PAPER

Stochastic Divergence Minimization for Biterm Topic Models

Zhenghang CUI^{†a)}, Issei SATO^{†,††b)}, *Nonmembers*, and Masashi SUGIYAMA^{†,††c)}, *Member*

SUMMARY As the emergence and the thriving development of social networks, a huge number of short texts are accumulated and need to be processed. Inferring latent topics of collected short texts is an essential task for understanding its hidden structure and predicting new contents. A biterm topic model (BTM) was recently proposed for short texts to overcome the sparseness of document-level word co-occurrences by directly modeling the generation process of word pairs. Stochastic inference algorithms based on collapsed Gibbs sampling (CGS) and collapsed variational inference have been proposed for BTM. However, they either require large computational complexity, or rely on very crude estimation that does not preserve sufficient statistics. In this work, we develop a stochastic divergence minimization (SDM) inference algorithm for BTM to achieve better predictive likelihood in a scalable way. Experiments show that SDM-BTM trained by 30% data outperforms the best existing algorithm trained by full data.

key words: short text, topic model, biterm, stochastic inference algorithm

1. Introduction

As social network services are becoming dominant in people's daily life, a huge number of short text data have been accumulated. At the same time, other data which can be found on traditional web pages, such as article titles, can also be regarded as possessing the same attribute of short length. It would be an essential and interesting task to explore their inner structure for a wide range of applications, such as classification based on contents, or trend prediction. Because of the document level word co-occurrence sparsity caused by short document length, conventional topic models such as probabilistic latent semantic indexing (pLSA) proposed by Hofmann [1] or latent Dirichlet allocation (LDA) proposed by Blei et al. [2] fail to show favorable inference performance on short text data sets. Cheng et al. [3] proposed a biterm topic model (BTM) to alleviate the problem caused by document level word co-occurrence sparsity. Instead of each single word, the generation process of each unordered combination of two words, or a *biterm*, is modeled in BTM. Compared to conventional topic models, this modification makes BTM less sensitive to the shortness of each document, and more stable to clearly reveal the rela-

tionship between words. By explicitly modeling the word co-occurrences and combining words into biterms, it has been empirically shown by Cheng et al. [3] that BTM successfully alleviates the problem caused by document level word co-occurrence sparsity and keeps the generality and flexibility at the same time.

In order to infer model parameters and estimate latent topics for BTM, Cheng et al. [3] first proposed a batch inference algorithm based on collapsed Gibbs sampling (CGS) [4]. They also proposed two online CGS algorithms, the online Biterm (oBTM) algorithm and the incremental Biterm (iBTM) algorithm, to scale up for data sets of large size. The oBTM algorithm is based on the idea of updating hyperparameters between time slices, which is inspired by the online LDA algorithm proposed by AlSumait et al. [5]. On the other hand, the iBTM algorithm is based on the idea of resampling topics of observed biterms for a sufficient number of times after a new biterm is observed, which is inspired by an incremental Gibbs sampler for LDA proposed by Canini et al. [6]. However, these online algorithms for BTM are not working very efficiently in terms of memory usage.

In the variational Bayesian (VB) approach, collapsed variational Bayesian (CVB) algorithm, which only assumes the independence between latent variables, was proposed for LDA by Teh et al. [7]. Intractable integrals that exist in update formulas are evaluated by second-order Taylor approximation. Later, Asuncion et al. [8] reported that zero-order Taylor approximation can achieve better performance. A stochastic version of this algorithm, called SCVB0, was then proposed for LDA by Foulds et al. [9]. SCVB0 for LDA is based on heuristic derivation and its computational complexity per mini-batch depends on the vocabulary size. Although a similar SCVB0 algorithm for BTM was proposed by Awaya et al. [10] for better latent topics estimation, the reason why zero-order approximation is used instead of higher order approximations was not clearly explained for this model. Meanwhile, the risk of arithmetic underflow in floating point calculations exists when processing data sets of very large size, because SCVB0 for the BTM algorithm utilizes a scale coefficient to reduce the computational complexity of each iteration from $O(W)$ to $O(1)$, where W denotes the size of the vocabulary. Although as discussed in Awaya et al. [10], it rarely happens in practice and they proposed a workaround to avoid the underflow, it is still a drawback for an algorithm which has non-zero probability of causing arithmetic underflow and it is favorable to have an

Manuscript received September 26, 2017.

Manuscript revised November 30, 2017.

Manuscript publicized December 20, 2017.

[†]The authors are with the University of Tokyo, Tokyo, 113-0033 Japan.

^{††}The authors are with RIKEN, Tokyo, 103-0027 Japan.

a) E-mail: cui@ms.k.u-tokyo.ac.jp

b) E-mail: sato@k.u-tokyo.ac.jp

c) E-mail: sugi@k.u-tokyo.ac.jp

DOI: 10.1587/transinf.2017EDP7310

Table 1 Comparison of BTM inference algorithms. K denotes the total number of topics, N_B denotes the total number of biterns, W denotes the vocabulary size, B_t denotes the size of a biterm mini-batch and N and R are hyperparameters.

	Update Cost	Memory	Sufficient Statistics Preservation
oBTM	$O(KNB_t)$	$O(K(1+W) + B_t)$	YES
iBTM	$O(K R)$	$O(K(1+W) + N_B)$	YES
SCVB0 BTM	$O(K)$	$O(K(1+W))$	NO
SDM (This work)	$O(K)$	$O(K(1+W))$	YES

algorithm that does not have this problem in the first place. Moreover, it also approximates sufficient statistics in a crude way that they are not preserved.

Considering the issues discussed above, we propose a novel stochastic divergence minimization (SDM) algorithm for BTM based on minimizing the α -divergence to achieve better predictive performance[†]. For LDA, Sato et al. [12] showed that CVB0 update formulas actually solve a minimization problem based on α -divergence, and proposed an SDM algorithm [13] to alleviate the problems of SCVB0 for LDA. Their derivation and theory are highly specific to LDA and they have not been applied for any other models yet, therefore it is unclear if they can be applied for BTM. Moreover, unlike document level variables that are stochastically optimized in LDA, BTM models data in biterns and it is not trivial how to optimize biterm level variables. Inspired by these works, we first reformulate the zero-order approximation algorithm CVB0 for BTM as the answer to an α -divergence minimization problem. Then, we apply a stochastic approximation method to this optimization problem to develop a stochastic inference algorithm. This algorithm requires the same amount of memory and has the same computational complexity as SCVB0 for BTM, but provides better approximation in terms of α -divergence minimization formulation of CVB0.

In summary, we provide the following two contributions in this paper:

- We provide a novel interpretation of SCVB0 inference for BTM from the perspective of α -divergence minimization. Based on this interpretation, we derive an SDM algorithm for BTM.
- We analyze the convergence of our proposed method by using Martingale convergence theory.

