Fast sampling methods for Bayesian max-margin models

Wenbo Hu, Jun Zhu*, Bo Zhang

Dept. of Comp. Sci. & Tech., State Key Lab of Intell. Tech. & Sys., TNList Lab, Center for Bio-Inspired Computing Research, Tsinghua University, Beijing, 100084, China

ABSTRACT

Bayesian max-margin models have shown superiority in various practical applications, such as text categorization, collaborative prediction, social network link prediction and crowdsourcing, and they conjoin the flexibility of Bayesian modeling and predictive strengths of max-margin learning. However, Monte Carlo sampling for these models still remains challenging, especially for applications that involve large-scale datasets. In this paper, we present the stochastic subgradient Hamiltonian Monte Carlo (HMC) methods, which are easy to implement and computationally efficient. We show the approximate detailed balance property of subgradient HMC which reveals a natural and validated generalization of the ordinary HMC. Furthermore, we investigate the variants that use stochastic subsampling and thermostats for better scalability and mixing. Using stochastic subgradient Markov Chain Monte Carlo (MCMC), we efficiently solve the posterior inference task of various Bayesian max-margin models and extensive experimental results demonstrate the effectiveness of our approach.

1. Introduction

Bayesian max-margin (BMM) models have been shown to be very effective in many real-world applications, such as text analysis (Zhu, Ahmed, & Xing, 2012), collaborative prediction (Xu, Zhu, & Zhang, 2012), social network link prediction (Zhu, 2012) and crowdsourcing (Tian & Zhu, 2015). Such BMM models conjoin the advantages of the discriminative max-margin learning and flexible Bayesian models, and they achieve the best of the both worlds: obtaining the flexibility from a Bayesian model and meanwhile doing discriminative max-margin learning, through a newly-developed unified Bayesian inference framework, regularized Bayesian inference (RegBayes) (Zhu, Chen, & Xing, 2014).

In order to deal with large-scale datasets, developing effective and scalable inference methods is a crucial problem for Bayesian max-margin models, which is becoming a norm in many application areas. Previous variational-approximation-based inference methods are raised to solve the BMM models with mean-field assumptions on posterior distributions (Zhu et al., 2012). When the BMM models use nonparametric Bayesian priors, such variational methods need to adopt the model truncation to finish the variational approximation (Xu, Zhu, & Zhang, 2013; Zhu, Chen, & Xing, 2011). Moreover, in such inference scheme, solving support vec-

tor machine (SVM) subproblems is time-consuming, which motivated the further developments of the Gibbs classifier formulation and the data augmentation-based Gibbs sampler (Xu et al., 2013; Zhang, Zhu, & Zhang, 2014; Zhu, Chen, Perkins, & Xing, 2014).

In Bayesian inference, if we use a conjugate prior (w.r.t a given likelihood), we can easily derive the close-form posterior (Gelman, Carlin, Stern, & Rubin, 2014). However, the BMM models are usually non-conjugate due to the non-smoothness of the hinge loss, which is often involved in an unnormalized pseudo-likelihood. The straightforward Gibbs sampler is not applicable due to the non-conjugacy. With a newly discovered data augmentation technique (Polson & Scott, 2011), the augmented Gibbs sampler achieves accurate posterior sampling and is truncation-free for nonparametric BMM models (Xu et al., 2013; Zhang et al., 2014). However, the Gibbs samplers with data augmentation are not efficient either in high-dimensional spaces as they often involve inverting large matrices (Polson & Scott, 2011). Moreover, the benefit of introducing extra variables would be counteracted in the view of the extra computation on dealing with the extra sampling variables (Roberts & Stramer, 2002).

In this paper, we present the subgradient-based Hamiltonian Monte Carlo (HMC) methods for BMM models, which directly draw samples from the original posterior instead of the augmented one. After adopting some mild conditions of the posterior functions, we show the approximate detailed balance property for subgradient HMC methods. Then using stochastic subgradient estimation (Robbins & Monro, 1951; Welling & Teh, 2011), we further

* Corresponding author.
E-mail addresses: hwb13@mails.tsinghua.edu.cn (W. Hu), dcszj@mail.tsinghua.edu.cn (J. Zhu), dcszb@mail.tsinghua.edu.cn (B. Zhang).

develop the stochastic subgradient MCMC for fast computation. By annealing the discretization stepsizes properly, our stochastic subgradient MCMC methods approximately converge to the target posteriors of basic Bayesian SVM fairly efficiently. To apply stochastic subgradient MCMC on two different types of BMM models with latent variables, we design two different inference algorithms for latent structure discovery, including a nonparametric Bayesian model. Our stochastic subgradient MCMC can achieve dramatically fast sampling and meanwhile draw accurate posterior samples. We carry out extensive empirical studies on large-scale applications to show the effectiveness and scalability of the presented stochastic subgradient MCMC methods for BMM models.

We note that there have been several previous attempts of using subgradient information in HMC or Langevin Monte Carlo (Neal, 2012; Welling & Teh, 2011), yet our work stands as a first close investigation, in which we give the theoretical guarantee and carry out systematic studies on the stochastic subgradient MCMC for Bayesian max-margin learning.

2. Preliminaries

We first briefly review the Bayesian max-margin models with Gibbs classifiers. Then, we introduce the background knowledge of the inference methods, including Hamiltonian Monte Carlo (HMC) and its extension, as well as stochastic gradient Hamiltonian Monte Carlo.

2.1. Bayesian max-margin models

With the generic framework of *RegBayes* (Zhu, Chen, & Xing, 2014), we can design more flexible Bayesian models by adding proper regularization on the target posterior. Namely, after adding posterior regularization to a functional-optimization-reformulated Bayesian model, a *RegBayes* model generally solves the following problem,

$$\inf_{q(\mathcal{M})\in\mathcal{P}} \mathrm{KL}(q(\mathcal{M})||\pi(\mathcal{M})) - \mathbb{E}_q[\log p(\mathcal{D}|\mathcal{M})] + c \cdot \mathcal{R}(q), \quad (1)$$

where $\mathcal{M}$ denotes the model (parameters); $\mathcal{P}$ is the feasible space of probability distributions $q(\mathcal{M})$; $\mathrm{KL}(q(\cdot)||\pi(\cdot))$ is the KL divergence from the target posterior $q(\mathcal{M})$ to the prior $\pi(\mathcal{M})$; $\mathcal{D}$ is the observation dataset; c is a nonnegative regularization parameter and $\mathcal{R}(q)$ is a well-designed regularization term on q. It is not hard to show that if c equals to 0, the solution of problem (1) is the Bayes posterior $q(\mathcal{M}) \propto \pi(\mathcal{M})p(\mathcal{D}|\mathcal{M})$. If c is not zero, we have an extra dimension of freedom to introduce side information into the inference procedure through the posterior regularization term $\mathcal{R}(q)$. For example, when the regularization $\mathcal{R}$ is defined as a hinge loss in supervised learning tasks, such *Regbayes* models turn out to be Bayesian max-margin models and they successfully incorporate the flexibility of Bayesian models and the max-margin classifiers. This strategy has demonstrated promising performance in various tasks, including text classification and topic extraction (Zhu et al., 2012), social network analysis (Zhu, 2012), and matrix factorization (Xu et al., 2012).

In this paper, we consider two examples of Bayesian max-margin models with latent variables, including *max-margin topic model* (MedLDA) (Zhu et al., 2012) and *infinite SVM* (iSVM) (Zhu et al., 2011). But our methods can be applied to other BMM models. Specifically, MedLDA uses a topic model to find the latent topic representations of the documents and uses a max-margin classifier to do document classification. *Infinite SVM* generally uses a Bayesian nonparametric Dirichlet process prior to describe data multi-modality and meanwhile uses max-margin classifiers to do discriminative tasks. More details of these two examples will be provided along the development of the proposed fast samplers for them.

2.2. BMM models with a Gibbs classifier

In the supervised learning setting, there are generally two types of classifiers that can be used with a Bayesian model to define a BMM model, namely, expected classifiers and Gibbs classifiers. In this part, we give the introduction of the two formulations and analyze the merits of choosing Gibbs classifiers.

Let $\mathcal{D} = \{(x_d, y_d)\}_{d=1}^D$ be a given training set. For each data point $(x_d, y_d) \in \mathcal{D}$, x_d denotes the input features and y_d is the corresponding label, which can be binary or multi-valued. To build a classifier, a Bayesian max-margin model can either use the input features or learn a set of latent features. We use x'_d to denote the features that are fit into a classifier. We consider the linear classifier parameterized by η. Then if the labels are binary, the prediction rule is defined as

$$\hat{y}_d = \mathrm{sgn}\big[f(\eta, x'_d)\big], \quad f(\eta, x'_d) = \eta^\top x'_d, \quad (2)$$

where $\mathrm{sgn}(\cdot)$ is the sign function.

