Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)

Maximum Margin Dirichlet Process Mixtures for Clustering

Gang Chen[1], **Haiying Zhang**[2*] and **Caiming Xiong**[3]
[1]Computer Science and Engineering, SUNY at Buffalo, Buffalo, NY 14260, gangchen@buffalo.edu
[2]State Key Laboratory of Remote Sensing Science, RADI, Chinese Academy of Sciences, Beijing 100101
[3]MetaMind Inc., 172 University Avenue, Palo Alto, CA 94301, cmxiong@metamind.io

Abstract

The Dirichlet process mixtures (DPM) can automatically infer the model complexity from data. Hence it has attracted significant attention recently, and is widely used for model selection and clustering. As a generative model, it generally requires prior base distribution to learn component parameters by maximizing posterior probability. In contrast, discriminative classifiers model the conditional probability directly, and have yielded better results than generative classifiers. In this paper, we propose a maximum margin Dirichlet process mixture for clustering, which is different from the traditional DPM for parameter modeling. Our model takes a discriminative clustering approach, by maximizing a conditional likelihood to estimate parameters. In particular, we take a EM-like algorithm by leveraging Gibbs sampling algorithm for inference, which in turn can be perfectly embedded in the online maximum margin learning procedure to update model parameters. We test our model and show comparative results over the traditional DPM and other nonparametric clustering approaches.

Introduction

Bayesian nonparametric models (Antoniak 1974; Sethuraman and Tiwari 1981; Rasmussen 2000; Nguyen et al. 2014) have received a lot of attention in the machine learning community. The attractive property of these models is that the number of components can be learned automatically from data, without being specified in advance. One of the most popular nonparametric models for clustering is Dirichlet process mixture model (DPM) (Neal 2000; Teh 2010), which has been widely used on character recognition and document categorization (Blei and Jordan 2005; Kurihara, Welling, and Teh 2007). For DPM, the prior imposed by Dirichlet process (DP) is defined by two parameters: the concentration parameter α and the base measure G_0 respectively. In a DP mixture model, both two parameters heavily influence the model selection and clustering performance. In general, the concentration parameter can be learned from the data adaptively (Gilks and Wild 1992) and the component parameters can be estimated by maximizing posterior probability (Rasmussen 2000). However, it remains challenging to estimate these parameters due to the appearance of intractable normalizing constants in the likelihood. One possible trend is to either assume conjugate priors or employ approximate methods (Blei and Jordan 2005) to accelerate the inference. Another trend that catches great attention recently is to learn the parameters with the discriminative models (Vapnik 1995). An important question is whether it is possible to learn a discriminative model for nonparametric clustering; and whether the modeling performance is weakened without conjugate prior assumption.

In this work, we learn component parameters in a discriminative manner, instead of the generative method used in the DPM model. We prefer discriminative models because: (1) we can circumvent the intractable inference effectively, that is usually a challenge in DPM; (2) discriminative models can greatly reduce the parameter space. For example, the DPM with Gaussian mixtures need to store and update covariance matrix, which is very expensive in high dimension space; (3) it has been demonstrated that discriminative models generally yield higher accuracy than generative approaches (Nigam, Lafferty, and McCallum 1999; Jebara and Pentland 1998; Lafferty, McCallum, and Pereira 2001). Furthermore, maximum margin learning demonstrates promising results on both classification (Vapnik 1995; Tsochantaridis et al. 2005) and clustering problems (Xu et al. 2005; Chen, Zhu, and Zhang 2014). Recent advances in online maximum margin learning (Crammer et al. 2006) make it possible to embed Gibbs sampling inference into the discriminative model.

Hence, we propose a maximum margin Dirichlet process mixture model (MMDPM), which inherits the advantages of online maximum margin learning and nonparametric clustering. As a discriminative model, we directly optimize the conditional model to learn component parameters. More specifically, we use Gibbs sampling to infer each instance's label, and in turn we use it to learn model parameters and do the model selection in an online fashion. Our contributions can be summed up as follows:

- optimize a conditional model, instead of the joint likelihood as in DPM model;
- learn model parameters online, via Gibbs sampling and maximum margin learning in an unified framework;
- yield higher clustering accuracy with fast speed.

We test our model on both synthetic and real datasets, and

*Corresponding author: zhanghy@radi.ac.cn

show comparative results over DPM and other nonparametric clustering methods.

Maximum margin DPM

Our maximum margin DPM (MMDPM) extends the generative Dirichlet process mixture model (Ferguson 1973) by maximizing the conditional likelihood, and learns these parameters online. We start with Gaussian mixture model formulation and then take the limit as the number of mixture components approaches infinity to obtain DPM. Then, we introduce our model, with EM-like algorithm for parameter estimation via alternating Gibbs sampling and online maximum margin learning. Throughout the paper, vector quantities are written in bold. The index i always indicates observations, $i = \{1, ..., n\}$, index k runs over components, $k = \{1, .., K\}$, and index t indicates iterations. Generally, variables that play no role in conditional distributions are dropped from the condition for simplicity.