Table 1 summarizes properties of this work and existing works. Our method requires smaller update costs and memory usage with sufficient statistics properly preserved.

The remainder of this paper is organized as follows. In Sect. 2, we review related works on BTM, their existing inference algorithms, and the theoretical background for SDM. In Sect. 3, we introduce our proposed SDM algorithm. In Sect. 4, we conduct experiments to compare our

proposed method against existing methods and discuss the result. In Sect. 5, we conclude this paper.

2. Related Works

In this section, we first describe the details of BTM. Then, we describe existing algorithms based on the sampling approach. Finally, we describe existing algorithms based on the variational Bayesian approach.

2.1 Biterm Topic Model (BTM)

BTM was proposed to alleviate the problem of document level word co-occurrence sparsity by modifying the word generating part of the model. Instead of modeling the generation of each word, BTM directly models the generation of biterns, which are unordered combinations of two unique words. For example, a document of n unique words will generate $\binom{n}{2}$ combinations of two words. Compared to conventional topic models, this modification makes BTM less sensitive to the short length of each document, and biterns are more stable to clearly reveal the relationship between words. The notation used in the following is listed below.

- Each biterm is denoted by $b_i = \{w_{i1}, w_{i2}\}$.
- θ denotes the topic proportion vector. Its length is K and all of its entries sum to 1.
- Φ denotes the word distribution matrix with size $K \times W$. Each row vector ϕ_k has length W and sums to 1.
- z_i denotes the topic indicator variable for biterm b_i .

The generation process is described formally as follows.

1. Draw $\theta \sim \text{Dirichlet}(\gamma)$.
2. For each topic k ,
 - a. Draw $\phi_k \sim \text{Dirichlet}(\beta)$.
3. For each biterm b_i ,
 - a. Draw $z_i \sim \text{Multinomial}(\theta)$.
 - b. Draw $w_{i1}, w_{i2} \sim \text{Multinomial}(\phi_{z_i})$.

In the above, $\text{Dirichlet}(\gamma)$ denotes a Dirichlet distribution with parameter γ , and $\text{Multinomial}(\theta)$ denotes a multinomial distribution with parameter θ .

2.2 Collapsed Gibbs Sampling Algorithms

To infer the posterior distribution of a probabilistic model, collapsed Gibbs sampling (CGS) and variational Bayes (VB) are two main approaches. We will first introduce algorithms of the CGS approach.

2.2.1 Batch CGS Inference Algorithm

A batch CGS inference algorithm [3] estimates all of the three parameters, z which indicates the topic assignments, θ which indicates the topic proportion and Φ which indicates

[†]Readers can refer to Minka [11] for the theory of α -divergences.

the word distribution for each topic. Parameters θ and Φ are first integrated out, z_i for each biterm b_i is then sampled by using the posterior distribution conditioned on all of the other variables. However, this algorithm cannot efficiently scale to large data sets.

The formula used to sample z_i for each biterm b_i is

$$P(z_i = k | z^{\setminus i}, B) \propto (n_k^{\setminus i} + \gamma) \frac{(n_{w_1|k}^{\setminus i} + \beta)(n_{w_2|k}^{\setminus i} + \beta)}{(n_{\cdot|k}^{\setminus i} + W\beta)(n_{\cdot|k}^{\setminus i} + W\beta + 1)}, \quad (1)$$

where $z^{\setminus i}$ is the whole topic assignment vector without considering b_i , $n_k^{\setminus i}$ is the count of biterms assigned to topic k without counting b_i and $n_{w|k}^{\setminus i}$ is the count of times that word w is assigned to topic k without counting b_i . The dot in $n_{\cdot|k}^{\setminus i}$ means taking the sum over all words. After a sufficient number of iterations over the whole data set, we can restore θ and Φ using following estimators which are used for calculating predictive probability Eq. (24) in experiments:

$$\begin{aligned} \phi_{k,w} &= \frac{n_{w|k} + \beta}{n_{\cdot|k} + W\beta}, \\ \theta_k &= \frac{n_k + \gamma}{N_B + K\gamma}, \end{aligned} \quad (2)$$

where n_k is the number of biterms assigned to topic k and $n_{w|k}$ is the count of times that word w is assigned to topic k . The dot in $n_{\cdot|k}$ means taking the sum over all words.

2.2.2 Online BTM Algorithm (oBTM)

Cheng et al. [3] introduced two algorithms to deal with large data sets. We will introduce the oBTM algorithm here, which is inspired by the similar algorithm proposed for LDA by AlSumait et al. [5].

For this algorithm, the data set is supposed to be separated into mini-batches. The mini-batch with index t is denoted by $B^{(t)}$. The number of biterms assigned to topic k within $B^{(t)}$ is denoted by $n_k^{(t)}$. The number of times word w is assigned to topic k within $B^{(t)}$ is denoted by $n_{w|k}^{(t)}$. Hyperparameters for θ are denoted by vector $\{\gamma_1, \dots, \gamma_K\}$ and hyperparameters for Φ are denoted by matrix $\{\beta_1, \dots, \beta_K\}$, where β_k is a vector consisting of $\{\beta_{k,1}, \dots, \beta_{k,W}\}$.

This algorithm fits parameters over each mini-batch, and uses $n_k^{(t)}$ and $n_{w|k}^{(t)}$ to adjust hyperparameters for the next mini-batch. As one loop through the mini-batch may not yield good results, the algorithm requires to repeat the loop N times per mini-batch, where N is a hyperparameter. The conditional distribution for sampling each topic z_i can be found in the original paper by Cheng et al. [3]. After processing each mini-batch, hyperparameters are updated by

$$\begin{aligned} \gamma_k^{(t+1)} &= \gamma_k^{(t)} + \lambda n_k^{(t)}, \\ \beta_{k,w}^{(t+1)} &= \beta_{k,w}^{(t)} + \lambda n_{w|k}^{(t)}, \end{aligned} \quad (3)$$

where $\lambda \in [0, 1]$ denotes the decay weight. The procedure is outlined in Alg. 1. Within the processing of each mini-batch,

Algorithm 1 Online BTM Algorithm

```

Set  $\gamma^{(1)} = \{\gamma_1, \dots, \gamma_K\}$ ,  $\beta_k = \{\beta_{k,1}, \dots, \beta_{k,W}\}_{k=1}^K$ .
for  $t = 1$  to  $T$  do
  for  $b_i \in B^{(t)}$  do
    Sample a topic for  $b_i$  using a uniform distribution.
  end for
  for iterations do
    for  $b_i \in B^{(t)}$  do
      Sample a topic for  $b_i$  using Eq. (1).
      Update  $n_k^{(t)}$ ,  $n_{w_1|k}^{(t)}$  and  $n_{w_2|k}^{(t)}$ .
      Update  $\gamma^{(t)}$  and  $\{\beta_k^{(t)}\}_{k=1}^K$  using Eq. (3).
    end for
  end for
  Compute  $\theta^{(t)}$  and  $\Phi^{(t)}$  using Eq. (2).
end for

```

Algorithm 2 Incremental BTM Algorithm

```

for  $b_i \in B$  do
  Sample a topic for  $b_i$ .
  Update statistics of  $n_k$  and  $n_{w|k}$ .
  Generate rejuvenation sequence  $R$ .
  for  $b_j \in R$  do
    Sample a topic for  $b_j$  using Eq. (1).
    Update statistics of  $n_k$  and  $n_{w|k}$ .
  end for
end for
Compute global parameters  $\theta$  and  $\Phi$  using Eq. (2).