For the above setting, an *expected classifier* learns a posterior distribution $q(\eta)$ in a hypothesis space of classifiers that the q-weighted classifier $\hat{y}_d = \mathrm{sgn}\big(\mathbb{E}_q[f(\eta, x'_d)]\big)$ will have the smallest possible risk, which is typically approximated by the training error $\mathcal{R}_\mathcal{D}(q) = \sum_{d=1}^D \mathbb{I}(\hat{y}_d \neq y_d)$, where $\mathbb{I}(\cdot)$ is an indicator function that equals to 1 if predicate holds otherwise 0. We define that $L(y_d, \mathbb{E}_q[f(\eta, x'_d)]) = \max(0, l - y_d\mathbb{E}_q[f(\eta, x'_d)])$ is the hinge loss function with regard to data point d and $l(\geq 1)$ is the cost of making a wrong prediction. Then, we can use the *RegBayes* formulation (Eq. (1)) to define a BMM model with an expected classifier by choosing the loss term $\mathcal{R} = \sum_{d=1}^D L(y_d, \mathbb{E}_q[f(\eta, x'_d)])$. It is known that the hinge loss $\mathcal{R}$ upper bounds the training error $\mathcal{R}_\mathcal{D}$.

Alternatively, the *Gibbs classifier* draws a classifier η according to $q(\eta)$ and uses it to do classification, which is proven to have nice generalization performance (Germain, Lacasse, Laviolette, & Marchand, 2009; McAllester, 2003). In the Gibbs classifier, the corresponding loss is the *expected hinge loss*,

$$\mathcal{R}' = \sum_{d=1}^D \mathbb{E}_q[L(y_d, f(\eta, x'_d))]. \quad (3)$$

Since the hinge loss function L is convex, we can show that $\mathcal{R}'$ is an upper bound of $\mathcal{R}$, using Jensen's inequality:

$$\mathbb{E}_q[L(y_d, f(\eta, x'_d))] \geq L(y_d, \mathbb{E}_q[f(\eta, x'_d)]). \quad (4)$$

Then, the *expected hinge loss* $\mathcal{R}'$ is also the upper bound of the expected training error of the Gibbs classifier $\mathcal{R}'(q) \geq \sum_d \mathbb{E}_q[\mathbb{I}(y_d \neq \hat{y}_d)]$. Therefore, the *Gibbs classifier* formulation gives a more relaxed model while at the same time can obtain uncertainty because we draw a single model for each time. In addition, with Gibbs classifiers, truncation-free sampling can be performed for BMM models with Bayesian nonparametric priors, which is more accurate than variational approximation. The BMM models with *Gibbs classifiers* are already shown to have better performance of both classification results and efficiency of the inference algorithms (Xu et al., 2013; Zhang et al., 2014; Zhu, Chen, Perkins, et al., 2014).

2.3. Hamiltonian Monte Carlo

One popular MCMC inference method is Hamiltonian Monte Carlo (HMC), also known as Hybrid Monte Carlo (Neal, 2012). Hamiltonian Monte Carlo is built on the molecular dynamics and the advantage of HMC over random walk Metropolis and Gibbs sampling is proposing a distant move with a high acceptance probability. More recently, the stochastic extensions of HMC are developed for fast sampling.

Formally, we are interested in the posterior distribution $p(\theta|\mathcal{D}) \propto \exp(-U(\theta; \mathcal{D}))$, where θ denotes the variables of inter-

est and U is the potential energy function in the Hamiltonian dynamics (Arnold, 1978). Consider the general case where a posterior distribution jointly takes into account the prior belief and data. The energy function is written as

$$U(\theta; \mathcal{D}) = -\log p_0(\theta) - \log p(\mathcal{D}|\theta), \tag{5}$$

where $p_0(\theta)$ is the prior and $p(\mathcal{D}|\theta) = \prod_d p(x_d|\theta)$ is the likelihood given the common i.i.d assumption.[1] After introducing auxiliary momentum variables r and its symmetric positive-definite mass M, the HMC sampler simulates the joint distribution: $p(\theta, r) \propto \exp\left(-U(\theta; \mathcal{D}) - r^\top M^{-1} r/2\right)$.

Assuming a differentiable potential energy $U(\theta)$, we can use an HMC sampler to infer the posterior distribution via simulating the dynamics with some discretization integrators such as the Euler or leapfrog. Specifically, using the conventional leapfrog integrator with stepsize h, the HMC method performs the following steps:

$$\begin{cases} r_{t+\frac{1}{2}} = r_t - \dfrac{h}{2} \nabla_\theta U(\theta_t|\mathcal{D}) \\ \theta_{t+1} = \theta_t + hM^{-1} r_{t+\frac{1}{2}} \\ r_{t+1} = r_{t+\frac{1}{2}} - \dfrac{h}{2} \nabla_\theta U(\theta_{t+1}|\mathcal{D}), \end{cases} \tag{6}$$

where r_0 is initialized as $r_0 \sim \mathcal{N}(0, M)$. Having obtained samples of (θ, r), we discard the momentum variable r and get samples of θ from our target posterior.

In particular, if only one leapfrog step is used and M is set to be the identity matrix, we can obtain Langevin Monte Carlo (LMC), a special case of HMC (Neal, 2012).

To compensate for the discretization error, a Metropolis-Hastings correction step is employed to retain the invariance of the target distribution.

2.4. Stochastic gradient HMC

One challenge of the gradient-based HMC methods on dealing with massive data is the expensive evaluation of the posterior gradient $\nabla_\theta U(\theta; \mathcal{D})$. To save time, an unbiased noisy gradient estimate $\nabla_\theta \tilde{U}(\theta; \mathcal{D})$ can be constructed by subsampling the whole dataset, as in stochastic optimization (Bottou, 2010; Robbins & Monro, 1951).

This idea was first proposed in Welling and Teh (2011) to develop the stochastic gradient Langevin dynamics (SGLD), and was later extended by Chen, Fox, and Guestrin (2014) for stochastic gradient HMC with friction and by Ding et al. (2014) for stochastic gradient HMC with thermostats. In these stochastic MCMC methods, the gradient of the log-posterior is estimated as

$$\nabla_\theta \tilde{U}(\theta; \mathcal{D}) = \frac{|\mathcal{D}|}{|\tilde{\mathcal{D}}|} \nabla_\theta U(\theta; \tilde{\mathcal{D}}), \tag{7}$$

where $\tilde{\mathcal{D}}$ is a randomly-drawn subset of $\mathcal{D}$. Since $|\tilde{\mathcal{D}}| \ll |\mathcal{D}|$, computing this noisy gradient estimate turns out much cheaper, hence rendering the overall algorithm scalable.

We now briefly review the stochastic gradient HMC with thermostats, or stochastic gradient Nosé–Hoover thermostat (SGNHT) (Ding et al., 2014). SGNHT uses the simple Euler integrator and introduces a thermostat variable ξ to control the momentum fluctuations as well as the injected noise. The dynamics is simulated as:

$$\begin{cases} r_{t+1} = r_t - h\xi_t r_t - h\nabla_\theta \tilde{U}(\theta_t|\mathcal{D}) + \sqrt{2A}\mathcal{N}(0, h) \\ \theta_{t+1} = \theta_t + hr_{t+1} \\ \xi_{t+1} = \xi_t + h\left(\dfrac{1}{n} r_{t+1}^\top r_{t+1} - 1\right), \end{cases} \tag{8}$$

where A is the diffusion factor parameter and n is the dimension of θ and r. r_0 is initialized from the standard normal distribution $\mathcal{N}(0, \mathbf{I})$ and ξ_0 is initialized as A.

Such stochastic gradient MCMC methods are shown to have a weak posterior-mean convergence instead of a strong sample-wise convergence (Chen, Ding, & Carin, 2015; Sato & Nakagawa, 2014). Such weak convergence is sufficient in many real-world applications.

3. Stochastic subgradient MCMC

One central part in all the above HMC methods is the (stochastic) gradient of the log-posterior. However, such a gradient might not always be available. In this section, we investigate a more general subgradient-based HMC method, analyze its theoretical properties, and use it for the fast inference of Bayesian linear SVMs.

3.1. Subgradient HMC and its approximate detailed balance

When the log-posterior is non-differentiable, gradient-based HMC is not applicable. Using the more general subgradients could potentially address this problem, in analogy to the subgradient descent methods in deterministic optimization (Shor, Kiwiel, & Ruszcayski, 1985).

By plugging the posterior subgradient $\partial_\theta U(\theta_t|\mathcal{D})$ in the ordinary HMC, we come up with the subgradient HMC with a leapfrog method as:

$$\begin{cases} r_{t+1/2} = r_t - \dfrac{h}{2} \partial_\theta U(\theta_t|\mathcal{D}) \\ \theta_{t+1} = \theta_t + hM^{-1} r_{t+1/2} \\ r_{t+1} = r_{t+1/2} - \dfrac{h}{2} \partial_\theta U(\theta_{t+1}|\mathcal{D}), \end{cases} \tag{9}$$

where r_0 is initialized as $r_0 \sim \mathcal{N}(0, M)$ and h is the discretization stepsize.

From a theoretical perspective, we may not be able to readily analyze the volume preservation property of the Hamiltonian dynamics with a non-differentiable potential energy nor the detailed balance of a general subgradient HMC sampler. Instead, we give an approximated theoretical analysis based on several practical assumptions of the potential energy.

