An Adaptive Learning Rate for Stochastic Variational Inference

Rajesh Ranganath RAJESHR@CS.PRINCETON.EDU
Princeton University, 35 Olden St., Princeton, NJ 08540

Chong Wang CHONGW@CS.CMU.EDU
Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA, 15213

David M. Blei BLEI@CS.PRINCETON.EDU
Princeton University, 35 Olden St., Princeton, NJ 08540

Eric P. Xing EPXING@CS.CMU.EDU
Carnegie Mellon University, 5000 Forbes Ave., Pittsburgh, PA, 15213

Abstract

Stochastic variational inference finds good posterior approximations of probabilistic models with very large data sets. It optimizes the variational objective with stochastic optimization, following noisy estimates of the natural gradient. Operationally, stochastic inference iteratively subsamples from the data, analyzes the subsample, and updates parameters with a decreasing learning rate. However, the algorithm is sensitive to that rate, which usually requires hand-tuning to each application. We solve this problem by developing an adaptive learning rate for stochastic inference. Our method requires no tuning and is easily implemented with computations already made in the algorithm. We demonstrate our approach with latent Dirichlet allocation applied to three large text corpora. Inference with the adaptive learning rate converges faster and to a better approximation than the best settings of hand-tuned rates.

1 Introduction

Stochastic variational inference lets us use complex probabilistic models to analyze massive data (Hoffman et al., to appear). It has been applied to topic models (Hoffman et al., 2010), Bayesian nonparametric models (Wang et al., 2011; Paisley et al., 2012), and models of large social networks (Gopalan et al., 2012). In each of these settings, stochastic variational inference finds as good posterior approximations as traditional variational inference, but scales to much larger data sets.

Variational inference tries to find the member of a simple class of distributions that is close to the true posterior (Bishop, 2006). The main idea behind stochastic variational inference is to maximize the variational objective function with stochastic optimization (Robbins & Monro, 1951). At each iteration, we follow a noisy estimate of the natural gradient (Amari, 1998) obtained by subsampling from the data. Following these stochastic gradients with a decreasing learning rate guarantees convergence to a local optimum of the variational objective.

Traditional variational inference scales poorly because it has to analyze the entire data set at each iteration. Stochastic variational inference scales well because at each iteration it needs only to analyze a subset. Further, computing the stochastic gradient on the subset is just as easy as running an iteration of traditional inference on data of the subset's size. Thus any implementation of traditional inference is simple to adapt to stochastic inference.

However, stochastic variational inference is sensitive to the learning rate, a nuisance that must be set in advance. With a quickly decreasing learning rate it moves too cautiously; with a slowly decreasing learning rate it makes erratic and unreliable progress. In either case, convergence is slow and performance suffers.

In this paper, we develop an adaptive learning rate for stochastic variational inference. The step size decreases when the variance of the noisy gradient is large, mitigating the risk of taking a large step in the wrong direction.

Proceedings of the 30^{th} International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

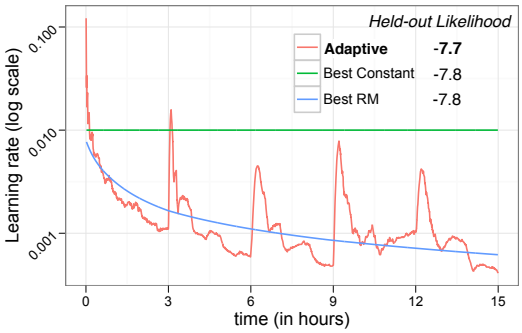

Figure 1. The adaptive learning rate on a run of stochastic variational inference, compared to the best Robbins-Monro and best constant learning rate. Here the data arrives non-uniformly, changing its distribution every three hours. (The algorithms do not know this.) The adaptive learning rate spikes when the data distribution changes. This leads to better predictive performance, as indicated by the held-out likelihood in the top right.

The step size increases when the norm of the expected noisy gradient is large, indicating that the algorithm is far away from the optimal point. With this approach, the user need not set any learning-rate parameters to find a good variational distribution, and it is implemented with computations already made within stochastic inference. Further, we found it consistently led to improved convergence and estimation over the best decreasing and constant rates.

Figure 1 displays three learning rates: a constant rate, a rate that satisfies the conditions of Robbins & Monro (1951), and our adaptive rate. These come from three fits of latent Dirichlet allocation (LDA) (Blei et al., 2003) to a corpus of 1.8M New York Times articles. At each iteration, the algorithm subsamples a small set of documents and updates its estimate of the posterior.

We can see that the adaptive learning rate exhibits a special pattern. The reason is that in this example we subsampled the documents in two-year increments. This engineers the data stream to change at each epoch, and the adaptive learning rate adjusts itself to those changes. The held-out likelihood scores (in the top right) indicate that the resulting variational distribution gave better predictions than the two competitors. (We note that the adaptive learning rate also gives better performance when the data are sampled uniformly. See Figure 3.)

Stochastic variational inference assumes that data are subsampled uniformly at each iteration and is sensitive to the chosen learning rate (Hoffman et al., to appear). The adaptive learning rate developed here solves these problems. It accommodates practical data streams, like the chronological example of Figure 1, where it is difficult to uniformly subsample the data; and it gives a robust way to use stochastic inference without hand-tuning.