```

the algorithm iterates N times to sample the topics for each biterm in the mini-batch. Each sampling requires the calculation of probabilities for each one of K topics. Therefore the computational cost for each loop is KNB_t .

2.2.3 Incremental BTM Algorithm (iBTM)

The oBTM algorithm cannot update parameters immediately after the arrival of a new biterm. The iBTM algorithm is proposed to cope with this problem. This idea is inspired by the incremental Gibbs sampler proposed by Canini et al. [6]. Specifically, the main idea is that after the arrival of a new biterm, the algorithm executes the following two steps. It first samples the topic assignment for the new biterm. Then, it randomly constructs a biterm sequence called a *rejuvenation sequence* from all of the existing biterms, and resamples the topic assignments of biterms in the sequence. The procedure is outlined in Alg. 2. Within the processing of each biterm, topics are sampled for each biterm in the rejuvenation sequence with length $|R|$. Each sampling requires the calculation of probabilities for each one of K topics. So the computational cost for each loop is $K|R|$.

2.3 Variational Bayes (VB) Algorithms

Here, we will introduce inference algorithms of the variational Bayesian (VB) approach. The basic VB for BTM has not been explicitly proposed. Its objective can be formulated as $\arg\min_q \text{KL}[q(\theta, \Phi, z) \| p(\theta, \Phi, z | B)]$, where KL denotes the

Kullback-Leibler divergence and $q(\cdot)$ denotes the variational distribution.

2.3.1 CVB0 Algorithm for BTM

VB for BTM assumes the independence among all elements of θ , Φ and z , which is a very strong assumption and may prevent VB from achieving good approximation. To mitigate this problem, Teh et al. [7] proposed collapsed Variational Bayesian (CVB) inference for LDA. CVB only assumes the independence of latent variables and its objective can be formulated as $\text{argmin}_q \text{KL}[q(z)||p(z|B)]$.

For evaluating intractable expectation terms, Asuncion et al. [8] found that zero-order Taylor approximation outperforms second-order Taylor approximation. Using this approximation, the updating formula for variational parameter $\hat{z}_{i,k}$ presented by Awaya et al. [10] can be deduced as

$$\hat{z}_{i,k} \propto (N_k^{\setminus i} + \alpha) \frac{(N_{w|1|k}^{\setminus i} + \beta)(N_{w|2|k}^{\setminus i} + \beta)}{(2N_k^{\setminus i} + W\beta)(2N_k^{\setminus i} + W\beta + 1)}, \quad (4)$$

where $N_k = \sum_{b_i \in B} \hat{z}_{i,k}$, $N_{w|k} = \sum_{b_i \in B_w} \hat{z}_{i,k}$, B_w denotes the set of biterms containing word w and $\setminus i$ means counting without considering b_i . Note that $\hat{z}_{i,k}$ represents the probability that the biterm b_i is assigned to the topic k .

2.3.2 SCVB0 Algorithm for BTM

CVB0 for BTM is a batch algorithm. Its stochastic version, SCVB0 for BTM, has been developed by Awaya et al. [10] based on the idea of ignoring the subtraction of the current biterm and updating sufficient statistics of latent variables in a stochastic way. Storing all variational parameters is not necessary and a very crude estimation of N_k and $N_{w|k}$ when biterm b_i is observed can be expressed by

$$\hat{N}_k = |B| \hat{z}_{i,k}, \quad (5)$$

$$\hat{N}_{w|k} = \begin{cases} |B| \hat{z}_{i,k} & \text{if } w \in b_i, \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

This estimation does not preserve sufficient statistics because $\sum_k \hat{N}_{w|k} \neq n_w$, where n_w denotes how many times the word w appears. Then, N_k and $N_{w|k}$ are updated using the following formulas:

$$N_k \leftarrow (1 - \rho_t)N_k + \rho_t \hat{N}_k, \quad (7)$$

$$N_{w|k} \leftarrow (1 - \rho_t)N_{w|k} + \rho_t \hat{N}_{w|k}, \quad (8)$$

where ρ_t denotes the step size.

Naively updating each $N_{w|k}$ for all the words requires the computational complexity of $O(W)$. To reduce this to $O(1)$, Awaya et al. [10] proposed the following technique implicitly represent the value of $N_{w|k}$. A dummy matrix $A_{w|k}$, which has the same size as $N_{w|k}$, and a scaling coefficient a are stored. It is assumed that $N_{w|k} = a \times A_{w|k}$ are satisfied for each entry. Every time when we need to update $N_{w|k}$, we can

Algorithm 3 SCVB0 BTM Algorithm

```

Randomly initialize  $N_k$  and  $N_{w|k}$ .
for  $b_i \in B$  do
  for each topic  $k$  do
    Compute  $\hat{z}_{i,k}$  using Eq. (4).
    Update  $N_k$  using Eq. (7) and  $N_{w|k}$  using Eq. (8).
  end for
end for
Compute global parameters  $\theta$  and  $\Phi$  using Eq. (2).
```

just multiply a by $(1 - \rho_t)$ and manually computes the values of $A_{w|1|k}$ and $A_{w|2|k}$. The assumption $N_{w|k} = a \times A_{w|k}$ still holds and the computational complexity is $O(1)$. The step size ρ_t is usually defined as $\rho_t = (t + \tau)^{-\kappa}$, where t denotes the time step and τ and κ are hyperparameters.

After processing all of the biterms, global parameters are restored by $\theta_k \propto N_k + \alpha$ and $\phi_{k,w} \propto N_{w|k} + \beta$. The procedure is outlined in Alg. 3. Within the processing of each biterm, only $\hat{z}_{i,k}$ is updated for each topic. Each update costs a constant time, therefore the computational cost for each loop is K .