In practical Bayesian models, the non-smoothness of the posterior often lies in the hinge loss induced likelihoods which are mainly considered in this paper. These posteriors are continuous everywhere and piece-wise smooth with only a finite number of non-smooth points. The sampler will hit those non-differentiable states with probability zero. Under such practical assumptions, we show the following *approximate detailed balance* property, which claims that the subgradient HMC satisfies the detailed balance property with a polynomial smooth of the potential energy.

We first give a polynomial smooth of the potential energy U_0. The continuous and piece-wise differentiable posterior U_0 is non-smooth on a finite set $S = \{s_i\}_{i=1}^m$ and then the ϵ-neighborhoods around all s_i are defined as $B(s_i, \epsilon) = \{\theta | \|\theta - s_i\| < \epsilon\}$, $i = 1, 2, \cdots, m$. By setting ϵ small enough, the ϵ-neighborhoods can be mutually disjoint: $B(s_i, \epsilon) \cap B(s_j, \epsilon) = \varnothing$, $\forall s_i, s_j \in S, i \neq j$. Using such mutually disjoint neighborhoods, U_ϵ will be constructed as

$$U_\epsilon(\theta) = \begin{cases} U_0(\theta), & \forall s_i \in S, \ \theta \notin B(s_i, \epsilon) \\ \mathcal{P}_{i,\epsilon}(\theta), & \theta \in B(s_i, \epsilon), \end{cases} \tag{10}$$

where $\mathcal{P}_{i,\epsilon}$ is a multi-dimensional Hermite's interpolating polynomial (Bajaj, 1993) satisfying

$$\begin{cases} \mathcal{P}_{i,\epsilon}(s_i \pm \epsilon) = U_0(s_i \pm \epsilon), \\ \nabla_q \mathcal{P}_{i,\epsilon}(s_i \pm \epsilon) = \partial_q U_0(s_i \pm \epsilon). \end{cases} \tag{11}$$
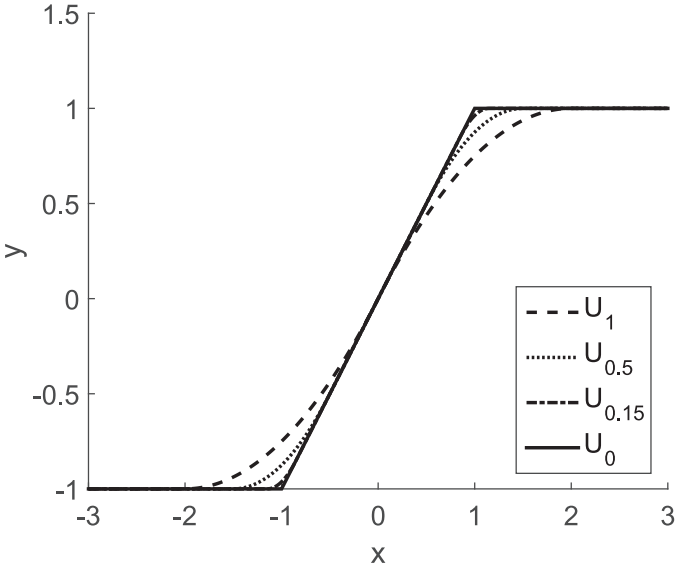

Fig. 1. Illustration of Polynomial Smooth Construction.

According to the definition of U_ϵ (Eqs. (10) and (11)), we can see that U_ϵ is smooth everywhere. Moreover, when $\theta \notin B(s_i, \epsilon), \forall s_i \in S$, we have

$$U_\epsilon(\theta) = U_0(\theta), \quad \nabla_\theta U_\epsilon(\theta) = \partial_\theta U_0(\theta). \tag{12}$$

When ϵ is small enough, the posterior subgradients $\partial_\theta U_0$ used in Eq. (9) is approximately the same with $\nabla_\theta U_\epsilon$ and it will be scarcely possible for the sampler to hit those neighborhoods since the measure of the neighborhood is bounded by ϵ. Then subgradient HMC can be equivalent to drawing samples from a smooth posterior U_ϵ instead. With this approximation, the subgradient HMC satisfies detailed balance and is thus valid for generating approximate samples from the true posterior U_0.

We give an intuitive illustration of the theoretical analysis. In Fig. 1, we construct several polynomial smooth functions U_ϵ for a continuous but non-smooth function U_0. As can be seen, when ϵ is as small as 0.15, $U_{0.15}$ is very close to U_0 and it's very unlikely for a sampler to use finite samples (such as 100 samples), to hit the two neighborhoods $B(-1, 0.15)$ and $B(1, 0.15)$.

3.2. Stochastic subgradient MCMC in practice

We can obtain the version of stochastic subgradient Langevin dynamics (SSGLD) by replacing the gradient of the log-posterior with its subgradient. More formally, SSGLD generates samples by simulating the following dynamics:

$$\begin{cases} \theta_{t+1} = \theta_t - \frac{h^2}{2} \partial_\theta \tilde{U}(\theta_{t+1}|\mathcal{D}) + h\nu_t \\ \nu_t \sim \mathcal{N}(0, I), \end{cases} \tag{13}$$

where $\partial_\theta \tilde{U}(\theta; \mathcal{D}) \triangleq -\partial_\theta \log p(\theta) - \frac{|\mathcal{D}|}{|\tilde{\mathcal{D}}|} \partial_\theta \log p(\tilde{\mathcal{D}}|\theta)$ is the stochastic noisy estimate of the subgradient $\partial_\theta U(\theta; \mathcal{D})$.

In existing SGLD methods (Welling & Teh, 2011), it is recommended to use a polynomial decaying stepsize to save the MH correction step of the Langevin proposals. When the stepsize properly decays, the Markov chain would gradually converge to the target posterior. One subtle part of the method is thus on tuning the discretization stepsize. A pre-specified annealing scheme (if not chosen properly) would make the chain either miss or oscillate around the target. More recent work Teh, Thiéry, and Vollmer (2014) recommends some relatively optimal scheme for SGLD. Inspired by adaptive stepsizes for (sub)gradient descent (Aga-Grad) methods (Duchi, Hazan, & Singer, 2011), we, in this paper,

adopt the same adaptive stepsize setting for our SSGLD methods (Li, Chen, Carlson, & Carin, 2016). As we shall see in the experiments, such a scheme is beneficial to yield faster mixing speeds.

We can derive stochastic subgradient Hamiltonian Monte Carlo likewise. We adopt an improved version of stochastic gradient HMC (Ding et al., 2014) to derive our stochastic subgradient Nosé–Hoover thermostat (SSGNHT), which generates samples via the following iterations:

$$\begin{cases} r_{t+1} = r_t - \xi_t r_t h - h\partial_\theta \tilde{U}(\theta_t|\mathcal{D}) + \sqrt{2A}\mathcal{N}(0, h) \\ \theta_{t+1} = \theta_t + hr_{t+1} \\ \xi_{t+1} = \xi_t + h\left(\frac{1}{n}r_t^\top r_t - 1\right). \end{cases} \tag{14}$$

Again we omit the MH correction step and the SSGNHT simulations would generate posterior samples more efficiently with the properly decaying stepsizes and thermostat initialization.

3.3. Stochastic subgradient MCMC for Bayesian linear SVMs

The stochastic subgradient MCMC can be used for fast sampling of Bayesian linear SVM. Let $\mathcal{D} = \{(x_d, y_d)\}_{d=1}^D$ be the given training dataset, where x_d is the n-dimensional feature vector of the dth instance and $y_d \in \{-1, +1\}$ is the binary label. We use linear classifiers with a weight vector $\eta \in \mathbb{R}^n$ and the decision rule is naturally $\hat{y} = \text{sgn}(\eta^\top x)$. Then for a Bayesian linear SVM model, we are interested in learning the posterior distribution $p(\eta|\mathcal{D}) \propto p_0(\eta) \prod_d \psi(y_d|x_d, \eta)$. The prior is commonly set as a standard normal distribution $p_0(\eta) = \mathcal{N}(0, I)$, and the per-datum unnormalized likelihood is $\psi(y_d|x_d, \eta) = \exp(-c \cdot \max(0, l - y_d\eta^\top x_d))$. Then, the subgradient of the log-posterior involves evaluating the subgradient of the non-differentiable log-likelihood

$$\partial_\eta \log \psi(y_d|x_d, \eta) = \begin{cases} -cy_d x_d & l - y_d\eta^\top x_d > 0 \\ 0 & l - y_d\eta^\top x_d \le 0. \end{cases} \tag{15}$$

With this subgradient, we can use the stochastic subgradient MCMC method to do fast sampling for the Bayesian linear SVM model.

4. Fast sampling for Bayesian max-margin models with latent variables

We now show how to leverage the above stochastic subgradient MCMC methods to derive fast sampling algorithms for Bayesian max-margin models with latent variables. We develop algorithms for two different BMM models with latent variables.