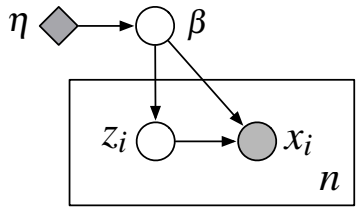

Figure 2. Graphical model for hierarchical Bayesian models with global hidden variables β, local hidden variables $z_{1:n}$, and local observations $x_{1:n}$. The hyperparameter η is fixed.

In the main paper, we review stochastic variational inference, derive our algorithm for adaptively setting the step sizes, and present an empirical study using LDA on three large corpora. In the appendices, we present proofs and a discussion of convergence. Our adaptive algorithm requires no hand-tuning and performs better than the best hand-tuned alternatives.

2 Stochastic Variational Inference

Variational inference performs approximate posterior inference by solving an optimization problem. The idea is to posit a family of distributions with free variational parameters, and then fit those parameters to find the member of the family close (in KL divergence) to the posterior. In this section, we review mean-field variational inference for a large class of models. We first define the model class, describe the variational objective function, and define the mean-field variational parameters. We then derive both the "classical" coordinate ascent inference method and stochastic variational inference, a scalable alternative to coordinate ascent inference. In the next section, we describe and derive our method for adapting the learning rate in stochastic variational inference.

Model family. We consider the family of models in Figure 2 (Hoffman et al., to appear). There are three types of random variables. The observations are $x_{1:n}$, the local hidden variables are $z_{1:n}$, and the global hidden variables are β. The model assumes that the observations and their corresponding local hidden variables are conditionally independent given the global hidden variables,

$$p(\beta, x_{1:n}, z_{1:n} \mid \eta) = p(\beta \mid \eta) \prod_{i=1}^{n} p(z_i, x_i \mid \beta). \quad (1)$$

Further, these distributions are in the exponential family,

$$p(z_i, x_i \mid \beta) = h(z_i, x_i) \exp\left\{\beta^\top t(z_i, x_i) - a(\beta)\right\} \quad (2)$$

$$p(\beta \mid \eta) = h(\beta) \exp\left\{\eta^\top t(\beta) - a(\eta)\right\}, \quad (3)$$

where we overload notation for the base measures $h(\cdot)$, sufficient statistics $t(\cdot)$, and log normalizers $a(\cdot)$. The term $t(\beta)$ has the form $t(\beta) = [\beta; -a(\beta)]$.

Finally, the model satisfies *conditional conjugacy*. The prior $p(\beta \mid \eta)$ is conjugate to $p(z_i, x_i \mid \beta)$. This means that the distribution of the global variables given the observations and local variables $p(\beta \mid z_{1:n}, x_{1:n})$ is in the same family as the prior $p(\beta \mid \eta)$. This differs from classical Bayesian conjugacy because of the local variables. In this model class, the conditional distribution given only the observations $p(\beta \mid x_{1:n})$ is not generally in the same family as the prior.

This is a large class of models. It includes Bayesian Gaussian mixtures, latent Dirichlet allocation (Blei et al., 2003), probabilistic matrix factorization (Salakhutdinov & Mnih, 2008), hierarchical linear regression (Gelman & Hill, 2007), and many Bayesian nonparametric models (Hjort et al., 2010).

Note that in Eq. 1 and 2 we used the joint conditional $p(z_i, x_i \mid \beta)$. This simplifies the set-up in Hoffman et al. (to appear), who assume the local prior $p(z_i \mid \beta)$ is conjugate to $p(x_i \mid z_i, \beta)$. We also make this assumption, but writing the joint conditional eases our derivation of the adaptive learning-rate algorithm.

Variational objective and mean-field family. Our goal is to approximate the posterior distribution $p(\beta, z_{1:n} \mid x_{1:n})$ using variational inference. We approximate it by positing a variational family $q(\beta, z_{1:n})$ over the hidden variables β and $z_{1:n}$ and then finding the member of that family close in KL divergence to the posterior. This optimization problem is equivalent to maximizing the *evidence lower bound* (ELBO), a bound on the marginal probability $p(x_{1:n})$,

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(x_{1:n}, z_{1:n}, \beta)] - \mathbb{E}_q[\log q(z_{1:n}, \beta)].$$

To solve this problem, we must specify the form of the variational distribution $q(\beta, z_{1:n})$. The simplest family is the *mean-field family*, where each hidden variable is independent and governed by its own variational parameter,

$$q(z_{1:n}, \beta) = q(\beta \mid \lambda) \textstyle\prod_{i=1}^n q(z_i \mid \phi_i). \tag{4}$$

We assume that each variational distribution comes from the same family as the conditional.

With the family defined, we now write the ELBO in terms of the variational parameters

$$\mathcal{L}(\lambda, \phi_{1:n}) = \mathbb{E}_q[\log p(\beta \mid \eta)] - \mathbb{E}_q[\log q(\beta \mid \lambda)] + \textstyle\sum_{i=1}^n \left(\mathbb{E}_q[\log p(x_i, z_i \mid \beta)] - \mathbb{E}_q[\log q(z_i \mid \phi_i)]\right).$$

Mean-field variational inference optimizes this function.