2.4 α -divergence

We will introduce the concepts of α -divergence and local divergence projection inference. More details can be found in the work by Minka [11] and Amari [14].

2.4.1 Definition of α -divergence

The α -divergence can be perceived as a generalized KL divergence. The α -divergence from $p(x)$ to $q(x)$, indexed by $\alpha \in (-\infty, \infty)$, is defined as

$$D_\alpha[p||q] = \frac{\int \alpha p(x) + (1 - \alpha)q(x) - p(x)^\alpha q(x)^{1-\alpha} dx}{\alpha(1 - \alpha)}.$$

Notice that $p(x)$ and $q(x)$ need not to be normalized.

In this work, the purpose of using α -divergence is to reformulate CVB0, therefore the value of α is fixed.

2.4.2 Local α -Divergence Projection

Suppose that the distribution $q(x)$ we approximate can be fully factorized. That is, $q(x) = \prod_{i=1}^n q(x_i)$, where x_i denotes the i -th element of the vector x . $x = (x_1, x_2, \dots, x_n)^\top$, where $\top$ denotes the transpose. Depending on $p(x)$, in many cases the minimization of $D_\alpha[p(x)||q(x)]$ is intractable, we resort to the local divergence minimization in which we substitute $p(x_i|x^{\setminus i})p(x^{\setminus i})$ for $p(x_i|x^{\setminus i})q(x^{\setminus i})$. Local divergence minimization is also used in other algorithms such as the expectation propagation algorithm [16]. We focus on each single x_i and then optimize each $q(x_i)$ by taking $\text{argmin}_{q(x_i)} D_\alpha[p(x_i|x^{\setminus i})q(x^{\setminus i})||q(x)]$, where $x^{\setminus i}$ represents all but the i -th entry of x and $q(x) = q(x_i|x^{\setminus i})q(x^{\setminus i})$. Its update formula can be obtained by taking the derivative of the α -divergence and equating it to zero:

$$q(x_i) \propto \mathbb{E}_{q(x^{(i)})} \left[\left(\frac{p(x)}{q(x^{(i)})} \right)^\alpha \right]^{\frac{1}{\alpha}}.$$

Since the expectation over $q(x^{(i)})$ is computationally intractable, we approximate it by first splitting the numerator as $q(x_i) \propto \mathbb{E}_{q(x^{(i)})} \left[\left(p(x_i|x^{(i)}) \frac{p(x^{(i)})}{q(x^{(i)})} \right)^\alpha \right]^{\frac{1}{\alpha}}$. We then substitute $p(x^{(i)})$ with $q(x^{(i)})$ to obtain the approximation

$$q(x_i) \propto \mathbb{E}_{q(x^{(i)})} [(p(x_i|x^{(i)}))^\alpha]^{\frac{1}{\alpha}}.$$

This is the method we will use in the derivation of a stochastic divergence minimization algorithm to approximate the α -divergence.

3. Proposed Method

SCVB0 for BTM is based on heuristic update formulas. Sufficient statistics are not preserved since $\sum_{k=1}^K \hat{N}_{w|k} \neq N_w$, which is also observed in experiments. Moreover, although the scaling coefficient a significantly reduces the computational complexity, it bears the risk of underflow with large datasets, when $(1 - \rho_t)$ is multiplied repeatedly during the processing. We address these problems in this section.

3.1 Derivation of Divergence Minimization

We assume the independence between latent topics of the given bitersms as $q(B, z) = \prod_{i=1}^{N_B} q(b_i, z_i)$. We then estimate this distribution by α -divergence minimization:

$$q^*(B, z) = \operatorname{argmin}_{q(B, z)} D_\alpha[p(B, z) \| q(B, z)], \quad (9)$$

where $q^*(B, z)$ denotes the optimized distribution. We used joint distributions to apply α -divergence minimization. After the derivation for joint distributions, we obtain the posterior distributions by marginalization.

Since it is intractable to compute this minimization, we consider the following local divergence minimization. Note that

$$p(b_i, z_i = k) \propto (n_k^{(i)} + \gamma) \frac{(n_{w_{i1}|k}^{(i)} + \beta)(n_{w_{i2}|k}^{(i)} + \beta)}{(n_{\cdot|k}^{(i)} + W\beta)(n_{\cdot|k}^{(i)} + W\beta + 1)}. \quad (10)$$

We then parameterize $q(b_i, z_i)$ as follows:

$$q(b_i, z_i = k) = \frac{a_k^{(i)} b_{k, w_{i1}}^{(i)} b_{k, w_{i2}}^{(i)}}{c_k^{(i)} (c_k^{(i)} + 1)}, \quad (11)$$

$$a_k^{(i)} = \tilde{n}_k^{(i)} + \gamma, \quad b_{k, w}^{(i)} = \tilde{n}_{w|k}^{(i)} + \beta, \quad c_k^{(i)} = \tilde{n}_{\cdot|k}^{(i)} + W\beta. \quad (12)$$

Note that $\tilde{n}_k^{(i)}$, $\tilde{n}_{w|k}^{(i)}$ and $\tilde{n}_{\cdot|k}^{(i)}$ here are not counts, but they are just parameters of the function defined above.

We will solve it for $a_k^{(i)}$ as an example. For the joint variational distribution without the term of $a_k^{(i)}$, we define

$$q^{(a)}(b_i, z_i = k) = \frac{b_{k, w_{i1}}^{(i)} b_{k, w_{i2}}^{(i)}}{c_k^{(i)} (c_k^{(i)} + 1)}. \quad (13)$$

Recalling that the α -divergence does not need the distributions to be normalized, we can define the following local projections:

$$(a_k^{(i)})^* = \operatorname{argmin}_{a_k} D_\alpha[(n_k^{(i)} + \gamma) q^{(a)}(B, z) \| a_k^{(i)} q^{(a)}(B, z)], \quad (14)$$

where

$$q^{(a)}(B, z) = q^{(a)}(b_i, z_i) q(B^{(i)}, z^{(i)}). \quad (15)$$

Taking the derivative of Eq. (14) with respect to $a^{(i)}$ and equating it to zero yields

$$\sum_{z^{(i)}} q^{(a, i)}(b_i, z) - \frac{\sum_{z^{(i)}} (n_k^{(i)} + \gamma)^\alpha q^{(a, i)}(b_i, z)}{(a_k^{(i)})^\alpha} = 0. \quad (16)$$

With $\sum_{z^{(i)}} q(z^{(i)}) = 1$ and $\alpha = 1$, we obtain

$$\begin{aligned} a_k^{(i)} &= \left[\sum_{z^{(i)}} (n^{(i, k)} + \gamma) q(z^{(i)}) \right] \\ &= \mathbb{E}_{q(z^{(i)})} [(n^{(i, k)} + \gamma)] \\ &= \mathbb{E}_{q(z^{(i)})} [n^{(i, k)}] + \gamma. \end{aligned} \quad (17)$$