Dirichlet process mixture model

The Dirichlet process (DP) (Ferguson 1973) is parameterized by a base distribution G_0 and a positive scaling parameter α. A DPM model can be constructed as a limit of a parametric mixture model (Neal 2000; Blei and Jordan 2005) with the DP prior. For instance, we can generate $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^n$ with symmetric Dirichlet prior:

$$\begin{array}{l} \boldsymbol{\pi}|\alpha \sim Dir(\alpha/K, ..., \alpha/K) \\ z_i|\boldsymbol{\pi} \sim Discrete(\pi_1, ..., \pi_K) \\ \boldsymbol{\theta}_k \sim G_0(\beta) \\ \mathbf{x}_i|z_i, \{\boldsymbol{\theta}_k\}_{k=1}^K \sim p(\mathbf{x}_i|\boldsymbol{\theta}_{z_i}) \end{array} \tag{1}$$

In this model, each datum $\mathbf{x}_i$ is generated by sampling one of K clusters firstly, say, cluster k, according to the multinomial distribution that is parameterized by $\boldsymbol{\pi}$, and then sampling from the distribution of this cluster $p(\mathbf{x}_i|\boldsymbol{\theta}_{z_i})$ that is parameterized by $\boldsymbol{\theta}_k$. In this equation, an indicator variable $z_i \in \{1, ..., K\}$ are stochastic variables which encodes the class (or mixture component) to which observation $\mathbf{x}_i$ belongs. The mixture weight $\boldsymbol{\pi}$ is given a symmetric Dirichlet prior with a hyperparameter α and the cluster parameters $\{\boldsymbol{\theta}_k\}_{k=1}^K$ are given with a common prior distribution $G_0(\beta)$ with parameter β.

Fixing all but a single indicator z_i, we can obtain the conditional probability for each individual indicator

$$p(z_i = k|\mathbf{z}_{-i}, \alpha) = \int_{\boldsymbol{\pi}} p(z_i|\boldsymbol{\pi}) p(\boldsymbol{\pi}|\alpha) = \frac{n_{-i,k} + \alpha/K}{n - 1 + \alpha} \tag{2}$$

where the subscript $-i$ indicates all indices except for i, and $n_{-i,k}$ is the number of data points, excluding $\mathbf{x}_i$, that are associated with class k. Let K go to infinity, the conditional distribution of the indicator variables reaches the following limits (Görür and Rasmussen 2010):

$$p(z_i|\mathbf{z}_{-i}, \alpha) = \begin{cases} p(z_i = k|\mathbf{z}_{-i}, \alpha) = \dfrac{n_{-i,k}}{n - 1 + \alpha} \\ p(z_i \neq z_{i'} \text{ for all } i \neq i'|\mathbf{z}_{-i}, \alpha) = \dfrac{\alpha}{n - 1 + \alpha} \end{cases} \tag{3}$$

where i' in the right hand side of Eq. (3) is the set of existed cluster indicators. In other words, the prior for assigning instance $\mathbf{x}_i$ to either an existing component k or to a new one cluster conditioned on the other component assignments (z_i) is given by Chinese restaurant process (Blei and Jordan 2005). For DPM, we need to specify the base distribution G_0 to complete the model. Note that G_0 specifies the prior on the component parameters $\{\boldsymbol{\theta}_k\}_{k=1}^K$.

In the following part, we introduce our discriminative model with Gibbs sampling for inference and maximum margin learning for parameter estimation.

Gibbs sampling

Given the data points $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^n$ ($\mathbf{x}_i \in \mathbb{R}^d$) and their cluster indicators $\mathcal{Z} = \{z_i\}_{i=1}^n$, the Gibbs sampling involves iterations that alternately draws from conditional probability while keeping other variables fixed. Recall that for each indicator variable z_i, we can derive its conditional posterior in DPM as follows:

$$\begin{aligned} & p(z_i = k|\mathbf{z}_{-\mathbf{i}}, \mathbf{x}_i, \{\boldsymbol{\theta}_k\}_{k=1}^K, \alpha, \beta) && (4) \\ = \; & p(z_i = k|\mathbf{x}_i, \mathbf{z}_{-\mathbf{i}}, \{\boldsymbol{\theta}_k\}_{k=1}^K) && (5) \\ \propto \; & p(z_i = k|\mathbf{z}_{-\mathbf{i}}, \{\boldsymbol{\theta}_k\}_{k=1}^K) p(\mathbf{x}_i|z_i = k, \{\boldsymbol{\theta}_k\}_{k=1}^K) && (6) \\ = \; & p(z_i = k|\mathbf{z}_{-\mathbf{i}}, \alpha) p(\mathbf{x}_i|\boldsymbol{\theta}_k) && (7) \end{aligned}$$

where $p(z_i = k|\mathbf{z}_{-\mathbf{i}}, \alpha)$ is determined by Eq. (3), and $p(\mathbf{x}_i|\boldsymbol{\theta}_k)$ is the likelihood for the current observation $\mathbf{x}_i$. To estimate $\boldsymbol{\theta}_k$, we need to maximize the conditional posterior, which depends on observations belonging to this cluster and prior $G_0(\beta)$. If the current set for the cluster k, can be denoted as $\mathbf{x}_k$, with the number of elements $n_k = |\mathbf{x}_k|$, then DPM learns $\boldsymbol{\theta}_k$ by maximizing the posterior:

$$p(\boldsymbol{\theta}_k|\mathbf{x}_k, \beta) = p(\boldsymbol{\theta}_k|\beta) \prod_{i=1}^{|\mathbf{x_k}|} p(\mathbf{x}_{k_i}|\boldsymbol{\theta}_k) \tag{8}$$

For this model, a conjugate base distribution may exist, which can provide guarantee that the posterior probability can be computed in closed form and learn the model parameters explicitly. However, in general it is hard to choose an appropriate prior base distribution, i.e., often chosen based on mathematical and convenient concern. Moreover, it has the unappealing property of prior dependency, and cannot reflect the observed data distribution in real scenarios, i.e., the observations obey a certain shape in Fig. 2.