Coordinate ascent variational inference. We focus on optimizing the global variational parameter λ. We write the ELBO as a function of this parameter, implicitly optimizing the local parameters $\phi_{1:n}$ for each value of λ,

$$\mathcal{L}(\lambda) \triangleq \max_{\phi} \mathcal{L}(\lambda, \phi_{1:n}). \tag{5}$$

We call this the λ-ELBO. It has the same optimum as the full ELBO $\mathcal{L}(\lambda, \phi_{1:n})$.

Define the λ-optimal local variational parameters $\phi^\lambda = \arg\max_\phi \mathcal{L}(\lambda, \phi)$.[1] Using the distributions in Eq. 2 and Eq. 3, the λ-ELBO is

$$\mathcal{L}(\lambda) = a(\lambda) + \nabla_\lambda a(\lambda)^\top \left(-\lambda + \eta + \textstyle\sum_{i=1}^n \bar{t}_{\phi_i^\lambda}(x_i)\right) + c, \tag{6}$$

where $\bar{t}_{\phi_i^\lambda}(x_i) \triangleq \mathbb{E}_{q(z_i \mid \phi_i^\lambda)}[t(x_i, z_i)]$ and c is a constant that does not depend on λ.

The natural gradient (Amari, 1998) of the λ-ELBO is

$$\tilde{\nabla}\mathcal{L}(\lambda) = -\lambda + \eta + \textstyle\sum_{i=1}^n \bar{t}_{\phi_i^\lambda}(x_i). \tag{7}$$

We derive this in the appendix.

With this perspective, coordinate ascent variational inference can be interpreted as a fixed-point iteration. Define

$$\lambda_t^* = \eta + \textstyle\sum_{i=1}^n \bar{t}_{\phi_i^{\lambda_t}}(x_i) \tag{8}$$

This is the optimal value of λ when $\phi_{1:n}$ is fixed at the optimal $\phi_{1:n}^\lambda$ for $\lambda = \lambda_t$. Given the current estimate of the global parameters λ_t, coordinate ascent iterates between (a) computing the optimal local parameters for the current setting of the global parameters λ_t and (b) using Eq. 8 to update $\lambda_{t+1} = \lambda_t^*$. Each setting of λ_t in this sequence increases the ELBO.

Hoffman et al. (to appear) point out that this is inefficient. At each iteration Eq. 8 requires analyzing all the data to compute $\phi_{1:n}^{\lambda_t}$, which is infeasible for large data sets. The solution is to use stochastic optimization.

Stochastic variational inference. Stochastic inference optimizes the ELBO by following noisy estimates of the natural gradient, where the noise is induced by repeatedly subsampling from the data.

Let i be a random data index, $i \sim \text{Unif}(1, \ldots, n)$. The ELBO with respect to this index is

$$\mathcal{L}_i(\lambda) = a(\lambda) + \nabla_\lambda a(\lambda)^\top(-\lambda + \eta + n\bar{t}_{\phi_i^\lambda}(x_i)). \tag{9}$$

[1] The optimal $\phi_{1:n}^\lambda$ can be obtained with n local closed-form optimizations through standard mean-field updates (Bishop, 2006). This takes the advantage of the model assumption that the local prior $p(z_i \mid \beta)$ is conjugate to $p(x_i \mid z_i, \beta)$. We omit the details here; see Hoffman et al. (to appear).

The expectation of $\mathcal{L}_i(\lambda)$, with respect to the random data index, is equal to the λ-ELBO in Eq. 6. Thus we can obtain noisy estimates of the gradient of the ELBO by sampling a data point index and taking the gradient of $\mathcal{L}_i(\lambda)$. We follow such estimates with a decreasing step-size ρ_t. This is a stochastic optimization algorithm (Robbins & Monro, 1951) of the variational objective.

We now compute the natural gradient of Eq. 9. Notice that $\mathcal{L}_i(\lambda)$ is equal to the λ-ELBO, but where x_i is repeatedly seen n times. So, we can use the natural gradient in Eq. 7 to compute the natural gradient of $\mathcal{L}_i(\lambda)$,

$$\tilde{\nabla}\mathcal{L}_i(\lambda) = -\lambda + \eta + n\bar{t}_{\phi_i^\lambda}(x_i). \tag{10}$$

Following noisy gradients with a decreasing step-size ρ_t gives us stochastic variational inference. At iteration t:

1. Sample a data point $i \sim \text{Unif}(1, \ldots, n)$.
2. Compute the optimal local parameter $\phi_i^{\lambda_t}$ for the ith data point given current global parameters λ_t.
3. Compute intermediate global parameters to be the coordinate update for λ under $\mathcal{L}_i$,

$$\hat{\lambda}_t = \eta + n\bar{t}_{\phi_i^{\lambda_t}}(x_i). \tag{11}$$

4. Set the global parameters to be a weighted average of the current setting and intermediate parameters,

$$\lambda_{t+1} = (1-\rho_t)\lambda_t + \rho_t\hat{\lambda}_t. \tag{12}$$

This is much more efficient than coordinate ascent inference. Rather than analyzing the whole data set before updating the global parameters, we need only analyze a single sampled data point.

As an example, and the focus of our empirical study, consider probabilistic topic modeling (Blei et al., 2003). A topic model is a probabilistic model of a text corpus. A topic is a distribution over a vocabulary, and each document exhibits the topics to different degree. Topic modeling analysis is a posterior inference problem: Given the corpus, we compute the conditional distribution of the topics and how each document exhibits them. This posterior is frequently approximated with variational inference.