Similarly, we can derive solutions to other terms with $\alpha = 1$ for $b_{k, w}^{(i)}$ and $\alpha = -1$ for $c_k^{(i)}$ as

$$b_{k, w}^{(i)} = \mathbb{E}_{q(z^{(i)})} [n_{w|k}^{(i)}] + \beta, \quad c_k^{(i)} = \mathbb{E}_{q(z^{(i)})} [n_{\cdot|k}^{(i)}] + W\beta.$$

We then obtain the update formula for $q(z_i)$ as

$$\begin{aligned} q(z_i = k | b_i) &\propto (\mathbb{E}[n_k^{(i)}] + \gamma) \\ &\times \frac{(\mathbb{E}[n_{w_{i1}|k}^{(i)}] + \beta)(\mathbb{E}[n_{w_{i2}|k}^{(i)}] + \beta)}{(\mathbb{E}[n_{\cdot|k}^{(i)}] + W\beta)(\mathbb{E}[n_{\cdot|k}^{(i)}] + W\beta + 1)}, \end{aligned} \quad (18)$$

which is identical to the one obtained by SCVB0 for BTM.

3.2 Derivation of SDM for BTM

For the three terms we defined in α -divergence minimization, we have $\mathbb{E}[n_k^{(i)}] = 2 \mathbb{E}[n_{\cdot|k}^{(i)}]$. Therefore, $a_k^{(i)}$ can be restored from $c_k^{(i)}$ and we need only to compute the values of $b_k^{(i)}$. $c_k^{(i)}$ can be calculated as $c_k^{(i)} = \sum_{w=1}^W b_{k, w}^{(i)}$, so we can update it by

$$c_k^{(i)} \leftarrow c_k^{(i)} - (b_{k, w}^{(i)})^{(\text{old})} + (b_{k, w}^{(i)})^{(\text{new})}. \quad (19)$$

For this reason, we will only focus on the stochastic approximation for the term $b_{k, w}^{(i)}$.

We can rewrite $b_{k, w}^{(i)}$ in the form of fixed point iteration with step size ρ_t :

$$\begin{aligned} (b_{k, w}^{(i)})^{(t+1)} &= (1 - \rho_t)(b_{k, w}^{(i)})^{(t)} + \rho_t(\mathbb{E}[n_{w|k}^{(i)}] + \beta) \\ &= (b_{k, w}^{(i)})^{(t)} + \rho_t(\mathbb{E}[n_{w|k}^{(i)}] + \beta - (b_{k, w}^{(i)})^{(t)}). \end{aligned} \quad (20)$$

We then replace $\mathbb{E}[n_{w|k}^{(i)}]$ with approximation $\tilde{n}_{w|k}^{(i)}$, which is

Algorithm 4 SDM BTM Algorithm

```

Randomly initialize  $b_{k,w}$  and compute  $c_k$ .
for  $b_i \in B$  do
  for  $w \in b_i$  do
    Update  $b_{k,w}^{t(w)}$  using Eq. (23).
    Update  $c_k$  using Eq. (19).
    Increment  $t(w)$ .
  end for
end for
Compute global parameters  $\theta$  and  $\Phi$  using Eq. (2).

```

defined by

$$\tilde{n}_{w|k}^i = (n_w - 1)q(z_{i'} = k|b_{i'}), \quad (21)$$

where $i' \neq i$ is a random sample with $b_{i'}$ also containing the word w . This estimation preserves sufficient statistics because $\sum_k \tilde{n}_{w|k}^i = n_w^i$. We then substitute it into the update formula:

$$(b_{k,w}^i)^{(t+1)} = (b_{k,w}^i)^{(t)} + \rho_t(\tilde{n}_{w|k}^i + \beta - (b_{k,w}^i)^{(t)}). \quad (22)$$

This update formula is actually a realization of update based on word vocabulary. It only updates for a word w when it appears in a biterm. Furthermore, we process all the biterns in an one-pass fashion, which means $(b_{k,w}^i)^{(t)}$ indicates the latest value when word w appeared in a previous biterm and a previous updated value can be used for $q(z_{i'} = k|b_{i'})$. For this reason, we focus on the perspective of word type update and reformulate the update formula as

$$b_{k,w}^{t(w)+1} = b_{k,w}^{t(w)} + \rho_{t(w)}(\tilde{n}_{w|k}^i + \beta - (b_{k,w}^i)^{t(w)}), \quad (23)$$

where $t(w)$ represents how many times the word w has been updated so far and $\rho_{t(w)} = 1/(1 + t(w))^\kappa$, and we may make $b_{k,w}^{t(w)+1} = (b_{k,w}^i)^{(t+1)}$. The procedure is outlined in Alg. 4. Within the processing of each biterm, updates for $b_{k,w}$ needs the sampling for the biterns $b_{i'}$ and b_i , which requires K steps of basic computations for each one. The update for c_w and $t(w)$ can be executed in constant time, therefore the computational cost for each loop is K .

4. Experiments

In this section, we experimentally compare the proposed and existing methods.

4.1 Experimental Settings

We used the data set *Tweets2011* (<http://trec.nist.gov/data/tweets>) to evaluate the algorithms. The raw texts are in multiple languages and very noisy. We applied a filter to keep only English tweets and obtained the bag-of-words representation through preprocessing. As shown in Fig. 1, most of the documents have length less than 10. There are around 3.5 million documents and the average document length is 6.94. We generated 95 million biterns and divided them into a training set and a test set by ratio 4 : 1.

Fig. 1 Document length distribution in Tweets2011.

We evaluated the four algorithms, namely, the proposed SDM algorithm, the SCVB0 algorithm for BTM, the online BTM algorithm, and the incremental BTM algorithm. For the existing methods, hyperparameters are set in the same way as the original paper. We set the length of the rejuvenation sequence to 10 for the incremental BTM algorithm and the inner iteration times to 10 for the online BTM algorithm. For all of the algorithms, we set γ to be $50/K$ and β to be 0.01. For step size hyperparameters, same as the setting in Awaya et al. [10] and Sato et al. [12], $\tau = 1000$ and $\kappa = 0.8$ are used for SCVB0, while $\kappa = 0.51$ is used for SDM. We used $\kappa = 0.51$ because it is used in the SDM algorithm for LDA and showed favorable effects. An inappropriate value for κ may result in the updates for $b_{k,w}$ that is unstable or slow to converge.