In our conditional likelihood model, we replace the generative model in DPM with our discriminative SVM classifier. More specifically, we relax the prior restriction $G_0(\beta)$ and learn the component parameters $\{\boldsymbol{\theta}_k\}_{k=1}^K$ in a discriminative manner. Thus, we define the following likelihood for instance $\mathbf{x}_i$ in Eq. (7):

$$p(\mathbf{x}_i|\boldsymbol{\theta}_k) \propto exp(\mathbf{x}_i^T \boldsymbol{\theta}_k - \lambda||\boldsymbol{\theta}_k||^2) \tag{9}$$

where λ is a regularization constant to control weights between the two terms above. By default, the prediction function should be proportional to $\text{argmax}_k(\mathbf{x}_i^T \boldsymbol{\theta}_k)$, for $k \in \{1, ..., K\}$. In our likelihood definition, we also minus $\lambda||\boldsymbol{\theta}_k||^2$ in Eq. (9), which can keep the maximum margin

beneficial properties in the model to separate clusters as far away as possible. Moreover, it can get rid of trivial clustering results (Hoai and Zisserman 2013). Note that the seminal work (Platt 1999) basically fits a sigmoid function over SVM decision values (e.g. $\mathbf{x}_i^T\boldsymbol{\theta}_k$) to scale it to the range of $[0, 1]$, which can then be interpreted as a kind of probability. Compared to (Platt 1999), we maximize $(\mathbf{x}_i^T\boldsymbol{\theta}_k)$ and minimize $\lambda||\boldsymbol{\theta}_k||^2$ simultaneously in Eq. (9), so our method can keep larger margin between clusters. Another understanding for the above likelihood is that Eq. (9) satisfies the general form of exponential families, which are functions solely of the chosen sufficient statistics (Sudderth 2006). Thus, such probability in Eq. (9) makes our model general enough to handle real applications.

Taking the similar form as in Eq. (7), we get the final Gibbs sampling strategy for our MMDPM model:

$$\begin{aligned} &p(z_i = k|\mathbf{z}_{-\mathbf{i}}, \mathbf{x}_i, \{\boldsymbol{\theta}_k\}_{k=1}^K, \alpha, \lambda) \\ &\propto p(z_i = k|\mathbf{z}_{-\mathbf{i}}, \alpha) exp(\mathbf{x}_i^T\boldsymbol{\theta}_k - \lambda||\boldsymbol{\theta}_k||^2) \end{aligned} \tag{10}$$

we will introduce to learn component parameters $\{\boldsymbol{\theta}_k\}_{k=1}^K$ in the following Sec . For the new created cluster, we generate $\boldsymbol{\theta}_{K+1}$ that is perpendicular to all the previous $\boldsymbol{\theta}_k$, for $k \in \{1, ..., K\}$ (Xiao, Zhou, and Wu 2011; Hoai and Zisserman 2013). Basically, we random generate a vector $\boldsymbol{\theta} \in \mathbb{R}^d$, and then we project it on the weight vector $\boldsymbol{\theta}_k$, $k \in \{1, .., K\}$, and compute the residual vectors:

$$\boldsymbol{\theta}_{K+1} := \boldsymbol{\theta} - (\boldsymbol{\theta}_k^T\boldsymbol{\theta})\boldsymbol{\theta}_k \tag{11}$$

The residual is the component of $\boldsymbol{\theta}_{K+1}$ that is perpendicular to $\boldsymbol{\theta}_k$, for $k \in \{1, ..., K\}$.

For the model we consider, we leverage MCMC algorithms for inference on the model discussed above by sampling each variable from posterior conditional probability given in Eq. (10) with others fixed in an alternative way. In addition, we update α using Adaptive Rejection Sampling (ARS) (Gilks and Wild 1992) as suggested in (Rasmussen 2000).

Maximum margin learning

Given the clustering label for each instance, we can use K-means to estimate the component parameters. Unfortunately, K-means cannot keep an larger margin properties between clusters. In our work, we estimate the component parameters under the maximum margin framework. More specifically, we use the variant of the passive aggressive algorithm (PA) (Crammer et al. 2006) to learn component parameters. Basically, our online algorithm treats the labeling inference with Gibbs sampling as groundtruth. Then, for any instance in a sequential manner, it infers an outcome with the current model. If the prediction mismatches its feedback, then the online algorithm update its model under the maximum margin framework, presumably improving the chances of making an accurate prediction on subsequent rounds.