The global variables in a topic model are the topics, a set of distributions over the vocabulary that is shared across the entire collection. The local variables are the topic proportions, hidden variables that encode how each document exhibits the topics. Given the global topics, these local variables are conditionally independent.

At any iteration of coordinate ascent variational inference, we have a current estimate of the topics and we use them to examine each document. With stochastic inference, we need only to analyze a subsample of documents at each iteration. Stochastic inference lets topic modelers analyze massive collections of documents (Hoffman et al., 2010).

To assure convergence of the algorithm the step sizes ρ_t (used in Eq. 12) need to satisfy the conditions of Robbins & Monro (1951),

$$\textstyle\sum_{t=1}^{\infty}\rho_t = \infty, \quad \sum_{n=1}^{\infty}\rho_t^2 < \infty. \tag{13}$$

Choosing this sequence can be difficult and time-consuming. A sequence that decays too quickly may take a long time to converge; a sequence that decays too slowly may cause the parameters to oscillate too much. To address this, we propose a new method to adapt the learning rate in stochastic variational inference.

3 An Adaptive Learning Rate for Stochastic Variational Inference

Our method for setting the learning rate adapts to the sampled data. It is designed to minimize the expected distance between the stochastic update in Eq. 12 to the optimal global variational parameter λ_t^* in Eq. 8, i.e., the setting that guarantees the ELBO increases. The expectation of the distance is taken with respect to the randomly sampled data index. The adaptive learning rate is pulled by two signals. It grows when the current setting is expected to be far away from the coordinate optimal setting, but it shrinks with our uncertainty about this distance.

In this section, we describe the objective function and compute the adaptive learning rate. We then describe how to estimate this rate, which depends on several unknown quantities, by computing moving averages within stochastic variational inference.

The adaptive learning rate. Our goal is to set the learning rate ρ_t in Eq. 12. The expensive *batch update* λ_t^* in Eq. 8 is the update we would make if we processed the entire data set; the cheaper *stochastic update* λ_{t+1} in Eq. 12 only requires processing one data point. We estimate the learning rate that minimizes the expected error between the stochastic update and batch update.

The squared norm of the error is

$$J(\rho_t) \triangleq (\lambda_{t+1} - \lambda_t^*)^\top(\lambda_{t+1} - \lambda_t^*). \tag{14}$$

Note this is a random variable because λ_{t+1} depends on a randomly sampled data point. We obtain the *adaptive learning rate* ρ_t^* by minimizing $\mathbb{E}_n[J(\rho_t)|\lambda_t]$. This leads to a stochastic update that is close in expectation to the

batch update.[2]

The randomness in $J(\rho_t)$ from Eq. 14 comes from the intermediate global parameter $\hat{\lambda}_t$ in Eq. 11. Its mean and covariance are

$$\mathbb{E}_n[\hat{\lambda}_t] = \lambda_t^*,$$
$$\text{Cov}_n[\hat{\lambda}_t] = \mathbb{E}_n[(\hat{\lambda}_t - \lambda_t^*)(\hat{\lambda}_t - \lambda_t^*)^\top] \triangleq \Sigma. \quad (15)$$

Minimizing $\mathbb{E}_n[J(\rho_t)|\lambda_t]$ with respect to ρ_t gives

$$\rho_t^* = \frac{(\lambda_t^* - \lambda_t)^\top(\lambda_t^* - \lambda_t)}{(\lambda_t^* - \lambda_t)^\top(\lambda_t^* - \lambda_t) + \text{tr}(\Sigma)}. \quad (16)$$

The derivation can be found in the appendix. The learning rate grows when the batch update λ_t^* is far from the current parameter λ_t. The learning rate shrinks, through the trace term, when the intermediate parameter has high variance (i.e., uncertainty) around the batch update.

However, this learning rate depends on unknown quantities—the batch update λ_t^* and the variance Σ of the intermediate parameters around it. We now describe our algorithm for estimating the adaptive learning rate within the stochastic variational inference algorithm.

Estimating the adaptive learning rate. In this section we estimate the adaptive learning rate in Eq. 16 within the stochastic variational inference algorithm. We do this by expressing it in terms of expectations of the noisy natural gradient of the ELBO—a quantity that is easily computed within stochastic inference—and then approximating those expectations with moving averages.

Let g_t be the sampled natural gradient defined in Eq. 10 at iteration t. Given the current estimate of global variational parameters λ_t, the expected value of g_t is the difference between the current parameter and batch update,

$$\mathbb{E}_n[g_t] = \mathbb{E}_n[-\lambda_t + \hat{\lambda}_t] = -\lambda_t + \lambda_t^*. \quad (17)$$

Its covariance is equal to the covariance of the intermediate parameters $\hat{\lambda}_t$

$$\text{Cov}_n[g_t] = \text{Cov}_n[\hat{\lambda}_t] = \Sigma.$$

We can now rewrite the denominator of the adaptive learning rate in Eq. 16,

$$\mathbb{E}_n[g_t^\top g_t] = \mathbb{E}_n[g_t]^\top \mathbb{E}_n[g_t] + \text{tr}(\Sigma)$$
$$= (-\lambda_t + \lambda_t^*)^\top(-\lambda_t + \lambda_t^*) + \text{tr}(\Sigma).$$

[2] We also considered a more general error function with a metric to account for the non-Euclidean relationship between parameters. For the variational family we studied in Section 4, the Dirichlet, this did not make a difference. However, we suspect a metric could play a role in variational families with more curvature, such as the Gaussian.