We evaluated the average predictive log likelihood of the test set B_{test} based on the model parameters calculated by each algorithm. When θ and Φ are estimated, the average predictive log likelihood can be calculated by

$$\begin{aligned}
& \frac{1}{|B_{\text{test}}|} \log p(B_{\text{test}}) \\
&= \frac{1}{|B_{\text{test}}|} \sum_{b_i \in B_{\text{test}}} \log p(b_i) \\
&= \frac{1}{|B_{\text{test}}|} \sum_{b_i \in B_{\text{test}}} \sum_{k=1}^K p(z_i = k|\theta_k) \\
&\quad p(w_{i1}|z_i = k, \phi_{k,w_{i1}})p(w_{i2}|z_i = k, \phi_{k,w_{i2}}) \\
&= \frac{1}{|B_{\text{test}}|} \sum_{b_i \in B_{\text{test}}} \sum_{k=1}^K \theta_k \phi_{k,w_{i1}} \phi_{k,w_{i2}}.
\end{aligned} \quad (24)$$

Each set of experiments is repeated 10 times. All of the experiments are conducted on an Ubuntu server with 2.9 GHz Intel Xeon E5-2667 CPU and 64 GB memory.

4.2 Results

In Fig. 2, we compare the results of experiments conducted under different topic numbers. In Fig. 3, the horizontal axis represents the percentage of documents that have been processed. In both of these figures, the vertical axis represents the mean value of the average log likelihood of the test set,

Fig. 2 Average test-set likelihood after processing all data once.

Fig. 3 Average test set likelihood with the number of topics $K = 100$.

Fig. 4 Average running time for processing all data once.

which is the higher the more favorable.

On the other hand, in Fig. 4 and Fig. 5, instead of the percentage of documents, the running time is used to measure the progress. Standard deviations are also shown by error bars over 10 runs. SDM-BTM represents the proposed SDM algorithm for BTM. SCVB0-BTM represents the SCVB0 algorithm for BTM. iBTM represents the incremental BTM algorithm. oBTM represents the online BTM algorithm.

As shown in Fig. 2, Fig. 3 and Fig. 5, compared to online algorithms based on CGS, SDM-BTM shows a higher convergence speed and a better convergence result. SDM-BTM trained by 30% data outperforms SCVB0-BTM trained by full data. This is because SDM-BTM is a deter-

Fig. 5 Average test set likelihood with the number of topics $K = 100$.

ministic algorithm while CGS algorithms are based on sampling. CGS algorithms usually converge slower due to the inner characteristic of a Monte Carlo Markov Chain sampling algorithm. However, as shown in this experiment, deterministic algorithms such as SDM-BTM empirically converge faster and perform better. Compared to SCVB0-BTM, SDM-BTM performs better on a real world data set because SDM-BTM preserves the sufficient statistics correctly. Although SCVB0-BTM has the risk of arithmetic underflow when processing a large data set, it is not observed in our experiments.

As shown in Fig. 4, SDM-BTM shares a similar time consumption to SCVB0-BTM, which is very small compared to those of iBTM and oBTM. iBTM and oBTM consume infeasible time for processing all the data even with very small parameters.

5. Conclusions

In this paper, we first reviewed BTM and its existing inference algorithms. We then reconstructed CVB0 inference for BTM and proposed a novel SDM inference algorithm, which is a stochastic inference algorithm that can be applied to practical circumstances. It outperformed existing methods in our experiments.

For future work, it would be essential to explore the relationship between the number of topics and the performance. Developing new inference algorithms based on other α -divergences or conducting experiments on various data sets and studying their results would also be important challenges.

References

- [1] T. Hofmann, "Probabilistic latent semantic indexing," *SIGIR '99*, New York, NY, USA, pp.50–57, ACM, 1999.
- [2] D.M. Blei, A.Y. Ng, and M.I. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol.3, pp.993–1022, March 2003.
- [3] X. Cheng, X. Yan, Y. Lan, and J. Guo, "BTM: Topic modeling over short texts," *IEEE Trans. Knowl. Data Eng.*, vol.26, no.12, pp.2928–2941, Dec. 2014.
- [4] T.L. Griffiths and M. Steyvers, "Finding scientific topics," *Proceedings of the National Academy of Sciences*, vol.101, no.Suppl. 1,

- pp.5228–5235, April 2004.
- [5] L. AlSumait, D. Barbará, and C. Domeniconi, “On-line LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking,” *ICDM*, pp.3–12, Dec. 2008.
 - [6] K.R. Canini, L. Shi, and T.L. Griths, “Online inference of topics with latent Dirichlet allocation,” *Proceedings of AI Stats*, 2009.
 - [7] Y.W. Teh, D. Newman, and M. Welling, “A collapsed variational Bayesian inference algorithm for latent Dirichlet allocation,” *Proceedings of the 19th International Conference on Neural Information Processing Systems, NIPS’06*, Cambridge, MA, USA, pp.1353–1360, MIT Press, 2006.
 - [8] A. Asuncion, M. Welling, P. Smyth, and Y.W. Teh, “On smoothing and inference for topic models,” *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI ’09*, Arlington, Virginia, United States, pp.27–34, AUAI Press, 2009.
 - [9] J. Foulds, L. Boyles, C. DuBois, P. Smyth, and M. Welling, “Stochastic collapsed variational Bayesian inference for latent Dirichlet allocation,” *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’13*, New York, NY, USA, pp.446–454, ACM, 2013.
 - [10] N. Awaya, J. Kitazono, T. Omori, and S. Ozawa, “Stochastic collapsed variational Bayesian inference for biterm topic model,” *2016 International Joint Conference on Neural Networks (IJCNN)*, pp.3364–3370, July 2016.
 - [11] T. Minka, “Divergence measures and message passing,” tech. rep., 2005.
 - [12] I. Sato and H. Nakagawa, “Rethinking collapsed variational Bayes inference for LDA,” *Proceedings of the 29th International Conference on Machine Learning, ICML 2012*, Edinburgh, Scotland, UK, June 26 - July 1, 2012.
 - [13] I. Sato and H. Nakagawa, “Stochastic divergence minimization for online collapsed variational Bayes zero inference of latent Dirichlet allocation,” *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’15*, New York, NY, USA, pp.1035–1044, ACM, 2015.
 - [14] S. Amari, *Differential-geometrical methods in statistics*, Lecture notes in Statistics, Springer-Verlag, Berlin, Heidelberg, 1990.
 - [15] D.P. Bertsekas and J.N. Tsitsiklis, “Gradient convergence in gradient methods with errors,” *SIAM Journal on Optimization*, vol.10, no.3, pp.627–642, 2000.
 - [16] T.P. Minka, “A Family of Algorithms for Approximate Bayesian Inference,” Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA, USA, 2001.