We denote the instance presented to the algorithm on round t by $\mathbf{x}_t \in \mathbb{R}^d$, which is associated with a unique label $z_t \in \{1, ..., K\}$. Note that the label z_t is determined by the above Gibbs sampling algorithm in Eq. (10). Let's define $\mathbf{w} = [\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K]$ is a parameter vector with K clusters (by concatenating all the parameters $\{\boldsymbol{\theta}_k\}_{k=1}^K$ into $\mathbf{w}$, that means $\mathbf{w}^{z_t}$ is z_t-th block in $\mathbf{w}$, or says $\mathbf{w}^{z_t} = \boldsymbol{\theta}_{z_t}$), and $\boldsymbol{\Phi}(\mathbf{x}_t, z_t)$ is a feature vector relating input $\mathbf{x}_t$ and output z_t, which is composed of K blocks, and all blocks but the z_t-th blocks of are set to be the zero vector while the z_t-th block is set to be $\mathbf{x}_t$. We denote by $\mathbf{w_t}$ the weight vector used by the algorithm on round t, and refer to the term $\gamma(\mathbf{w}_t; (\mathbf{x}_t, z_t)) = \mathbf{w}_t \cdot \boldsymbol{\Phi}(\mathbf{x}_t, z_t) - \mathbf{w}_t \cdot \boldsymbol{\Phi}(\mathbf{x}_t, \hat{z}_t)$ as the (signed) margin attained on round t, where $\hat{z}_t = \max_{z \in [1,K]} \mathbf{w}_t \cdot \boldsymbol{\Phi}(\mathbf{x}_t, z)$. In our work, we use hinge-loss function, which is defined by the following,

$$\begin{aligned} &\ell(\mathbf{w}; (\mathbf{x}_t, z_t)) \\ &= \begin{cases} 0 & \text{if } \gamma(\mathbf{w}; (\mathbf{x}_t, z_t)) \geq 1 \\ 1 - \gamma(\mathbf{w}; (\mathbf{x}_t, z_t)) & \text{otherwise} \end{cases} \end{aligned} \tag{12}$$

Following the passive aggressive (PA) algorithm (Crammer et al. 2006), we optimize the objective function:

$$\begin{aligned} \mathbf{w}_{t+1} &= \arg\min_{\mathbf{w}} \frac{1}{2}||\mathbf{w} - \mathbf{w}_t||^2 + C\xi \\ &s.t.\ \ell(\mathbf{w}; (\mathbf{x}_t, z_t)) \leq \xi \end{aligned} \tag{13}$$

where the L_2 norm of $\mathbf{w}$ on the right hand size can be thought as Gaussian prior in SVM classifier. If there's loss, then the updates of PA-1 has the following closed form

$$\begin{aligned} \mathbf{w}_{t+1}^{z_t} &= \mathbf{w}_t^{z_t} + \tau_t \mathbf{x}_t, \\ \mathbf{w}_{t+1}^{\hat{z}_t} &= \mathbf{w}_t^{\hat{z}_t} - \tau_t \mathbf{x}_t, \end{aligned} \tag{14}$$

where $\tau_t = \min\{C, \frac{\ell(\mathbf{w}_t; (\mathbf{x}_t, z_t))}{||\mathbf{x}_t||^2}\}$. Note that the Gibbs sampling step can decide the indicator variable z_t for $\mathbf{x}_t$, we think it is the ground truth assignment for $\mathbf{x}_t$, and then we update our parameter $\mathbf{w}$ using the above Eq. (14).

Parameter space analysis: if the data dimension is d, and the current cluster number is K, then $\mathbf{w}$ need $d \times K$ in our model. While for the DPM model, if we assume a Gaussian distribution, we need to update and store $d^2 \times K$ for covariance matrix, and that is computationally expensive for high dimensional data. Even for the diagonal covariance matrix, it still requires $2d \times K$ to store both mean and covariance.

Time complexity analysis: In the algorithm, we do both inference and learning, so it needs $O(n \times d^2 \times K)$ in each iteration. Note that K is changing in each iteration. While for the DPM model, the component parameters updating requires Cholesky decomposition in most cases. Thus, our online updating on the component parameters is more efficient than the DPM model.

Model coherence: Given component parameters $\{\boldsymbol{\theta}_k\}_{k=1}^K$ estimated from maximum margin learning, we can predict the likelihood for each $\mathbf{x}_t$ as $p(\mathbf{z}_i|\mathbf{x}_t; \boldsymbol{\theta}_k) \propto exp(\mathbf{x}_i^T\boldsymbol{\theta}_k)$. If we want to keep a large margin between different clusters, then we can introduce a prior $\{\boldsymbol{\theta}_k\}_{k=1}^K$ and finally get the posterior probability in Eq. (9). Note that Eq. (9) is close to the probabilistic formula of K-means.

Algorithm

We list the pseudo code below in Algorithm 1 and implemented it in Matlab. Our method takes an EM-like algorithm to estimate model parameters: infer the label of the current instance with Gibbs sampling; and update model parameters given the label for that instance.

Algorithm 1 Maximum margin Dirichlet process model

Input: sequential training data $\mathcal{X}$, C, λ, iterations T
Output: $\mathbf{w}$, and $\mathcal{Z}$

Initialize $\mathbf{w_1}$, labels $\mathcal{Z}$ for training data, and prior α;
for $i = 1; i < n; i++$ **do**
  $\mathbf{x}_i = \mathbf{x}_i - mean(\mathcal{X})$;
end for
for $t = 1$ to T **do**
  Permutate $\mathcal{X}$;
  for $i = 1; i < n; i++$ **do**
    Select an instance $(\mathbf{x}_i, z_i)$, and update $n_{z_i} = n_{z_i} - 1$;
    Eliminate empty clusters (if there's empty cluster, then $K = K - 1$);
    Create a new cluster $\boldsymbol{\theta}_{K+1}$ using Eq. (11), and update the number of clusters $K = K + 1$;
    for $j = 1; j <= K; j++$ **do**
      Calculate posterior probability in Eq. (10) for each cluster;
    end for
    Sample its assignment according to the posterior probability in Eq. (10);
    Update its assignment $\hat{z}_i$ and $n_{\hat{z}_i} = n_{\hat{z}_i} + 1$;
    Update $\mathbf{w}_t$ using maximum margin clustering algorithm in Eq. (14);
  end for
  Update α with ARS algorithm
end for
Return $\mathbf{w}$ and $\mathcal{Z}$;

Experiments

In this section, we conduct empirical studies on both synthetic and real datasets to evaluate the performance of our method. We also compare the computational cost between our model and baselines when we vary the number of data samples and dimensionality.