4.1. Fast sampling for max-margin topic models

For parametric BMM models, whose model parameter number is fixed, we just calculate the (stochastic) log-posterior subgradient and run our stochastic subgradient MCMC method. In this part, we use Gibbs MedLDA (Zhu, Chen, Perkins, et al., 2014) as an example to show how to do fast sampling for parametric BMM models.

4.1.1. Gibbs MedLDA

As illustrated in Fig. 2, the max-margin topic model has two parts: 1) a latent Dirichlet allocation model for modeling underlying topic structures of the given documents and 2) a max-margin classifier for predicting document labels. The LDA part is a hierarchical Bayesian model which uses an admixture of K topics, $\mathbf{\Phi} = \{\mathbf{\Phi}_k\}_{k=1}^K$, as a latent document representation. Here each topic $\mathbf{\Phi}_k$ is a multinomial distribution over a V-word vocabulary and has the symmetric Dirichlet prior $\text{Dir}(\beta)$. For a single document d, N_d words are generated and the detailed process is

1. draw a topic proportion $\theta_d \sim \text{Dir}(\alpha)$,
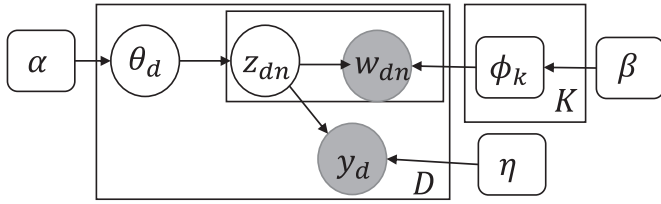
Fig. 2. Graphical model representation of Gibbs MedLDA.

2. for each word $n(1 \leq n \leq N_d)$:
   (a) draw a topic assignment $z_{dn} \sim \text{Multinomial}(\theta_d)$,
   (b) draw the observed word $w_{dn} \sim \text{Multinomial}(\Phi_{z_{dn}})$.

Given a set of documents $\mathbf{W} = \{w_d\}_{d=1}^{D}$, we denote its latent topic proportions as $\mathbf{\Theta} = \{\theta_d\}_{d=1}^{D}$ and its topic assignments as $\mathbf{Z} = \{z_d\}_{d=1}^{D}, z_d = \{z_{dn}\}_{n=1}^{N_d}$. Let $\bar{z}_d$ be the average topic assignments of the words in document d, with element $\bar{z}_{dk} = \frac{1}{N_d} \sum_n^{N_d} \mathbb{I}(z_{dn} = k)$.

We use the Gibbs classifier formulation to build the Gibbs MedLDA model. If we have drawn a sample of the topic assignments $\mathbf{Z}$ and the classifier weights η from the posterior distribution, we can get the prediction of the document label $y_d \in \{1, 2, \cdots, L\}$ as,

$$\hat{y} = \text{argmax}_y f(y, \bar{z}_d | \eta), \quad f(y, \bar{z}_d | \eta) = \eta^\top g(y, \bar{z}_d), \quad (16)$$

where $g(y, \bar{z}_d)$ is a long vector consisting of L subvectors with the yth being $\bar{z}_d$ and all others being zero. The corresponding *expected hinge loss* is

$$\mathcal{R}'(q(\eta, \mathbf{\Theta}, \mathbf{Z}, \mathbf{\Phi}))$$
$$= \sum_{d=1}^{D} \mathbb{E}_q \left[\max\left(0, l + \max_{y \neq y_d} f(y, \bar{z}_d | \eta) - f(y_d, \bar{z}_d | \eta) \right) \right]. \quad (17)$$

Then, Gibbs MedLDA infers the latent topic assignments $\mathbf{Z}$ and the classifier weights η by solving the following *RegBayes* problem:

$$\min_{q(\eta, \mathbf{\Theta}, \mathbf{Z}, \mathbf{\Phi})} \mathcal{L}(q(\eta, \mathbf{\Theta}, \mathbf{Z}, \mathbf{\Phi})) + c \cdot \mathcal{R}'(q(\eta, \mathbf{\Theta}, \mathbf{Z}, \mathbf{\Phi})), \quad (18)$$

where $\mathcal{L} = \text{KL}(q||p_0(\eta, \mathbf{\Theta}, \mathbf{Z}, \mathbf{\Phi})) - \mathbb{E}_q[\log(p(\mathbf{W}|\mathbf{Z}, \mathbf{\Phi})]$ is the reformulated objective when doing standard Bayesian inference.

4.1.2. Fast sampling for Gibbs MedLDA

Instead of sampling in the whole space, which may lead to low efficiency (Griffiths & Steyvers, 2004), we collapse out $\mathbf{\Theta}$ and draw samples form the collapsed distribution,

$$p(\mathbf{W}, \mathbf{Z}, \mathbf{\Phi}, y|\alpha, \beta) = p(\eta)p(\mathbf{\Phi}|\beta) \prod_{d=1}^{D} p(\mathbf{w}_d, z_d|\alpha, \mathbf{\Phi}) \psi(y_d|z_d, \eta),$$

where

$$p(\mathbf{w}_d, z_d|\alpha, \mathbf{\Phi}) = \prod_{k=1}^{K} \frac{\Gamma(\alpha + C_{dk.})}{\Gamma(\alpha)} \prod_{w=1}^{W} \Phi_{kw}^{C_{dkw}}. \quad (19)$$

$C_{dk.}$ is the number of words in document d that is assigned to topic k and C_{dkw} is the number of words w in document d that is assigned to topic k. $\psi(y_d|z_d, \boldsymbol{\eta})$ is defined as,

$$\psi(y_d|z_d, \eta)$$
$$= \exp\left[-c \max\left(0, l + \max_{y \neq y_d} \eta^\top g(y, \bar{z}_d) - \eta^\top g(y_d, \bar{z}_d) \right) \right]. \quad (20)$$

For the collapsed posterior of MedLDA, we can sample classifiers η using stochastic subgradient MCMC and sample the topic model parameters $\mathbf{\Phi}$ using the SGRLD method (Patterson & Teh, 2013). With the randomly-drawn document minibatch $\tilde{\mathbf{W}}$, we get

the stochastic subgradient of the log posterior with respect to η as,

$$\begin{cases} \partial_\eta \log \psi = 0; & \text{if } \psi(y_d|z_d, \eta) = 1, \\ \partial_{\eta_{y^*}} \log \psi = -c\bar{z}_d, \ \partial_{\eta_{y_d}} \log \psi = c\bar{z}_d; & \text{if } \psi(y_d|z_d, \eta) < 1, \end{cases} \quad (21)$$

where $y^* = \text{argmax}_{y \neq y} \eta^\top g(y, \bar{z}_d)$. Here, η_y is the yth subvector of η which is corresponding to the non-zero elements of $g(y, \bar{z}_d)$ and in the second case of the calculation, the subgradients with respect to the unmentioned subvectors of η are zero. With the stochastic posterior subgradient with respect to η, we can use stochastic subgradient MCMC to sample η.

We use the expanded-mean formulation for Φ: $\Phi_{kn} = |\pi_{kn}|/(\sum_n |\pi_{kn}|)$ and follow the SGRLD iterations to sample the admixture Φ on the Riemannian manifold (Eq. (10) in Patterson & Teh (2013)).

The stochastic posterior (sub)gradients with respect to Φ and η are calculated given the expectation of $\bar{z}$ (Mimno, Hoffman, & Blei, 2012). To calculate the expectation of $\bar{z}$, the Gibbs sampling iterations for the topic assignments of document d is as follows:

$$p(z_{dn} = k|z_{d,-n}, \Phi, \eta) \propto (\alpha + C_{dk.}^{-n})\Phi_{kn}\psi(y_d|\bar{z}_d^*, \eta), \quad (22)$$

where $z_{d,-n}$ is the topic assignments of other documents, $\bar{z}_d^*$ is the average topic assignments $\bar{z}_d$ after setting topic z_{dn} as k and $C_{dk.}^{-n}$ is the number of words assignment as topic k in document d after removing word n. With the learned topic admixture Φ and classifier weights η, we randomly draw a sample of Φ and η and make predictions as described in Zhu, Chen, Perkins, et al. (2014). The overall stochastic sampler for Gibbs MedLDA is concluded in Algorithm 1.

Algorithm 1 SSGRLD For Gibbs MedLDA.

Input: documents $(w_d, y_d), d = 1, \cdots, D$.
Initialization
repeat
    Draw a stochastic subset $\tilde{\mathcal{D}}$
    Draw topic assignments of the documents in $\tilde{\mathcal{D}}$ using Eq. (22)
    Compute stochastic posterior (sub)gradient with respect to Φ and η
    Run subgradient sampler for η and Φ with the stochastic posterior subgradient
until Converge

4.2. Fast sampling for infinite SVMs

Another important type of Bayesian max-margin models with latent variables uses Bayesian nonparametric priors. Such BMM models are defined on infinite-dimensional spaces and the size of the models will be learned from the data. Typical example of this type is infinite SVM (Zhu et al., 2011) and we use the HMC-within-Gibbs strategy to build fast sampling methods for this type of models.