Appendix A: Convergence Proof

In this appendix, we will show that the proposed SDM algorithm does converge to the minimum of the divergence. This is not obvious since the term used in updating is not a gradient of the divergence minimization problem. We will proof the convergence of the stochastic approximation by Eq. (22) for the optimization problem.

First, we will repeat the definition of the stochastic optimization problem based on Martingale convergence theory. The optimization problem is to find $b^* = \operatorname{argmin}_b f(b)$ and the update formula is $b^{(t+1)} = b^{(t)} + \rho_t s^{(t)}$. We also let $\mathcal{F}^{(t)}$ be the history of the variable sequence, which is also called *filtration*: $\mathcal{F}^{(t)} = \{b^{(1)}, \dots, b^{(t)}, s^{(1)}, \dots, s^{(t-1)}, \rho_1, \dots, \rho_t\}$.

Then we show the Martingale theory as follows.

Theorem 1. Assume step size ρ_t , function f and stochastic search direction $s^{(t)}$ satisfy following four conditions.

1. Step size ρ_t is a non-negative scalar and satisfies

$$\lim_{t \rightarrow \infty} \rho_t = 0, \sum_{t=1}^{\infty} \rho_t = \infty, \sum_{t=1}^{\infty} (\rho_t)^2 < \infty. \quad (\text{A} \cdot 1)$$

2. Function f is continuously differentiable and there exists some constant L such that

$$\|\nabla f(b) - \nabla f(\bar{b})\| \leq L\|b - \bar{b}\|, \forall b, \bar{b} \in \mathbb{R}^K. \quad (\text{A} \cdot 2)$$

3. There exists a positive constant C such that

$$C \nabla f(b^{(t)})^\top \mathbb{E}[s^{(t)} | \mathcal{F}^{(t)}] \leq -\|\nabla f(b^{(t)})\|^2, \forall t > 0. \quad (\text{A} \cdot 3)$$

4. There exist positive constants A and B such that

$$\mathbb{E}[\|s^{(t)}\|^2 | \mathcal{F}^{(t)}] \leq A + B\|\nabla f(b^{(t)})\|^2, \forall t > 0. \quad (\text{A} \cdot 4)$$

Then the update equation $b^{(t+1)} = b^{(t)} + \rho_t s^{(t)}$ satisfies the three holds with probability one.

1. The sequence $f(b^{(t)})$ converges.
2. $\lim_{t \rightarrow \infty} \nabla f(b^{(t)}) = 0$.
3. Every limit point of $b^{(t)}$ is a stationary point of f .

The detailed proof of above theory can be found in the super-Martingale convergence theorem [15].

First, we reform the stochastic update formula. Given the objective function as

$$\begin{aligned} D_1(b_{k,w_{i1}}^{i1}) &= D_1[(n_{w_{i1}|k}^{i1} + \beta)q^{b_{i1,i}}(B, z)] \|b_{z_i, w_{i1}}^{i1} q^{b_{i1,i}}(B, z)\| \\ &= D_1[(n_{w_{i1}|k}^{i1} + \beta)q^{b_{i1,i}}(b_i, z_i)q(B^{i1}, z^{i1})] \\ &\quad \|b_{z_i, w_{i1}}^{i1} q^{b_{i1,i}}(b_i, z_i)q(B^{i1}, z^{i1})\|, \end{aligned} \quad (\text{A} \cdot 5)$$

we take its derivatives regarding to $b_{k,w_{i1}}^{i1}$:

$$\frac{\partial}{\partial b_{k,w_{i1}}^{i1}} D_1(b_{k,w_{i1}}^{i1}) = -\frac{q^{b_{i1,i}}(b_i, z_i)}{b_{k,w_{i1}}^{i1}} (\mathbb{E}[n_{w_{i1}|k}^{i1}] + \beta - b_{k,w_{i1}}^{i1}). \quad (\text{A} \cdot 6)$$

Next, we define the stochastic direction

$$\begin{aligned} s_{i',k} &= n_{w_{i1}|k}^{i1} q(z_{i'} = k | b_{i'}) + \beta - b_{k,w_{i1}}^{i1} \\ &= \mathbb{E}[n_{w_{i1}|k}^{i1}] + \beta - b_{k,w_{i1}}^{i1} + \xi_{i',k}, \end{aligned} \quad (\text{A} \cdot 7)$$

where $\xi_{i',k} = n_{w_{i1}|k}^{i1} q(z_{i'} = k | b_{i'}) - \mathbb{E}[n_{w_{i1}|k}^{i1}]$ and $b_{i'}$ is another biterm that contains the word w_{i1} . Then we rewrite the stochastic direction as

$$s_{i',k} = \frac{b_{k,w_{i1}}^{i1} c_k^{i1} (c_k^{i1} + 1)}{d_k^{i1} b_{k,w_{i2}}^{i1}} \frac{\partial}{\partial b_{k,w_{i1}}^{i1}} D_1(b_{k,w_{i1}}^{i1}) + \xi_{i',k}. \quad (\text{A} \cdot 8)$$

The step size we actually use satisfies the first condition of Theorem 1 and it is not hard to show that the objective function $D_1(b_{k,w_{i1}}^{i1})$ of minimization satisfies the second condition. Therefore, what left is to show the satisfaction of

the third and forth condition. From the paper that introduced stochastic divergence minimization for LDA [13], the proof of following lemma can be found.

Lemma 1. *If the stochastic noise term $\xi^{(t)}$ satisfies the following conditions, then the stochastic direction $s^{(t)}$ satisfies the third and forth condition of the theorem 1.*

1. $\{\xi^{(t)}\}$ is a Martingale difference sequence with respect to filtration $\mathcal{F}^{(t)}$, which means that $\mathbb{E}[\xi^{(t)}|\mathcal{F}^{(t)}] = 0, \forall t > 0$.
2. $\xi^{(t)}$ has bounded variance. For example, it is square integrable with

$$\mathbb{E}[\|\xi^{(t)}\|^2|\mathcal{F}^{(t)}] \leq C(1 + \|\nabla D_1(b_{k,w_{i1}}^{(t)})\|^2), \forall t > 0, \quad (\text{A} \cdot 9)$$

for some constant C .

Therefore, what left is to prove the following lemma.