Experiment setup: For both DPM and MMDPM, we approximated the infinite vector of the mixing proportions using a finite symmetric Dirichlet prior. For DPM, we assume the distributions generating the instances of each component were Gaussians (mean and precision matrix), and assume β obey normal-inverse Wishart prior for its mean and precision. In our MMDPM setting, we initialize $\lambda = 3$ in the conditional model in Eq. (9) if it is not specified, and $C = 0.01$ in the passive aggressive updating algorithm in Eq. (14). In general, a larger λ leads to a larger number of clusters. As for the number of iterations, we set $T = 100$. The initial number of components was set to 1 and the concentration parameter α was set to 4 in all experiments. We implemented our algorithm with Matlab, and all experiments were conducted on Intel(R) Core(TM) i7-3770K CPU running at 3.50GHz with 32 GB of RAM.

Evaluation measure: The evaluation of unsupervised clustering against a gold standard is not straightforward because the clusters found by the algorithm are not associated with the classes in the gold standard. In our experiments, we use the widely used F-measure (Achtert et al. 2012), V-measure (Rosenberg and Hirschberg 2007) and adjusted Rand Index (Hubert and Arabie 1985; Rand 1971) to evaluate the clustering results.

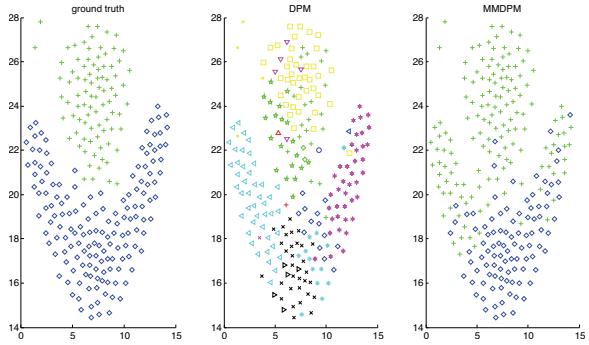


Figure 1: The experimental comparison on the Frame dataset. Different colors and shapes indicate different clusters. The left is the ground truth. The middle is the result (F-score: 0.43) using DPM clustering; the right is our method's result (F-score: 0.60). It demonstrates that our method can get better clustering performance here.

Dataset: The synthetic datasets are composed of 3 toy datasets (available on line[1]): Jain's toy dataset (Jain 2007), Aggregation (Gionis, Mannila, and Tsaparas 2007) and Frame dataset (Fu and Medico 2007). For the real datasets, we test our method on Iris, Wine, Glass and Wdbc datasets, which are available from the UCI Machine Learning Data Repository[2]. We also test our method on MNIST digits[3], 20 newsgroup dataset[4] and the Reuters data set.

Experimental results: We compare our method to the standard DPM. For the dataset with Gaussian distribution, DPM can get better results. For example, DPM can get very good performance on Aggregation dataset in Table 1. While for other datasets with no Gaussian distribution, our method outperforms DPM. We also show the clustering results on these 2-dimension toy datasets for visual understanding. The Frame dataset is non-linear separable, see the left image in Fig. 1. The DPM can cluster the points well in an local view (i.e., it divides the toy data into 12 compact clusters), see the middle in Fig. 1, but it cannot separate these two semantic clusters well. Jain's toy dataset has 2 clusters with spiral shape, see the left image in Fig. 2. Similarly, DPM can partition the points well in an local view into 6 compact clusters, but it cannot separate these two semantic clusters well. For both cases, our method can get better performance here even for these non linear separable toy datasets.

We also compare our method to the baseline on the real UCI datasets. We choose four widely used datasets from UCI repository: Iris, Glass, Wdbc and Wine. Note that these datasets have different dimensions and different number of clusters, which are very good to test model selection performance for our method. The experimental results in Table 1 demonstrate that our method can get better performance on all four datasets with F-measure, and outperform the baseline on three of the four datasets (Wdbc, Iris and Wine) with

[1] http://cs.joensuu.fi/sipu/datasets/
[2] http://www.ics.uci.edu/~mlearn/MLRepository.html
[3] http://yann.lecun.com/exdb/mnist/
[4] http://people.csail.mit.edu/jrennie/20Newsgroups

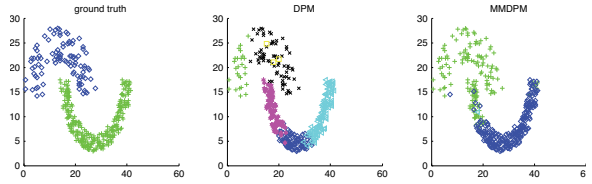


Figure 2: The experimental comparison on Jain' toy dataset. The left is the ground truth; the middle is the result (F-score: 0.590) of DPM; the right is our method's result (F-score: 0.73). Considering it is non-linear separable, our method can get better performance here.