Algorithm 1 Learning Rate Free Inference.

Initialize global variational parameter λ_1 randomly.
Initialize the window size τ_1. (See text for details.)
Initialize moving averages $\overline{g}_0$ and $\overline{h}_0$ given λ_1.
for $t = 1, ..., \infty$ **do**
  Sample a data point i.
  Compute the optimal local parameter $\phi_i^{\lambda_t}$.
  Set intermediate global parameters $\hat{\lambda}_t = \eta + n\bar{t}_{\phi_i^{\lambda_t}}(x_i)$.
  Update moving averages $\overline{g}_t$ and $\overline{h}_t$ as in Eq. 19 and 20.
  Set the estimate step size $\rho_t^* = \frac{\overline{g}_t^\top \overline{g}_t}{\overline{h}_t}$.
  Update the window size $\tau_{t+1} = \tau_t(1 - \rho_t^*) + 1$.
  Update global parameters $\lambda_{t+1} = (1 - \rho_t^*)\lambda_t + \rho_t^* \hat{\lambda}_t$.
end for
return λ.

Using this expression and the expectation value of the noisy natural gradient in Eq. 17, we rewrite the adaptive learning rate as

$$\rho_t^* = \frac{\mathbb{E}_n[g_t]^\top \mathbb{E}_n[g_t]}{\mathbb{E}_n[g_t^\top g_t]}. \quad (18)$$

Note that this transformation collapses estimating the required matrix Σ into the estimation of a scalar.

We could form a Monte Carlo estimate of these expectations by drawing multiple samples from the data at the current time step t and repeatedly forming noisy natural gradients. Unfortunately, this reduces the computational benefits of using stochastic optimization. Instead we adapt the method of Schaul et al. (2012), approximating the expectations within the stochastic algorithm with exponential moving averages across iterations.

Let the moving averages for $\mathbb{E}[g_t]$ and $\mathbb{E}[g_t^\top g_t]$ be denoted by $\overline{g}_t$ and $\overline{h}_t$ respectively. Let τ_t be the window size of the exponential moving average at time t. The updates are

$$\mathbb{E}[g_t] \approx \overline{g}_t = (1 - \tau_t^{-1})\overline{g}_{t-1} + \tau_t^{-1} g_t \quad (19)$$
$$\mathbb{E}[g_t^\top g_t] \approx \overline{h}_t = (1 - \tau_t^{-1})\overline{h}_{t-1} + \tau_t^{-1} g_t^\top g_t. \quad (20)$$

Plugging these estimates into Eq. 18, we can approximate the adaptive learning rate with

$$\rho_t^* \approx \frac{\overline{g}_t^\top \overline{g}_t}{\overline{h}_t}.$$

The moving averages are less reliable after large steps, so we update our memory size using the following rule

$$\tau_{t+1} = \tau_t(1 - \rho_t^*) + 1.$$

The full algorithm is in Algorithm 1.

We initialize the moving averages through Monte Carlo estimates of the expectations at the initialization of the

global parameters λ_1 and initialize τ_1 to be the number of samples used in to construct the Monte Carlo estimate. Finally, while we have described the algorithm with a single sampled data point at each iteration, it generalizes easily to mini-batches, i.e., when we sample a small subset of data at each iteration.

Convergence. In Section 4 we found that our algorithm converges in practice. However, proving convergence is an open problem. As a step towards solving this problem, we prove convergence of λ_t under a learning rate that minimizes the error to an unspecified local optimum λ^* rather than to the optimal batch update. (The complexity with the optimal batch update is that it changes at each iteration.) We call the resulting learning rate a_t the *idealized learning rate*. The objective we minimize is the expected value of

$$Q(a_t) \triangleq (\lambda_{t+1} - \lambda^*)^\top (\lambda_{t+1} - \lambda^*).$$

The idealized optimal learning rate that minimizes Q is

$$a_t^* = \frac{(\lambda_t^* - \lambda_t)^\top (\lambda_t^* - \lambda_t) + (\lambda_t^* - \lambda_t)^\top (\lambda^* - \lambda_t^*)}{(\lambda_t^* - \lambda_t)^\top (\lambda_t^* - \lambda_t) + tr(\Sigma)}.$$

The appendix gives a proof of convergence to a local optimum with learning rate a_t^*. We cannot compute this learning rate because it depends on λ^* and λ_t^*. Further, λ^* is hard to estimate.

Compared to the adaptive learning rate in Eq. 16, the idealized learning rate contains an additional term in the numerator. If the batch update λ_t^* is close to the local optimum λ^* then the learning rates are equivalent.

Related work. Schaul et al. (2012) describe the optimal rate for a noisy quadratic objective by minimizing the expected objective at each time step. They used their rate to fit neural networks. A possible generalization of their work to stochastic inference would be to take the Taylor expansion of the ELBO around the current parameter and maximize it with respect to the learning rate. However, we found (empirically) that this approach is inadequate. The Taylor approximation is poor when the step size is large, and the Hessian of the ELBO is not always negative definite. This led to unpredictable behaviors in this algorithm.