4.2.1. Gibbs infinite SVM

Real world data often have some latent clustering structures, where mixture-of-experts models are generally capable of capturing these local structures. When each expert is a linear SVM, the resultant mixture of SVMs learns a non-linear model instead of simply a linear one (Collobert, Bengio, & Bengio, 2002; Fu, Robles-Kelly, & Zhou, 2010). Recent work further presents a nonparametric extension, infinite SVM (iSVM) (Zhu et al., 2011) (See Fig. 3), which automatically infers the number of experts. Below, we apply the subgradient-based fast sampling method to infinite SVM.

Given a set of data $\mathcal{D} = \{(x_d, y_d)\}_{d=1}^{D}$, we let z_d denote the component assignment for the datum x_d. Each component is associated
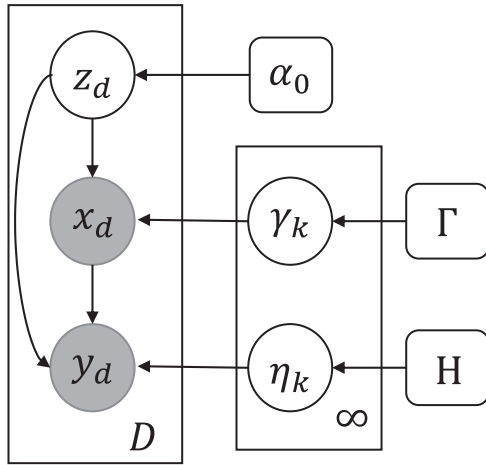
Fig. 3. Graphical model representation of Gibbs iSVM.

with a linear classifier η_{z_d} and a Gaussian likelihood $(\mu_{z_d}, \Sigma_{z_d})$ to describe the input features.[2] All the parameters follow some priors: a standard Gaussian prior for η and a Gaussian-Inverse-Wishart conjugate prior for (μ, Σ). In iSVM, we choose a *Chinese Restaurant Process* (CRP) (Pitman, 2006) prior for Z.

Though alternative approaches exist, we define the expert classifier as a Gibbs classifier to get uncertainty for the assignments Z and the classifier weights η. Namely, given the posterior distribution $q(Z, \eta)$, the Gibbs classifier draws a component assignment z_d and a classifier η_{z_d} for each data point x_d and makes prediction:

$$\hat{y}_d = \underset{y}{\arg\max} f(y, x_d, z_d)$$

$$f(y, x_d, z_d) = \eta_{z_d}^\top g(y, x_d), \ y \in \{1, 2, \cdots, L\}, \tag{23}$$

where $g(y, x_d)$ is a long vector consisting of L subvectors with the yth being x_d and all others being zero. We adopt the expected hinge loss for Gibbs iSVM,

$$\mathcal{R}'(q(\eta, \gamma, Z)) =$$

$$\mathbb{E}_{q(Z, \eta, \gamma)} \left[\sum_{d=1}^{D} \max \left(0, l + \max_{y \neq y_d} f(y, x_d, z_d) - f(y_d, x_d, z_d) \right) \right]. \tag{24}$$

Together with the Gibbs classifier and the expected hinge loss, we can define a *RegBayes* model for the mixture of Gibbs classifiers:

$$\min_{q(Z, \eta, \gamma)} \mathcal{L}(q(Z, \eta, \gamma)) + c \cdot \mathcal{R}'(q(Z, \eta, \gamma)), \tag{25}$$

where $\gamma = (\mu, \Sigma)$ are the mean and variance parameters for each Gaussian component and $\mathcal{L} = \text{KL}(q||p_0(\eta, \gamma, Z) - \mathbb{E}_q[\log(p(X|Z, \gamma)]$ is the objective function when doing standard Bayesian inference. With regard to the *RegBayes* formulation in Eq. (1), the normalized posterior distribution of infinite SVM is

$$q(Z, \eta, \gamma) \propto p_0(\eta, \gamma, Z) p(X|Z, \gamma) \prod_{d=1}^{D} \psi(y_d | x_d, \eta_{z_d}), \tag{26}$$

where $\psi(y_d | x_d, \eta_{z_d}) = \exp(-c \max(0, l + \max_{y \neq y_d} f(y, x_d, z_d) - f(y_d, x_d, z_d)))$. We refer readers to Zhu et al. (2011) and Zhang et al. (2014) for more details.

4.2.2. Fast sampling for Gibbs iSVM

We develop the fast sampling method for Gibbs iSVM by incorporating the stochastic subgradient MCMC method within the loop of a Gibbs sampler. The HMC-within-Gibbs strategy for iSVM is detailed below.

---

[2] The Gaussian likelihood is optional.

For Z: Give η, the conditional distribution is

$$p(Z|\eta) \propto p_0(Z) p(X|Z) \psi(Y|Z, \eta), \tag{27}$$

where $p(X|Z) = \int p_0(\gamma) p(X|Z, \gamma) \mathrm{d}\gamma$ is the marginal distribution via collapsing γ and $p_0(Z)$ is the CRP prior. Let α_0 be the hyperparameter of the CRP prior and $n_{-d,k}$ be the number of data points that belong to component k except d. Given classifiers η and assignments of other data points Z_{-d}, we sample component assignments z_d by normalizing the following two probabilities (existing component k and a new component):

(1) $p(z_d = k|Z_{-d}, \eta) \propto n_{-d,k} \psi(y_d | z_d = k, \eta_k) \cdot p(x_d | Z_{-d}, X_{-d}^k)$
(2) $p(z_d = new | Z_{-d}, \eta) \propto \alpha_0 p(x_d) \int \psi(y_d | \eta') p_0(\eta') \mathrm{d}\eta'$

In case 2), $p(x_d) = \int p(x_d | \gamma) p_0(\gamma) \mathrm{d}\gamma$ is the likelihood of the data d and can be computed in closed-form using the conjugate property. The second integral in case 2) can be approximated by using importance sampling.

For η: Give Z, the number of active cluster is known. We need to efficiently sample the classifier weights η_k of each component k from the following conditional distribution,

$$p(\eta_k | Z) \propto p_0(\eta_k) \prod_{d:z_d=k} \psi(y_d | z_d, \eta_{z_d}), \tag{28}$$

where $p_0(\eta_k)$ is a standard normal prior. With our proposed stochastic subgradient MCMC, the classifiers η can be directly sampled using only a minibatch of whole dataset. Here, we give the stochastic subgradients of the log conditional distribution:

$$\partial_{\eta_k} \log \left[p_0(\eta_k) \prod_{d:z_d=k} \psi(y_d | z_d, \eta_{z_d}) \right] \approx -\eta_k$$

$$+ \frac{|\mathcal{D}|}{|\tilde{\mathcal{D}}|} \sum_{d:z_d=k, (x_d,y_d) \in \tilde{\mathcal{D}}} \partial_\eta \log \psi(y_d | x_d, \eta_{z_d}), \tag{29}$$

where the subgradients of the multi-class hinge loss $\psi(y_d | x_d, \eta_{z_d})$ are similarly defined as Eq. (21). Using this subgradient in the SS-GLD (Eq. (13)) or SSGNHT (Eq. (14)), we can derive the stochastic subgradient inner sampler for classifiers η.

The whole stochastic HMC(LMC)-within-Gibbs algorithm structure is outlined in Algorithm. 2.

Algorithm 2 Stochastic HMC within Gibbs for infinite SVM.

Input: data $(x_d, y_d), d = 1, \cdots, N$, batchsize $\tilde{N}$.
Initialization
repeat
    sample z given η
    sample η given z using stochastic subgradient HMC
until Converge

---

5. Experiments

We now implement our stochastic subgradient MCMC on various Bayesian max-margin models, including the basic Bayesian linear SVM and two sophisticated Bayesian max-margin models with latent variables (GiSVM and Gibbs MedLDA). Our results demonstrate that stochastic subgradient MCMC can achieve great improvement on time efficiency and meanwhile still generating accurate posterior samples.