Dataset	DPM		MMDPM	
	F-score	V-score	F-score	V-score
Jain	0.59	0.46	0.73	0.41
Aggregation	0.91	0.90	0.79	0.75
Frame	0.43	0.41	0.60	0.29
Wdbc	0.43	0.26	0.85	0.57
Glass	0.49	0.43	0.51	0.38
Iris	0.71	0.66	0.75	0.68
Wine	0.36	0.42	0.68	0.48
MNIST	0.18	0.06	0.368	0.389

Table 1: The experimental comparison on synthetical datasets (the first three rows), UCI datasets (the middle four rows) and MNIST digits. Our method outperforms DPM on Jain and Frame datasets, except on Aggregation dataset. For the real UCI datasets, Our method outperform DPM on all the four datasets, except V-score on Glass dataset. It demonstrates that our method is significantly better than DPM.

V-measure. Refer to Table 1 for more details.

Character clustering MNIST digits consists of 28×28-size images of hand-written digits from '0' through '9'. We preprocess the images with PCA, reducing them into 100 dimensions, so as to retain about 90% of the total variance. Then, we randomly sample 2000 examples from 60000 training images for clustering analysis. The quantitative comparison is shown in Table 1, which demonstrates that our method yields better clustering results. We also analyze the time complexity. The result in Fig. 3(a) shows that our online method is almost linear in the number of training samples, compared to variational Bayesian DPM (Blei and Jordan 2005).

Newsgroup categorization The 20 Newsgroups dataset is a collection of approximately 18,846 newsgroup documents, which are divided into nearly across 20 different topics. The 20 Newsgroups dataset has total 61188 vocabularies, and each document is represented as a histogram by words counting. In all the following experiments, we keep the same parameters, except setting $\lambda = 0.2$.

Firstly, we selected the most frequent 250 words as the codebook for feature representation. And then we sampled 10000 examples, and projected them into 250 dimensions. We did clustering analysis on the projected examples (repeat 10 times to calculate the average), and compared it to widely used baselines, including K-means, Gaussian mixture model (GMM), Spectral clustering and DPM. For GMM and spectral clustering, we set the number of clusters equal to 20 (ground truth), and their results in a sense can be thought as the upper bound. The results in Table 2 shows our method outperforms baselines with F-measure. As for V-measure, our model also yield comparative result, which is better than spectral clustering and DPM.

In order to test how the accuracy changes with data dimensionality, we rank vocabularies according to their frequency, and vary the codebook size by selecting the most frequent words to encode each document into histogram. The time complexity comparison between our method and variational DPM is shown in Fig. 3(b). The accuracy comparison results between variational Bayes DPM and our method is shown in Fig. 3(c) and (d). It demonstrates that our method yield better results compared to DPM.

We take another experiment on the Reuters data set. In the experiment, we used the Reuters21578[5], which has the total 8293 documents with 18933 dimensional features for each document, belonging to 65 categories. Because the Reuters dataset has high dimension, we first projected it into 100 dimensions with PCA to keep 95% of the total variance. Then we normalize the data and do the clustering analysis on the projected data. Note that the clustering results of other baselines is evaluated on the same PCA projected data. As for the parameter setting, we keep the same parameters (e.g. $\alpha = 4$), except setting $\lambda = 80$. The clustering performance is shown in Table 3 and demonstrates that our method is significantly better than other methods on data clustering task.

Discussion

Recall that the DPM model maximizes the joint model

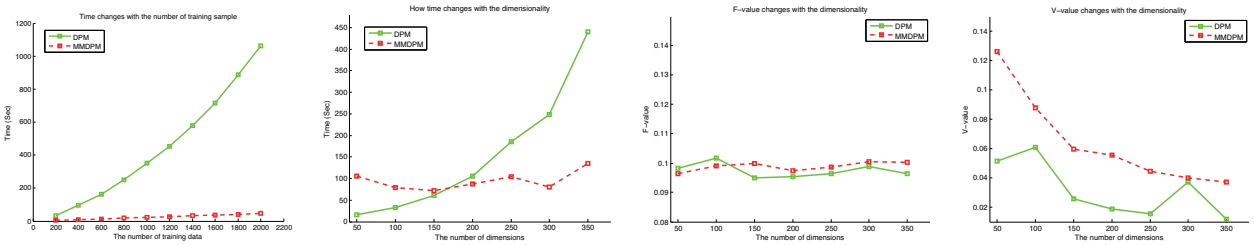
$$P(\mathcal{X}, \mathbf{z}, \{\boldsymbol{\theta}_k\}_{k=1}^K) \propto p(\mathbf{z}) \prod_{i=1}^N p(\mathbf{x}_i|\boldsymbol{\theta}_{z_i}) \prod_{k=1}^K p(\boldsymbol{\theta}_k|\beta) \quad (15)$$

where $p(\boldsymbol{\theta}_k|\beta)$ is defined by the base measure $G_0(\beta)$, and $p(\mathbf{z}) = \frac{\Gamma(\alpha)\prod_{k=1}^K \Gamma(n_k+\alpha/K)}{\Gamma(n+\alpha)\Gamma(\alpha/K)^K}$ for the symmetric Dirichlet prior. As for our discriminative model, we maximize the following conditional likelihood:

$$P(\mathbf{z}, \{\boldsymbol{\theta}_k\}_{k=1}^K|\mathcal{X}) \propto p(\mathbf{z}) \left[\prod_{i=1}^N p(\mathbf{x}_i|\boldsymbol{\theta}_{z_i})\right] \prod_{k=1}^K p(\boldsymbol{\theta}_k) \quad (16)$$

where $p(\mathbf{z})$ has the same definition as in DPM above, while we have no prior base measure restriction on $\{\boldsymbol{\theta}_k\}_{k=1}^K$ in our discriminative model. $p(\boldsymbol{\theta}_k)$ for $k = \{1, ..., K\}$ can be thought as the Gaussian prior in SVM classifier in Eq. (13). The essential difference between our model and DPM is that our approach is a discriminative model, without modeling $p(\mathcal{X})$. And, we maximize a conditional probability for parameter estimation, instead of joint distribution as in DPM. Just as the conditional random fields model (CRF) (Lafferty, McCallum, and Pereira 2001), we propose a similar discriminative model, which do not need to tune prior assumption