Our algorithm is in the same spirit of the approach proposed in Chien & Fu (1967). They studied the problem of stochastically estimating the mean of a normal distribution in a Bayesian setting. They derived an adaptive learning rate by minimizing the squared error to the unknown posterior mean. This has also been pursued for tracking problems by George & Powell (2006). Our approach differs from theirs in that our objective is tailored to variational inference and is defined in terms of the per-iteration coordinate optima (rather than a global optimum). Further, our estimators are based on the sampled natural gradient.

4 Empirical Study

We evaluate our adaptive learning rate with latent Dirichlet allocation (LDA). We consider stochastic inference in two settings: where the data are subsampled uniformly (i.e., where the theory holds) and where they are subsampled in a non-stationary way. In both settings, adaptive learning rates outperform the best hand-tuned alternatives.

Data sets. We analyzed three large corpora: *Nature*, *New York Times*, and *Wikipedia*. The *Nature* corpus contains 340K documents and a vocabulary of 4,500 terms; the *New York Times* corpus contains 1.8M documents and a vocabulary vocabulary of 8,000 terms; the *Wikipedia* corpus contains 3.6M documents and a vocabulary of 7,700 terms.

Evaluation metric. To evaluate our models, we held out 10,000 documents from each corpus and calculated its predictive likelihood. We follow the metric used in recent topic modeling literature (Asuncion et al., 2009; Wang et al., 2011; Hoffman et al., to appear). For a document w_d in $\mathcal{D}_{\text{test}}$, we split it in into halves, $w_d = (w_{d1}, w_{d2})$, and computed the predictive log likelihood of the words in w_{d2} conditioned on w_{d1} and $\mathcal{D}_{\text{train}}$. A better predictive distribution given the first half should give higher likelihood to the second half. The per-word predictive log likelihood is defined as

$$\text{likelihood}_{\text{pw}} \triangleq \frac{\sum_{d \in \mathcal{D}_{\text{test}}} \log p(w_{d2} | w_{d1}, \mathcal{D}_{\text{train}})}{\sum_{d \in \mathcal{D}_{\text{test}}} |w_{d2}|}.$$

Here $|w_{d2}|$ is the number of tokens in w_{d2}. This evaluation measures the quality of the estimated predictive distribution. It lets us compare methods regardless of whether they provide a bound. However, this quantity is intractable in general and so we used the same strategy as in Hoffman et al. (to appear) to approximate it.

Hyperparameters and alternative learning rates. We set the mini-batch size to 100 documents. We used 100 topics and set the Dirichlet hyperparameter on the topics to be 0.01. LDA also contains a Dirichlet prior over

the topic proportions, i.e., how much each document exhibits each topic. We set this to the uniform prior. These are the same values used in Hoffman et al. (2010).

We compared the adaptive learning rate to two others. Hoffman et al. (2010) use a learning rate of the form

$$\rho_t = (t_0 + t)^{-\kappa}, \tag{21}$$

where $\kappa \in (0.5, 1]$. This choice satisfies the Robbins-Monro conditions Eq. 13. They used a grid search to find the best parameters for this rate. Note that Snoek et al. (2012) showed that Bayesian optimization can speed up this search.

We also compared to a small constant learning rate (Collobert et al., 2011; Nemirovski et al., 2009). We found a best rate by searching over $\{0.1, 0.01, 0.001, 0.0001\}$. We report our results against the best Robbins-Monro learning rate and the best small constant learning rate.

Results. We compared the algorithms in two scenarios. First, we ran stochastic inference as described above. At each iteration, we subsample uniformly from the data, analyze the subsample, and use the learning rate to update the global variational parameters. We ran the algorithms for 20 hours.

Figure 3 shows the results. On all three corpora, the adaptive learning rate outperformed the best Robbins-Monro rate and constant rate.[3] It converged faster and formed better predictive distributions. Figure 4 shows how the learning rate behaves. Without tuning any parameters, it gave a similar shape to the Robbins-Monro rate.

Second, we ran stochastic inference where the data were subsampled non-uniformly. The theory breaks down in such a scenario, but this matches many applied situations and illustrates some of the advantages of an adaptive learning rate. We split the *New York Times* documents in into 10 segments based their publication dates. We ran the algorithm sequentially, sampling from each segment and testing on the next. Each pair of sampled and test segments form an epoch. We trained on each of these 5 epochs for three hours. Throughout the epochs we maintained a single approximate posterior. We compared the adaptive learning rate to the best Robbins-Monro and constant learning rates from the previous experiment.

Figure 1 shows the results. The adaptive learning rate spikes when the underlying sampling distribution changes. (Note that the algorithm does not know when the epoch is changing.) Further, it led to better predictive distributions.

[3] If we change the time budget, we usually find different best Robbins-Monro and constant learning rates. However, in all the cases we studied, the adaptive learning rate outperformed the best alternative rates.

5 Conclusion

We developed and studied an algorithm to adapt the learning rate in stochastic variational inference. Our approach is based on the minimization of the expected squared norm of the error between the global variational parameter and the coordinate optimum. It requires no hand-tuning and uses computations already found inside stochastic inference. It works well for both stationary and non-stationary subsampled data. In our study of latent Dirichlet allocation, it led to faster convergence and a better optimum when compared to the best hand-tuned rates.