Lemma 2. *The noise term $\xi^{(t)}$ satisfies the two conditions listed in the Lemma 1.*

Proof. For the first condition, we show that

$$\begin{aligned} & \mathbb{E}[\xi_k^{(t)}|\mathcal{F}^{(t)}] \\ &= \mathbb{E}_{i'}[\xi_{i',k}] \\ &= n_{w_{i1}}^i \mathbb{E}_{i'}[q(z_{i'} = k|b_{i'})\delta(w_{i1} \in b_{i'})] - \mathbb{E}[n_{w_{i1}|k}^i] \\ &= \sum_{i' \neq i} q(z_{i'} = k|b_{i'})\delta(w_{i1} \in b_{i'}) - \mathbb{E}[n_{w_{i1}|k}^i] \\ &= 0. \end{aligned} \quad (\text{A} \cdot 10)$$

Therefore, the first condition is satisfied. Recall Eq. (A·6):

$$\frac{\partial}{\partial b_{k,w_{i1}}^i} D_1(b_{k,w_{i1}}^i) = -\frac{q^{b_1}(b_i, z_i)}{b_{k,w_{i1}}^i} (\mathbb{E}[n_{w_{i1}|k}^i] + \beta - b_{k,w_{i1}}^i).$$

We then define stochastic gradient as

$$\begin{aligned} & \frac{\partial}{\partial b_{k,w_{i1}}^i} \tilde{D}_1(b_{k,w_{i1}}^i) \\ &= -\frac{q^{b_1}(b_i, z_i)}{b_{k,w_{i1}}^i} (n_{w_{i1}}^i q(z_{i'} = k) + \beta - b_{k,w_{i1}}^i). \end{aligned} \quad (\text{A} \cdot 11)$$

It can be confirmed that

$$\mathbb{E}_{i'}[\frac{\partial}{\partial b_{k,w_{i1}}^i} \tilde{D}_1(b_{k,w_{i1}}^i)] = \frac{\partial}{\partial b_{k,w_{i1}}^i} D_1(b_{k,w_{i1}}^i). \quad (\text{A} \cdot 12)$$

The difference between the stochastic gradient and the real gradient is

$$\begin{aligned} & \frac{\partial}{\partial b_{k,w_{i1}}^i} \tilde{D}_1(b_{k,w_{i1}}^i) - \frac{\partial}{\partial b_{k,w_{i1}}^i} D_1(b_{k,w_{i1}}^i) \\ &= \frac{q^{b_1}(b_i, z_i)}{b_{k,w_{i1}}^i} (n_{w_{i1}}^i q(z_{i'} = k) - \mathbb{E}[n_{w_{i1}|k}^i]) \end{aligned}$$

$$\begin{aligned} &= \frac{q^{b_1}(b_i, z_i)}{b_{k,w_{i1}}^i} \xi_{i',k} \\ &= \frac{a_k^i b_{k,w_{i2}}^i}{b_{k,w_{i1}}^i c_k^i (c_k^i + 1)} \xi_{i',k}, \end{aligned} \quad (\text{A} \cdot 13)$$

thus there exists a constant C so that

$$\xi_{i',k} \leq C \left\{ \frac{\partial}{\partial b_{k,w_{i1}}^i} \tilde{D}_1(b_{k,w_{i1}}^i) - \frac{\partial}{\partial b_{k,w_{i1}}^i} D_1(b_{k,w_{i1}}^i) \right\}. \quad (\text{A} \cdot 14)$$

Thus

$$\begin{aligned} & \mathbb{E}[\|\xi^{(t)}\|^2|\mathcal{F}^{(t)}] \\ & \leq C^2 \mathbb{E}[\|\nabla \tilde{D}_1(b_{k,w_{i1}}^i) - \nabla D_1(b_{k,w_{i1}}^i)\|^2|\mathcal{F}^{(t)}] \\ & \leq C^2 \mathbb{E}[\|\nabla \tilde{D}_1(b_{k,w_{i1}}^i)\|^2|\mathcal{F}^{(t)}]. \end{aligned} \quad (\text{A} \cdot 15)$$

We then introduce $D_1^*(b_{k,w_{i1}}^i) \geq \tilde{D}_1(b_{k,w_{i1}}^i)$ which is given by

$$\frac{\partial}{\partial b_{k,w_{i1}}^i} D_1^*(b_{k,w_{i1}}^i) = \frac{q^{b_1}(b_i, z_i)}{b_{k,w_{i1}}^i} (n_{w_{i1}}^i + \beta - b_{k,w_{i1}}^i). \quad (\text{A} \cdot 16)$$

Therefore, there exists another constant G such that

$$n_{w_{i1}}^i + \beta - b_{k,w_{i1}}^i \leq G(\mathbb{E}[n_{w_{i1}|k}^i] + \beta - b_{k,w_{i1}}^i), \quad (\text{A} \cdot 17)$$

such as

$$\|\nabla D_1^*(b_{k,w_{i1}}^i)\|^2 \leq G^2 \|\nabla D_1(b_{k,w_{i1}}^i)\|^2. \quad (\text{A} \cdot 18)$$

Therefore, we confirm

$$\begin{aligned} \mathbb{E}[\xi^{(t)}|\mathcal{F}^{(t)}] & \leq C^2 \mathbb{E}[\|\nabla \tilde{D}_1(b_{k,w_{i1}}^i)\|^2|\mathcal{F}^{(t)}] \\ & \leq C^2 G^2 \|\nabla D_1(b_{k,w_{i1}|k}^i)\|^2, \end{aligned} \quad (\text{A} \cdot 19)$$

which satisfies the second condition. $\square$

Zhenghang Cui was born in Shenyang, Liaoning, China, in 1993. He received the B.S. degree from the University of Tokyo, Tokyo, Japan in 2017. He is currently working towards the master's degree in the Department of Computer Science of the university. His research interests are Bayesian theory and machine learning.

Issei Sato received the Ph.D. degree from the University of Tokyo, Tokyo, Japan in 2011. He was an assistant professor at the university. Since 2015, he has been a lecturer at the university. His research interests are medical image analysis, Bayesian optimization, nonparametric Bayes and topic modeling.

Masashi Sugiyama was born in Osaka, Japan, in 1974. He received the degrees of Bachelor of Engineering, Master of Engineering, and Doctor of Engineering in Computer Science from Tokyo Institute of Technology, Japan in 1997, 1999, and 2001, respectively. In 2001, he was appointed Assistant Professor in the same institute, and he was promoted to Associate Professor in 2003. He moved to the University of Tokyo as Professor in 2014. He received an Alexander von Humboldt Foundation

Research Fellowship and researched at Fraunhofer Institute, Berlin, Germany, from 2003 to 2004. In 2006, he received an European Commission Program Erasmus Mundus Scholarship and researched at the University of Edinburgh, Edinburgh, UK. He received the Faculty Award from IBM in 2007 for his contribution to machine learning under non-stationarity, the Nagao Special Researcher Award from the Information Processing Society of Japan in 2011 and the Young Scientists' Prize from the Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science and Technology Japan for his contribution to the density-ratio paradigm of machine learning. His research interests include theories and algorithms of machine learning and data mining, and a wide range of applications such as signal processing, image processing, and robot control.