[5] http://www.cad.zju.edu.cn/home/dengcai/Data/TextData.html



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

Figure 3: How complexity and accuracy change with the number of training data and dimensionality. (a) The time comparison between variational bayes DPM and our method on a subset MNIST dataset with 2000 training examples. (b) Given the 2000 training samples from News 20 group, it shows how the time complexity changes with data dimensionality. (c) and (d) show how F-score and V-score change with data dimensionality on 20 news dataset.

Measures	Average accuracy					
	Methods					
	K-means	GMM	Spectral	DPM	DPVC (Knowles, Palla, and Ghahramani 2012)	MMDPM
F-score	0.088	0.088	0.095	0.094	0.09	**0.10**
V-score	0.100	**0.104**	0.061	0.049	0.02	0.066

Table 2: The experimental comparison on a subset of News 20 dataset, with total 10000 training examples and 250 codebooks. We compare the performances between our method and others baselines. It demonstrates that our method outperforms DPM significantly.

Measures	Average accuracy					
	Methods					
	K-means	GMM	Spectral	DPM	DPVC(Knowles, Palla, and Ghahramani 2012)	MMDPM
F-score	0.146	0.173	0.09	0.484	0.32	**0.507**
V-score	0.457	0.464	0.432	**0.472**	0.395	0.335
Adjusted Rand Index	0.100	0.123	0.062	0.383	0.211	**0.416**

Table 3: The experimental comparison on the Reuters dataset. We compare the performances between our method and others baselines. It demonstrates that our method outperforms DPM and DPVC significantly with adjusted rand index and v-measure.

for $\{\boldsymbol{\theta}_k\}_{k=1}^K$ constricted by $G_0(\beta)$ in DPM. Removing constraints reduces the statistical bias, and fit the model parameters well to the training data. In addition, the difficulty in modeling Eq. (15) is that it often contains many highly dependent features, which are difficult to model (Minka 2003; Sutton and Mccallum 2006).

Related work

Dirichlet process mixture model (DPM) adopts DP prior to determine model complexity and learn the mixture distributions of data automatically. Considering the advantages of DPM, it has been extensively used for model selection and clustering (Antoniak 1974; Sethuraman and Tiwari 1981; Neal 2000; Rasmussen 2000; Blei and Jordan 2005). In general, the conjugate prior assumption is preferred for mathematical and computational concern, otherwise it is not tractable to compute posterior probability.

One trend is to use DPM in different research fields. Vlachos et al. had applied DPM for verb clustering on natural language processing problem (Vlachos, Ghahramani, and Korhonen 2008). How to evaluate the influence of the base distribution to DPM is also explored in (Görür and Rasmussen 2010). Another direction is to speed up the inference in DPM. Markov chain Monte Carlo (MCMC) has been widely used for DPM inference, see (Neal 2000) for a survey of MCMC inference procedures for DP mixture models. Except MCMC, variational Bayesian approaches (Blei and Jordan 2005; Kurihara, Welling, and Teh 2007) are also proposed. In addition, many DPM variants have also been proposed recently. For example, (Shahbaba, Neal, and Ghahramani 2009) introduced a new nonlinear model for classification, which models the joint distribution of response variable and covariates, nonparametrically using DPM. Recently, (Hannah, Blei, and Powell 2011) proposed Dirichlet Process mixtures of Generalized Linear Models (DP-GLM), a new class of methods for nonparametric regression. (Zhang, Dai, and Jordan 2010) have derived a new Bayesian nonparametric kernel regression method based on the matrix-variate Dirichlet process mixture prior and introduced an MCMC algorithm for inference and prediction. Recently, Knowles et al. proposed a Dirichlet process variable clustering (DPVC) method by leveraging the correlation between variables and formulating the corresponding probabilistic model for non-parametric clustering (Knowles, Palla, and Ghahramani 2012). In general, typical

clustering algorithms, such as K-means and GMM (Chen et al. 2009), consider how similar entities (in terms of Euclidean distance) rather how correlated they are. Thus, DPVC can discover block diagonal covariance structures in data, and partition observed variables into sets of highly correlated variables for clustering.

On the other hand, much work has focused on maximum margin clustering and demonstrate promising results (Xu et al. 2005; Hoai and Zisserman 2013). However, these methods still need to specify the number of clusters.

Conclusion

In this paper, we propose a maximum margin Dirichlet process mixture model (MMDPM) for clustering. We infer indicator variables with Gibbs sampling, which can be perfectly embedded in our online maximum margin framework for parameters learning. In a sense, our conditional model can fit the dataset well and get better decision boundaries for clustering problem. Moreover, our model can greatly reduce the space storage and the learning time, compared to DPM. To the best of our knowledge, this is the first work to learn the model parameters for nonparametric clustering in an discriminative manner. The experimental results show the advantages of our method over the traditional DPM model.