Acknowledgements

We thank Tom Schaul and the reviewers for their helpful comments. Rajesh Ranganath is supported by an NDSEG fellowship. David M. Blei is supported by NSF IIS-0745520, NSF, IIS-1247664, NSF IIS-1009542, ONR N00014-11-1-0651, and the Alfred P. Sloan foundation. Chong Wang and Eric P. Xing are supported by AFOSR FA9550010247, NSF IIS1111142, NIH 1R01GM093156 and NSF DBI-0546594.

References

Amari, S. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

Asuncion, A., Welling, M., Smyth, P., and Teh, Y. On smoothing and inference for topic models. In *Uncertainty in Artificial Intelligence*, 2009.

Bishop, C. *Pattern Recognition and Machine Learning*. Springer New York., 2006.

Blei, D., Ng, A., and Jordan, M. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3: 993–1022, January 2003.

Chien, Y. and Fu, K. On Bayesian learning and stochastic approximation. *Systems Science and Cybernetics, IEEE Transactions on*, 3(1):28 –38, jun. 1967.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537, 2011.

Gelman, A. and Hill, J. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge Univ. Press, 2007.

George, A. and Powell, W. Adaptive stepsizes for recursive estimation with applications in approximate

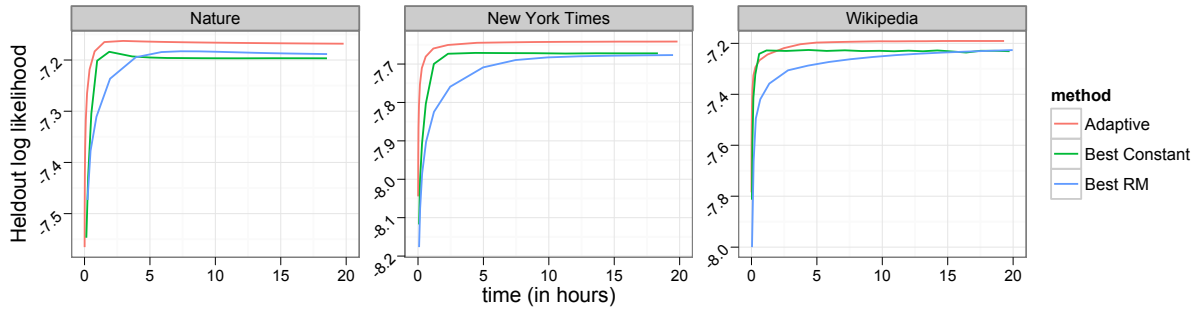

Figure 3. Held-out log likelihood on three large corpora. (Higher numbers are better.) The best Robbins-Monro rates from Eq. 21 were $t_0 = 1000$ and $\kappa = 0.7$ for the New York Times and Nature copora and $t_0 = 1000$ and $\kappa = 0.8$ for Wikipedia corpus. The best constant rate was 0.01. The adaptive learning rate performed best.

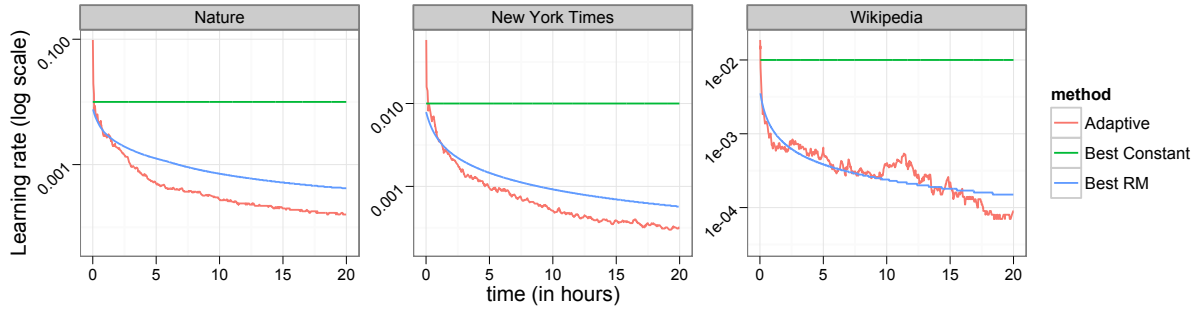

Figure 4. Learning rates. The adaptive learning rate has a similar shape to the best Robbins-Monro learning rate, but is obtained automatically and adapts when the data requires.

dynamic programming. *Machine learning*, 65(1):167–198, 2006.

Gopalan, P., Mimno, D., Gerrish, S., Freedman, M., and Blei, D. Scalable inference of overlapping communities. In *Neural Information Processing Systems*, 2012.

Hjort, N., Holmes, C., Mueller, P., and Walker, S. *Bayesian Nonparametrics: Principles and Practice*. Cambridge University Press, Cambridge, UK, 2010.

Hoffman, M., Blei, D., and Bach, F. Online inference for latent Drichlet allocation. In *Neural Information Processing Systems*, 2010.

Hoffman, M., Blei, D., Wang, C., and Paisley, J. Stochastic variational inference. *Journal of Machine Learning Research*, to appear.

Nemirovski, A., Juditsky, A., Lan, G., and Shapiro, A. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4): 1574–1609, 2009.

