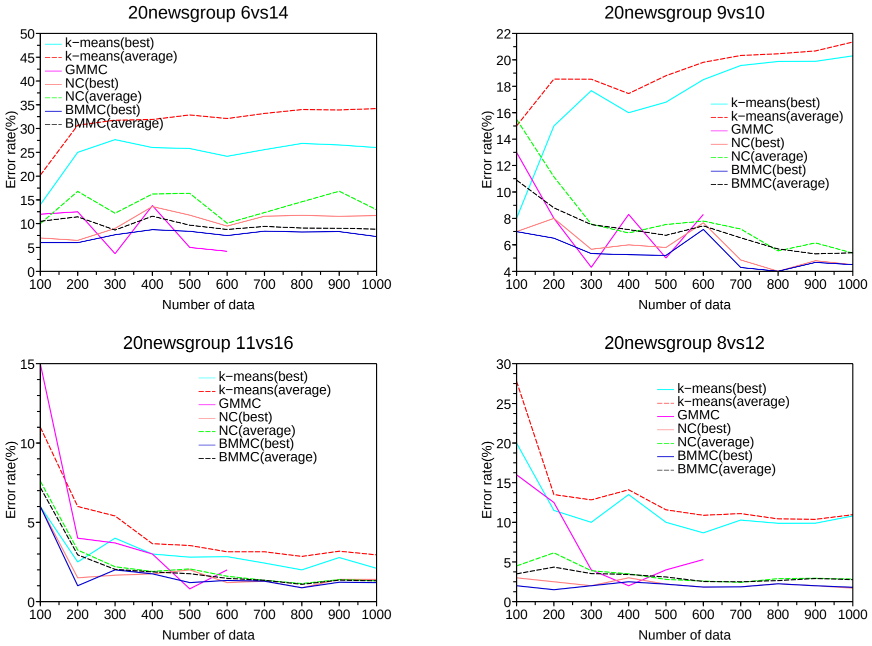
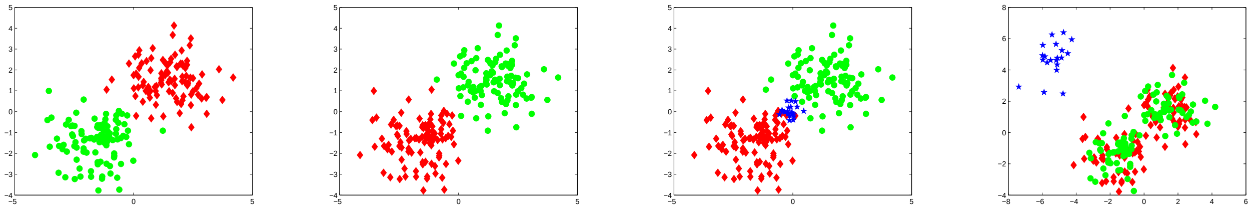


Figure 6. Empirical results on `20Newsgroups` with the number of instances increasing.



(a) (b) (c) (d)

Figure 7. Illustration of the effect of Universum on synthetic data set. The blue points are the Universa.

[12] N. Lawrence and M. Jordan, "Semi-supervised learning via gaussian processes," in *NIPS*, 2004.

[13] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[14] J. Weston, R. Collobert, F. Sinz, L. Bottou, and V. Vapnik, "Inference with the universum," in *ICML*, 2006.

[15] V. Vapnik, *Statistical Learning Theory*. Wiley-Interscience, 1998.

[16] ——, *Estimation of Dependences Based on Empirical Data: Springer Series in Statistics (2nd edition)*. Springer-Verlag Berlin, Inc., 2006.

[17] K. Huang, Z. Xu, I. King, and M. R. Lyu, "Semi-supervised learning from general unlabeled data," in *ICDM*, 2008.

[18] D. Zhang, J. Wang, F. Wang, and C. Zhang, "Semi-supervised classification with universum," in *SDM*, 2008.

[19] Y. Li, I. Tsang, J. Kwok, and Z. Zhou, "Tighter and convex maximum margin clustering," in *AISTATS*, 2009.

[20] O. Chapelle, B. Schölkopf, and A. Zien, Eds., *Semi-Supervised Learning*. Cambridge, MA: MIT Press, 2006.

[21] J. Martens, "Deep learning via hessian-free optimization," in *ICML*, 2010.

[22] S. Rogers and M. Girolami, "Multi-class semi-supervised learning with the ϵ-truncated multinomial probit gaussian process," *JMLR Workshop and Conference Proceedings 1*, pp. 17–32, 2007.

[23] F. Sinz, O. Chapelle, A. Agarwal, and B. Schölkopf, "An analysis of inference with the universum," in *NIPS*, 2008.

[24] I. Guyon, U. von Luxburg, and R. Williamson, "Clustering: Science or art?" in *NIPS 2009 Workshop on Clustering Theory*, 2009.

[25] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 1.21," http://cvxr.com/cvx, May 2010.

[26] U. von Luxburg and S. Ben-david, "Towards a statistical theory of clustering," in *In PASCAL workshop on Statistics and Optimization of Clustering*, 2005.

[27] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, July 2006.