References

Achtert, E.; Goldhofer, S.; Kriegel, H.-P.; Schubert, E.; and Zimek, A. 2012. Evaluation of clusterings - metrics and visual support. In *ICDE*.

Antoniak, C. E. 1974. Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *Annals of Statistics*.

Blei, D. M., and Jordan, M. I. 2005. Variational inference for dirichlet process mixtures. *Bayesian Analysis* 1:121–144.

Chen, G.; Meng, X.; Hu, T.; Guo, X. Y.; Liu, L.-X.; and Zhang, H. 2009. A multiphase region-based framework for image segmentation based on least square method. In *ICIP*, 3961–3964. IEEE Press.

Chen, C.; Zhu, J.; and Zhang, X. 2014. Robust bayesian max-margin clustering. In *NIPS*. Curran Associates, Inc. 532–540.

Crammer, K.; Dekel, O.; Keshet, J.; Shalev-Shwartz, S.; and Singer, Y. 2006. Online passive-aggressive algorithms. *JMLR*.

Ferguson, T. S. 1973. A Bayesian analysis of some nonparametric problems. *The Annals of Statistics* 209–230.

Fu, L., and Medico, E. 2007. Flame, a novel fuzzy clustering method for the analysis of dna microarray data. *BMC Bioinformatics* 8(1):3.

Gilks, W. R., and Wild, P. 1992. Adaptive rejection sampling for gibbs sampling. *Applied Statistics* 337–348.

Gionis, A.; Mannila, H.; and Tsaparas, P. 2007. Clustering aggregation. *ACM Trans. Knowl. Discov. Data* 1(1):4.

Görür, D., and Rasmussen, C. E. 2010. Dirichlet process gaussian mixture models: Choice of the base distribution. *J. Comput. Sci. Technol.*

Hannah, L. A.; Blei, D. M.; and Powell, W. B. 2011. Dirichlet process mixtures of generalized linear models. *JMLR* 1923–1953.

Hoai, M., and Zisserman, A. 2013. Discriminative subcategorization. In *CVPR*.

Hubert, L., and Arabie, P. 1985. Comparing partitions. *Journal of classification* 2(1):193–218.

Jain, A. K. 2007. Data clustering: User's dilemma. In *MLDM*, volume 4571, 1.

Jebara, T., and Pentland, A. 1998. Maximum conditional likelihood via bound maximization and the cem algorithm. In *NIPS*.

Knowles, D. A.; Palla, K.; and Ghahramani, Z. 2012. A nonparametric variable clustering model. In *NIPS*, 2996–3004.

Kurihara, K.; Welling, M.; and Teh, Y. 2007. Collapsed variational Dirichlet process mixture models. In *Proc. Int. Jt. Conf. Artif. Intell.*, volume 20, 2796–2801.

Lafferty, J. D.; McCallum, A.; and Pereira, F. C. N. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 282–289.

Minka, T. P. 2003. A comparsion of numerical optimizers for logistic regression. Technical report.

Neal, R. M. 2000. Markov chain sampling methods for dirichlet process mixture models. *JOURNAL OF COMPUTATIONAL AND GRAPHICAL STATISTICS* 249–265.

Nguyen, V.; Phung, D. Q.; Nguyen, X.; Venkatesh, S.; and Bui, H. H. 2014. Bayesian nonparametric multilevel clustering with group-level contexts. In *ICML*.

Nigam, K.; Lafferty, J.; and McCallum, A. 1999. Using maximum entropy for text classification. In *IJCAI-99 Workshop on Machine Learning for Information Filtering*.

Platt, J. C. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, 61–74.

Rand, W. 1971. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association* 66(336):846–850.

Rasmussen, C. E. 2000. The infinite gaussian mixture model. In *NIPS*.

Rosenberg, A., and Hirschberg, J. 2007. V-measure: A conditional entropy-based external cluster evaluation measure. In *EMNLP-CoNLL*, 410–420.

Sethuraman, J., and Tiwari, R. C. 1981. Convergence of Dirichlet measures and the interpretation of their parameter. Technical report.

Shahbaba, B.; Neal, R.; and Ghahramani, Z. 2009. Nonlinear models using dirichlet process mixtures. *JMLR* 1829–1850.

Sudderth, E. B. 2006. *Graphical models for visual object recognition and tracking*. Ph.D. Dissertation, MIT.

Sutton, C., and Mccallum, A. 2006. *Introduction to Statistical Relational Learning*. MIT Press.

Teh, Y. W. 2010. Dirichlet processes. In *Encyclopedia of Machine Learning*. Springer.

Tsochantaridis, I.; Joachims, T.; Hofmann, T.; and Altun, Y. 2005. Large margin methods for structured and interdependent output variables. *JMLR* 1453–1484.

Vapnik, V. N. 1995. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc.

Vlachos, A.; Ghahramani, Z.; and Korhonen, A. 2008. Dirichlet process mixture models for verb clustering. In *ICML workshop*.

Xiao, L.; Zhou, D.; and Wu, M. 2011. Hierarchical classification via orthogonal transfer. In *ICML*.

Xu, L.; Neufeld, J.; Larson, B.; and Schuurmans, D. 2005. Maximum margin clustering. In *NIPS17*, 1537–1544.

Zhang, Z.; Dai, G.; and Jordan, M. I. 2010. Matrix-variate dirichlet process mixture models. In *AISTATS*.