All experiments are done on a desktop computer with single-core rate up to 3.0 GHz. The stepsize parameter at iteration t decays via $h_t = h_0 * (1 + t/b)^{-\gamma}$. Normally, we set $b = 1$ for SVM classifier η and $b = 100$ for topic-word parameter Φ. We choose h_0 and γ via a grid search. Furthermore, the AdaGrad stepsizes are considered for stochastic subgradient Langevin dynamics method.
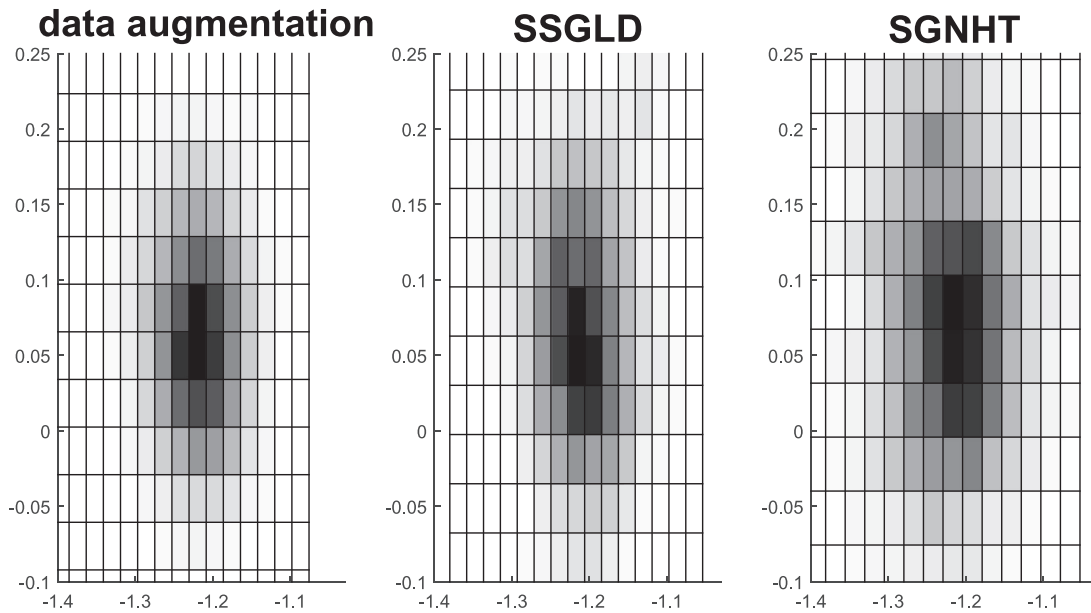
Fig. 4. Visual comparison of posterior samples.

5.1. Bayesian linear SVMs

We first consider the basic Bayesian linear SVM model and compare our stochastic subgradient sampling methods with the Gibbs sampler with data augmentation (Zhu, Chen, Perkins, et al., 2014) and the random walk Metropolis with stochastic MH test (Korattikara, Chen, & Welling, 2014) (stochastic random walk Metropolis, SRWM).

5.1.1. Results on synthetic data

We first test our methods on a 2D synthetic dataset to show that our methods give correct samples from the posterior distribution. Note that we view the results of this experiment as a simple proof of idea and hence choose the more direct visual comparison. We follow the Bayesian linear SVM model defined in Section 3.1 and generate 1000 observations as the synthetic dataset. Specifically, we generate features x from a uniform distribution $x_i \overset{i.i.d}{\sim} U(0,1)$ and the coefficient vector from a normal distribution $\eta \sim \mathcal{N}(0, 1/3 \cdot I)$. Given the features and coefficients, the labels are generated from the Bernoulli distribution with parameter δ, where,

$$\delta = \frac{\psi(y_i = 1 | x_i, \eta)}{\psi(y_i = 1 | x_i, \eta) + \psi(y_i = -1 | x_i, \eta)}.$$

We compare the samples obtained from SSGLD and SSGNHT with those from the data augmentation method which is an accurate sampler for Bayesian SVMs. We take 5,000 samples for each method after a sufficiently long burn-in stage and give the comparison in Fig. 4, where the densities of the obtained samples are shown via the grayscales of the grids. The results suggest that our stochastic subgradient MCMC methods are accurate, although the stochastic subsampling and the neglect of MH test bring some noise. This result is compatible with the previous weak convergence analysis of the ordinary HMC methods (Chen et al., 2015; Sato & Nakagawa, 2014).

5.1.2. Results on real data

We then test two stochastic subgradient MCMC methods, SS-GLD and SSGNHT on the Realsim dataset[3] and the larger UCI Higgs
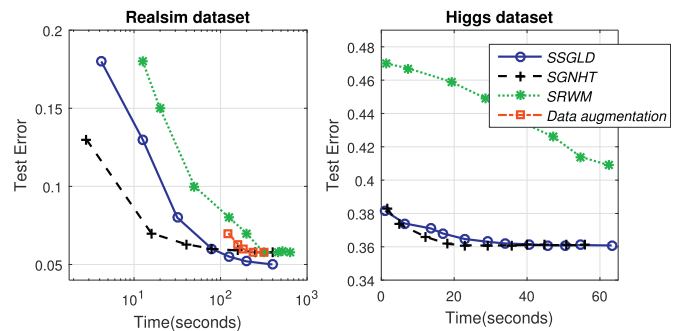

Fig. 5. Experimental results of Bayesian linear SVMs.

dataset (Asuncion & Newman, 2007). The Higgs dataset contains 1.1×10^7 samples in a 28-dimensional feature space. We randomly choose 10^7 samples as the training set and the rest as the testing set.

For the Realsim dataset, we set the stochastic batchsize $|\tilde{\mathcal{D}}| = 10$ for all stochastic inference methods. For Higgs dataset, we set $|\tilde{\mathcal{D}}|$ to be 1,000 for both SSGLD and SRWM and $|\tilde{\mathcal{D}}| = 100$ for SSGHNT. We use tuned polynomial decaying stepsizes for stochastic subgradient MCMC methods and specifically for SSGLD, we prefer adaptive stepsize AdaGrad, which has been successfully applied in the stochastic (sub)gradient descent (Duchi et al., 2011). For SRWM, the variance parameter is set as 0.01. These turn to be a good setting analyzed in the following sensitivity analysis in Section 5.1.3.

The convergence curves of various methods with respect to the running time on both datasets are shown in Fig. 5. We can see that our stochastic subgradient MCMC methods are several magnitudes faster than the baseline methods. Compared with the Gibbs sampling with data augmentation method, stochastic subgradient MCMC methods get much cheaper updates and hence are more scalable. Specially for the larger Higgs dataset, a single update of Gibbs sampling is not finished when the stochastic subgradient MCMC get converged. Furthermore, although both SRWM and stochastic subgradient MCMC use stochastic minibatches, stochastic subgradient MCMC methods mix much faster than SRWM because the posterior subgradient information provides the right direction to the true posterior.
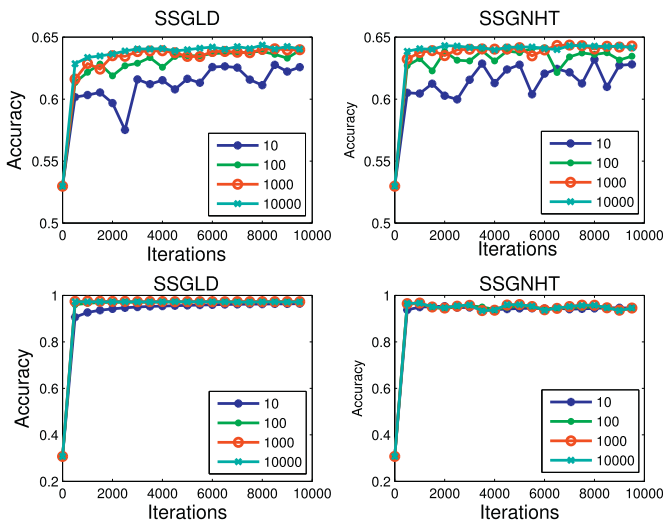
Fig. 6. Sensitivity analysis of the batchsize parameter for both SSGLD and SSGNHT on the Higgs dataset (**first row**); and the Realsim dataset (**second row**).
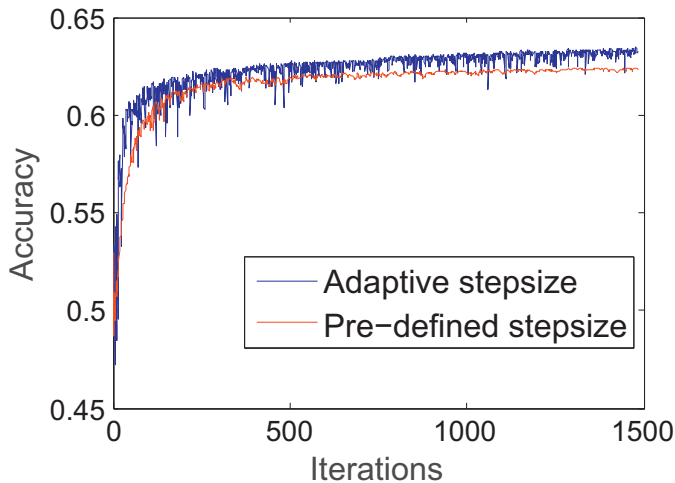

Fig. 7. Performance of SSGLD with AdaGrad.

5.1.3. Sensitivity analysis

Tuning the batchsize $|\tilde{\mathcal{D}}|$ reflects an accuracy-efficiency trade-off, analogous to the bias-variance tradeoff in stochastic Monte Carlo sampling (Korattikara et al., 2014). In general, using a smaller batchsize often leads to a larger injected noise, but the computation cost at each iteration is reduced, which is linear to the batchsize (i.e., $O(|\tilde{\mathcal{D}}|)$. When doing cross validation to select parameters, both accuracy and time efficiency are key factors that should be taken into consideration.

Fig. 6 presents the sensitivity analysis of the batchsize for the two stochastic subgradient MCMC methods on both Higgs and Realsim datasets. The performance of our stochastic subgradient MCMC appears to be fairly promising except for extremely tiny batchsizes.