Paisley, J., Wang, C., and Blei, D. The discrete infinite logistic normal distribution. *Bayesian Analysis*, 7(2): 235–272, 2012.

Robbins, H. and Monro, S. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3): pp. 400–407, 1951.

Salakhutdinov, R. and Mnih, A. Probabilistic matrix factorization. In *Neural Information Processing Systems*, 2008.

Schaul, T., Zhang, S., and LeCun, Y. No more pesky learning rates. *ArXiv e-prints*, June 2012.

Snoek, J., Larochelle, H., and Adams, R. Practical Bayesian optimization of machine learning algorithms. In *Neural Information Processing Systems*, 2012.

Wang, C., Paisley, J., and Blei, D. Online variational inference for the hierarchical Dirichlet process. In *International Conference on Artificial Intelligence and Statistics*, 2011.

Appendix

Natural gradient of the λ-ELBO. We can compute the natural gradient in Eq. 7 at λ by first finding

the corresponding optimal local parameters $\phi^\lambda = \arg\max_\phi \mathcal{L}(\lambda, \phi)$ and then computing the gradient of $\mathcal{L}(\lambda, \phi^\lambda)$, i.e., the ELBO where we fix $\phi = \phi^\lambda$. These are equivalent because

$$\begin{aligned}\nabla_\lambda \mathcal{L}(\lambda) &= \nabla_\lambda \mathcal{L}(\lambda, \phi^\lambda) + (\nabla_\lambda \phi^\lambda)^\top \nabla_\phi \mathcal{L}(\lambda, \phi^\lambda) \\ &= \nabla_\lambda \mathcal{L}(\lambda, \phi^\lambda).\end{aligned}$$

The notation $\nabla_\lambda \phi^\lambda$ is the Jacobian of ϕ^λ as a function of λ, and we use that $\nabla_\phi \mathcal{L}(\lambda, \phi)$ is zero at $\phi = \phi^\lambda$.

Derivation of the adaptive learning rate. To compute the adaptive learning rate we minimize $\mathbb{E}_n[J(\rho_t)|\lambda_t]$ at each time t. Expanding $\mathbb{E}_n[J(\rho_t)|\lambda_t]$, we get

$$\begin{aligned}\mathbb{E}_n[J(\rho_t)|\lambda_t] =& \mathbb{E}_n[(\lambda_t + \rho_t(\lambda_t - \hat{\lambda}_t) - \lambda_t^*)^\top \\ & (\lambda_t + \rho_t(\lambda_t - \hat{\lambda}_t) - \lambda_t^*)].\end{aligned}$$

We can compute this expectation in terms of the moments of the sample optimum in Eq. 15

$$\begin{aligned}\mathbb{E}_n[J(\rho_t)|\lambda_t] =& (1-\rho_t)^2(\lambda_t^* - \lambda_t)^\top(\lambda_t^* - \lambda_t) \\ & + \rho_t^2 tr(\Sigma).\end{aligned}$$

Setting the derivative of $\mathbb{E}_n[J(\rho_t)|\lambda_t]$ with respect to ρ_t equal to 0 yields the optimal learning in Eq. 16.

Convergence of the idealized learning rate. We show convergence of λ_t to a local optima with our idealized learning rate through martingale convergence. Let $M_{t+1} = Q(a_t^*)$, then M_t is a super-martingale with respect to the natural filtration of the sequence λ_t,

$$\mathbb{E}[M_{t+1}|\lambda_t] = \mathbb{E}[Q(a_t^*)|\lambda_t] \leq \mathbb{E}[Q(0)|\lambda_t] = M_t.$$

Since M_t is a non-negative supermartingale by the martingale convergence theorem, we know that a finite M_∞ exists and $M_t \to M_\infty$ almost surely. Since the M_t converge, the sequence of expected values $\mathbb{E}[M_t]$ converge to $\mathbb{E}[M_\infty]$. This means that the sequence of expected values form a Cauchy sequence, so the difference between elements of the sequence goes to zero,

$$\begin{aligned}D_t &\triangleq \mathbb{E}[M_{t+1}] - \mathbb{E}[M_t] \\ &= \mathbb{E}[\mathbb{E}[M_{t+1}|\lambda_t] - \mathbb{E}[M_t|\lambda_t]] \to 0.\end{aligned}$$

Substituting the idealized optimal learning rate into this expression gives

$$\begin{aligned}D_t =& \mathbb{E}[-((\lambda_t^* - \lambda_t)^\top(\lambda_t^* - \lambda_t) + (\lambda_t^* - \lambda_t)^\top(\lambda^* - \lambda_t^*))^2 \\ & ((\lambda_t^* - \lambda_t)^\top(\lambda_t^* - \lambda_t) + tr(\Sigma))^{-1}]. \qquad (22)\end{aligned}$$

Since the D_t's are a sequence of nonpositive random variables whose expectation goes to zero and that the variances are bounded (by assumption), the square portion of Eq. 22 must go to zero almost surely. This quantity going to zero implies that either $\lambda_t \to \lambda^*$ or $\lambda_t \to \lambda_t^*$. If $\lambda_t = \lambda_t^*$, then λ_t is a local optima under the assumption that the two parameter (ϕ and λ for the ELBO) function we are optimizing can be optimized via coordinate ascent. Putting everything together gives us that λ_t goes to a local optima almost surely.