In our experiments, adaptive stepsizes (AdaGrad) bring a better mixing rate than the polynomial decaying stepsizes. This may result from the flexible stepsize decaying at different dimensions. We also give an empirical analysis in Fig. 7. As can be seen, for the Higgs dataset, adaptive stepsizes bring better classification results than the pre-defined polynomial-decaying stepsizes.
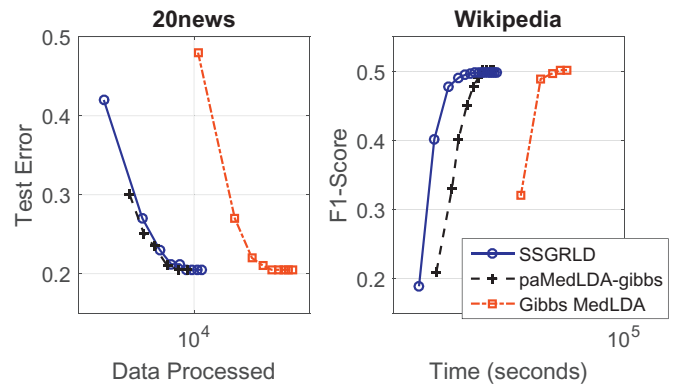

Fig. 8. Empirical results of different methods for MedLDA.

5.2. Gibbs max-margin topic models

Now, we implement the fast sampling for Gibbs MedLDA. We show the efficiency and accuracy of our stochastic subgradient Riemannian Langevin Dynamics (SSGRLD) using the 20news dataset and the larger Wikipedia dataset. Following the dataset setting in Zhu, Chen, Perkins, et al. (2014), the stop words are removed according to a standard list. We compare our SSGRLD with the data augmentation (Gibbs MedLDA) (Zhu, Chen, Perkins, et al., 2014) and its newly developed extension in the online Bayesian passive-aggressive learning framework (paMedLDA-gibbs) (Shi & Zhu, 2014). For the smaller 20news dataset, the involved three methods all use the binary version and then adopt the "one-vs-all" strategy for multi-class classification. For the larger Wikipedia dataset, the SSGRLD method uses the multi-class setting and other two use the multi-task formulation as described in Zhu, Chen, Perkins, et al. (2014) and Shi and Zhu (2014).

5.2.1. Classification performance

We first test on the 20news dataset which consists of 11,269 training documents and 20 categories. We set the hyper-parameters as $\alpha = 1, \beta = 1, c = 1, \ell = 164$ as suggested in Zhu, Chen, Perkins, et al. (2014). Fig. 8(left) shows the number of documents processed in order to reach a specific accuracy score, where topic number is set as 50. As we can see, the two stochastic samplers use much fewer documents and efficiently explore the data redundancy by using a minibatch at each iteration.

Then we test on the larger Wikipedia dataset which consists of 1.1 million training documents and 20 categories. We use the same hyper-parameter setting with the 20news dataset, except for a few settings: $c = 10, \ell = 196$ for SSGRLD and $\ell = 1$ for both Gibbs MedLDA and paMedLDA-gibbs. We set the topic number as 40. Fig. 8 shows the F1-scores as a function of time. It can be seen that SSGRLD produces comparable classification results. As for the efficiency, both SSGRLD and paMedLDA-gibbs are one order of magnitude more efficient than the previous Gibbs MedLDA. This is due to the minibatch training. Meanwhile, although in the same magnitude, SSGRLD is still faster than paMedLDA-gibbs. We argue that this is because SSGRLD does not use augmented variables and directly draws samples from the SVM classifier. Moreover, the matrix inversion involved in the data augmentation technique is costly in the whole procedure.

5.2.2. Topic representations

Finally, we visualize the discovered topic representations of SSGRLD on the 20news dataset. For the all 20 categories, we show
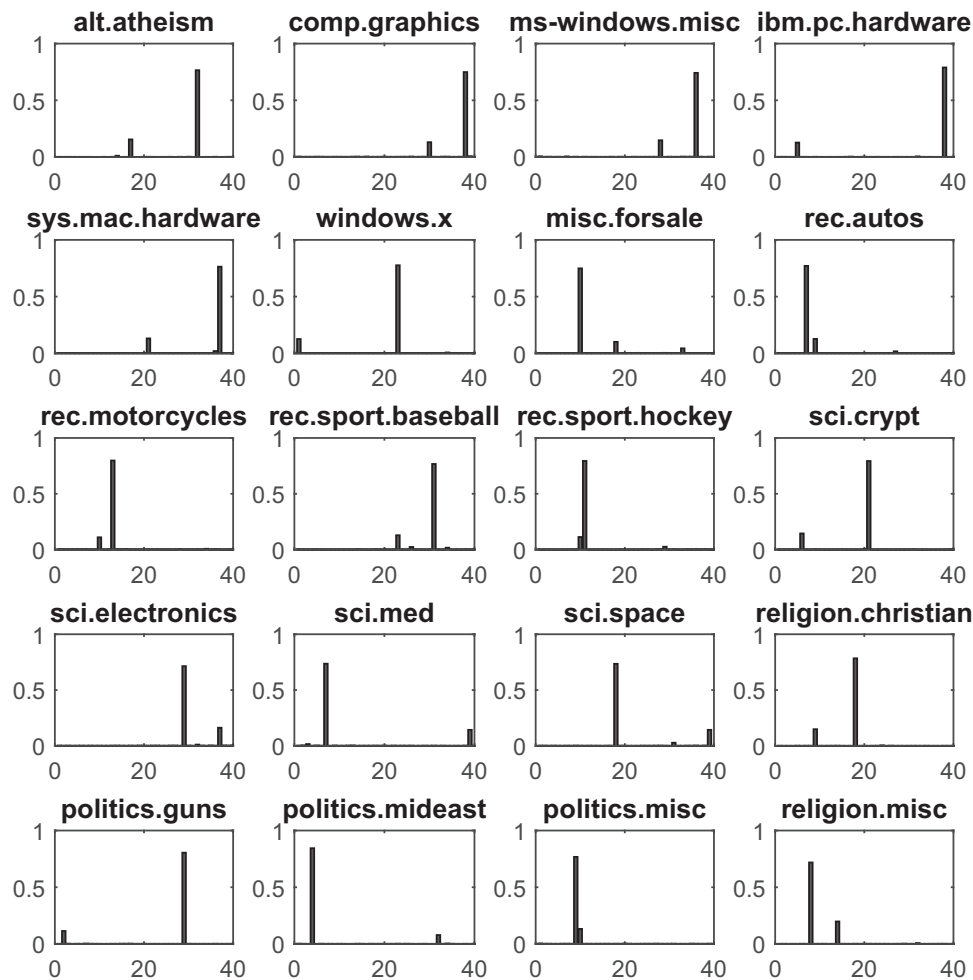
Fig. 9. Visualization of learnt topics by SSGRLD.

Table 1
Representative top words of the salient topic(s).

Category	Top words	Category	Top words
Atheism	God, don, atheism	Graphics	Image, jpeg, file
Windows	Windows, file, card	Pc	Scsi, drive, disk, mb, dos
Mac	Mac, apple, drive	Windows	Window, server, file
Forsale	Anonymity, sphinx	rec.autos	Car, engine, speed
Motocycle	Bike, ride, bmw	Baseball	Team, game, runs
Hockey	Team, nhl, season	Crypt	Key, chip, security, law
Electronics	Power, circuit, wire	Medical	Food, medical, doctor
Space	Nasa, launch, earth	Christian	God, jesus, church, bible
Guns	Gun, weapon, firearm	Mideast	Israel, turkish, jews, arab
Politics	Mr, president, states	Religion	Jesus, bible, christian

the average topic representations of the documents form each category. As we can see in Fig. 9, the average topic distribution for the corresponding classifier is very sparse (only one or two non-zero entries). We also give the most representative top words of the salient topic(s) of each category in Table 1. We can see that the top words of the salient topic(s) are highly related to the category information. For example, the salient topic learned by classifier *sci.space* has the top words as NASA, launch, moon, satellite, etc. These patterns are similar as those in (Shi & Zhu, 2014; Zhu, Chen, Perkins, et al., 2014).

5.3. Infinite SVMs

The proposed subgradient-based sampling methods can also be used for fast inference of infinite SVM (Zhu et al., 2011), a Dirichlet process mixture of large-margin kernel machines.

We choose two datasets, Protein and IJCNN1, to test our methods. The Protein dataset (Zhang et al., 2014) was created for Protein fold classifications and consists of 698 samples and 27 classes with 21 features. The IJCNN1 dataset[4] is originated from an engine system binary classification problem and consists of 49,990 training samples with 22 features.

We implement two inference methods for iSVM including SS-GNHT within Gibbs (Algorithm 2) and Gibbs sampling with data augmentation (Zhang et al., 2014). Other models are also implemented for comparison, such as multinomial logit model (MNL), linear SVM, RBF-SVM and DP mixture of generalized linear models (dpMNL) (Shahbaba & Neal, 2009). We use cross-validations to choose hyper-parameters and get the results in Table 2.

We can see that nonlinear models using a mixture-of-experts, such as GiSVM and dpMNL, are superior in classification. In the stochastic subgradient MCMC, η sampling step can be dramatically accelerated, with comparable or even better prediction performance. This superiority results from both stochastic subsam-

Table 2
Efficiency (in minutes) and accuracy of various models on the Protein and IJCNN1 datasets.

Datasets	Protein			IJCNN1		
	Accu(%)	Time for η	Total Time	Accu(%)	Time for η	Total Time
MNL	50.0	–	0.10	91.3	–	3.21
Linear SVM	50.8	–	0.03	91.0	–	0.56
RBF-SVM	53.1	–	0.11	93.9	–	2.79
dpMNL	**56.3**	–	7.64	94.0	–	7.62
Gibbs-iSVM	55.8 ± 0.0	8.31 ± 0.27	15.15 ± 0.29	$\mathbf{94.2 \pm 0.7}$	9.13 ± 0.95	22.71 ± 1.16
SSGNHT-iSVM	56.1 ± 0.0	$\mathbf{0.17 \pm 0.02}$	$\mathbf{7.32 \pm 0.26}$	$\mathbf{94.2 \pm 0.8}$	$\mathbf{1.17 \pm 0.08}$	$\mathbf{13.84 \pm 1.90}$

pling and avoiding the matrix inversion in the data augmentation technique.

6. Conclusions

We systematically investigate the fast sampling methods for Bayesian max-margin models. We first study a general subgradient HMC sampling method and several stochastic variants including SSGLD and SSGNHT. Theoretical analysis shows the approximated detailed balance of the proposed stochastic subgradient MCMC methods. Then we apply the stochastic subgradient samplers to Bayesian linear SVMs and two sophisticated Bayesian max-margin models with latent variables (GiSVM and Gibbs MedLDA). Extensive empirical studies demonstrate the effectiveness of the stochastic subgradient MCMC methods on improving time efficiency while maintaining a high accuracy of the samples.

The strengths of our methods are 1) fast inference for BMM models compared with the previous Gibbs sampling method with data augmentation; 2) accurate sampling which is as good as the Gibbs sampling with data augmentation and 3) applications to non-conjugate posterior sampling which cannot be simply accomplished. However, when the data sizes of the applications are too large to be processed in a single machine, it is still difficult to use only stochastic subgradient MCMC to solve the problem.

We consider the future work in three categories: algorithm-level, model-level and application-level. For the proposed algorithm itself, the future work includes further scaling up using parallel computation (Ahn, Shahbaba, & Welling, 2014). For the model setting, the future work includes applying our method to other models with continuous but non-smooth posteriors, such as sparse models with Laplacian priors. At the application level, we consider using our method to scale up several Bayesian max-margin models that are used in intelligent systems, such as nonparametric max-margin matrix factorization for collaborative filtering (Xu et al., 2012).

The big data is identified as an important building block of intelligent systems (Lehmann et al., 2015; Manyika et al., 2011) and the fast inference is becoming a central element therein (Park, Park, Jung, & Lee, 2015). For related Bayesian models (Pearl, 2014), big learning with Bayesian models is one of the recent research focuses (Zhu, Chen, & Hu, 2014). Particularly, the Bayesian max-margin models are well studied for various machine learning applications, but they still lack fast inference methods. Our method accomplishes fast sampling for the BMM models, which will be used in future large scale intelligent systems.

References

Ahn, S., Shahbaba, B., & Welling, M. (2014). Distributed stochastic gradient MCMC. *ICML*.

Arnold, V. I. (1978). *Mathematical methods of classical mechanics*: vol. 60. Springer.

Asuncion, A., & Newman, D. (2007). UCI machine learning repository.

Bajaj, C. L. (1993). Multi-dimensional Hermite interpolation and approximation for modelling and visualization. In *ICCG* (pp. 335–348). Citeseer.

Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *COMPSTAT* (pp. 177–186). Springer.

Chen, C., Ding, N., & Carin, L. (2015). On the convergence of stochastic gradient MCMC algorithms with high-order integrators. *NIPS*.

Chen, T., Fox, E., & Guestrin, C. (2014). Stochastic gradient Hamiltonian Monte Carlo. *ICML*.

Collobert, R., Bengio, S., & Bengio, Y. (2002). A parallel mixture of svms for very large scale problems. *Neural Computation, 14*(5), 1105–1114.

Ding, N., Fang, Y., Babbush, R., Chen, C., Skeel, R. D., & Neven, H. (2014). Bayesian sampling using stochastic gradient thermostats. *NIPS*.

Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *JMLR, 12*, 2121–2159.

Fu, Z., Robles-Kelly, A., & Zhou, J. (2010). Mixing linear SVMs for nonlinear classification. *Neural Networks, IEEE Transactions on, 21*(12), 1963–1975.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2014). *Bayesian data analysis*: vol. 2. Taylor & Francis.

Germain, P., Lacasse, A., Laviolette, F., & Marchand, M. (2009). PAC-Bayesian learning of linear classifiers. *ICML*.

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences, 101*(suppl 1), 5228–5235.

Korattikara, A., Chen, Y., & Welling, M. (2014). Austerity in MCMC land: Cutting the Metropolis-Hastings budget. *ICML*.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., et al. (2015). DB pedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web, 6*(2), 167–195.

Li, C., Chen, C., Carlson, D., & Carin, L. (2016). Preconditioned stochastic gradient Langevin dynamics for deep neural networks. *AAAI*.

Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C. et al. (2011). Big data: The next frontier for innovation, competition, and productivity. McKinsey Global Institute (http://www.mckinsey.com/insights/mgi/research/technology_and_innovation/big_data_the_next_frontier_for_innovation; accessed August 4, 2012).

McAllester, D. (2003). Pac-bayesian stochastic model selection. *Machine Learning, 51*, 5–21.

Mimno, D., Hoffman, M., & Blei, D. (2012). Sparse stochastic inference for latent dirichlet allocation. *ICML*.

Neal, R. M. (2012). MCMC using Hamiltonian dynamics. arXiv preprint, arXiv:1206.1901.

Park, Y., Park, S., Jung, W., & Lee, S.-g. (2015). Reversed CF: A fast collaborative filtering algorithm using a k-nearest neighbor graph. *Expert Systems with Applications, 42*(8), 4022–4028.

Patterson, S., & Teh, Y. W. (2013). Stochastic gradient Riemannian langevin dynamics on the probability simplex. *NIPS*.

Pearl, J. (2014). *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann.

Pitman, J. (2006). *Combinatorial stochastic processes: Ecole D'eté de probabilités de Saint-Flour XXXII-2002*. Springer.

Polson, N. G., & Scott, S. L. (2011). Data augmentation for support vector machines. *Bayesian Analysis, 6*(1), 1–23.

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, 400–407.

Roberts, G. O., & Stramer, O. (2002). Langevin diffusions and Metropolis-Hastings algorithms. *Methodology and Computing in Applied Probability, 4*(4), 337–357.

Sato, I., & Nakagawa, H. (2014). Approximation analysis of stochastic gradient Langevin dynamics by using Fokker-Planck equation and ito process. *Icml*.

Shahbaba, B., & Neal, R. (2009). Nonlinear models using Dirichlet process mixtures. *JMLR, 10*, 1829–1850.

Shi, T., & Zhu, J. (2014). Online Bayesian passive-aggressive learning. *Icml*.

Shor, N. Z., Kiwiel, K. C., & Ruszcayski, A. (1985). *Minimization methods for non-differentiable functions*. Springer-Verlag New York, Inc..

Teh, Y. W., Thiéry, A., & Vollmer, S. (2014). Consistency and fluctuations for stochastic gradient Langevin dynamics. arXiv preprint, arXiv:1409.0578.

Tian, T., & Zhu, J. (2015). Max-margin majority voting for learning from crowds. *Nips*.

Welling, M., & Teh, Y. W. (2011). Bayesian learning via stochastic gradient Langevin dynamics. *ICML*.

Xu, M., Zhu, J., & Zhang, B. (2012). Nonparametric max-margin matrix factorization for collaborative prediction. *NIPS*.

Xu, M., Zhu, J., & Zhang, B. (2013). Fast max-margin matrix factorization with data augmentation. *ICML*.

Zhang, A., Zhu, J., & Zhang, B. (2014). Max-margin infinite hidden Markov models. *ICML*.

Zhu, J. (2012). Max-margin nonparametric latent feature models for link prediction. *ICML*.

Zhu, J., Ahmed, A., & Xing, E. P. (2012). MedLDA: Maximum margin supervised topic models. *JMLR, 13*(1), 2237–2278.

Zhu, J., Chen, J., & Hu, W. (2014). Big learning with Bayesian methods. arXiv preprint arXiv:1411.6370.

Zhu, J., Chen, N., Perkins, H., & Xing, E. P. (2014). Gibbs max-margin topic models with data augmentation. *JMLR, 15*, 1073–1110.

Zhu, J., Chen, N., & Xing, E. P. (2011). Infinite SVM: a Dirichlet process mixture of large-margin kernel machines. *ICML*.

Zhu, J., Chen, N., & Xing, E. P. (2014). Bayesian inference with posterior regularization and applications to infinite latent svms. *JMLR, 15*, 1799–1847.