Faster Stochastic Variational Inference using Proximal-Gradient Methods with General Divergence Functions

Mohammad Emtiyaz Khan
Ecole Polytechnique Fédérale de Lausanne
Lausanne Switzerland
emtiyaz@gmail.com

Reza Babanezhad
University of British Columbia
Vancouver, Canada
babanezhad@gmail.com

Wu Lin
University of Waterloo
Waterloo, Canada
wu.lin@uwaterloo.ca

Mark Schmidt
University of British Columbia
Vancouver, Canada
schmidtm@cs.ubc.ca

Masashi Sugiyama
University of Tokyo
Tokyo, Japan
sugi@k.u-tokyo.ac.jp

Abstract

Several recent works have explored stochastic gradient methods for variational inference that exploit the geometry of the variational-parameter space. However, the theoretical properties of these methods are not well-understood and these methods typically only apply to conditionally-conjugate models. We present a new stochastic method for variational inference which exploits the geometry of the variational-parameter space and also yields simple closed-form updates even for non-conjugate models. We also give a convergence-rate analysis of our method and many other previous methods which exploit the geometry of the space. Our analysis generalizes existing convergence results for stochastic mirror-descent on non-convex objectives by using a more general class of divergence functions. Beyond giving a theoretical justification for a variety of recent methods, our experiments show that new algorithms derived in this framework lead to state of the art results on a variety of problems. Further, due to its generality, we expect that our theoretical analysis could also apply to other applications.

1 INTRODUCTION

Variational inference methods are one of the most widely-used computational tools to deal with the intractability of Bayesian inference, while stochastic gradient (SG) methods are one of the most widely-used tools for solving optimization problems on huge datasets. The last three years have seen an explosion of work exploring SG methods for variational inference (Hoffman et al., 2013; Salimans et al., 2013; Ranganath et al., 2013; Titsias & Lázaro-Gredilla, 2014; Mnih & Gregor, 2014; Kucukelbir et al., 2014). In many settings, these methods can yield simple updates and scale to huge datasets.

A challenge that has been addressed in many of the recent works on this topic is that the "black-box" SG method ignores the geometry of the variational-parameter space. This has lead to methods like the stochastic variational inference (SVI) method of Hoffman et al. (2013), that uses *natural gradients* to exploit the geometry. This leads to better performance in practice, but this approach only applies to conditionally-conjugate models. In addition, it is not clear how using natural gradients for variational inference affects the theoretical convergence rate of SG methods.

In this work we consider a general framework that (i) can be stochastic to allow huge datasets, (ii) can exploit the geometry of the variational-parameter space to improve performance, and (iii) can yield a closed-form update even for non-conjugate models. The new framework can be viewed as a stochastic generalization of the proximal-gradient method of Khan et al. (2015), which splits the objective into conjugate and non-conjugate terms. By linearizing the non-conjugate terms, this previous method as well as our new method yield simple closed-form proximal-gradient updates even for non-conjugate models.

While proximal-gradient methods have been well-studied in the optimization community (Beck & Teboulle, 2009), like SVI there is nothing known about the convergence rate of the method of Khan et al. (2015) because it uses "divergence" functions which do not satisfy standard assumptions. Our second contribution is to *analyze the convergence rate* of the proposed method. In particular, we generalize an existing result on the convergence rate of stochastic mirror descent in non-convex settings (Ghadimi et al., 2014) to allow a general class of divergence functions that includes the cases above (in both deterministic and stochas-

tic settings). While it has been observed empirically that including an appropriate divergence function enables larger steps than basic SG methods, this work gives the first theoretical result justifying the use of these more-general divergence functions. It in particular reveals how different factors affect the convergence rate such as the Lipschitz-continuity of the lower bound, the information geometry of the divergence functions, and the variance of the stochastic approximation. Our results also suggest conditions under which the proximal-gradient steps of Khan et al. (2015) can make more progress than (non-split) gradient steps, and sheds light on the choice of step-size for these methods. A notable aspect of our results is that, for the stochastic case and a fixed accuracy, there is always a sufficiently-small *fixed* step-size that leads to a solution with this accuracy or higher. Our experimental results indicate that the new method leads to improvements in performance on a variety of problems, and we note that the algorithm and theory might be useful beyond the variational inference scenarios we have considered in this work.

2 VARIATIONAL INFERENCE

Consider a general latent variable model where we have a data vector $\mathbf{y}$ of length N and a latent vector $\mathbf{z}$ of length D. In Bayesian inference, we are interested in computing the marginal likelihood $p(\mathbf{y})$, which can be written as the integral of the joint distribution $p(\mathbf{y}, \mathbf{z})$ over all values of $\mathbf{z}$. This integral is often intractable, and in variational inference we typically approximate it with the evidence lower-bound optimization (ELBO) approximation $\underline{\mathcal{L}}$. This approximation introduces a distribution $q(\mathbf{z}|\boldsymbol{\lambda})$ and chooses the variational parameters $\boldsymbol{\lambda}$ to maximize the following lower bound on the marginal likelihood:

$$\begin{aligned} \log p(\mathbf{y}) &= \log \int q(\mathbf{z}|\boldsymbol{\lambda}) \frac{p(\mathbf{y}, \mathbf{z})}{q(\mathbf{z}|\boldsymbol{\lambda})} \, d\mathbf{z}, \\ &\geq \max_{\boldsymbol{\lambda} \in \mathcal{S}} \underline{\mathcal{L}}(\boldsymbol{\lambda}) := \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\lambda})} \left[\log \frac{p(\mathbf{y}, \mathbf{z})}{q(\mathbf{z}|\boldsymbol{\lambda})} \right]. \end{aligned} \tag{1}$$

The inequality follows from concavity of the logarithm function. The set $\mathcal{S}$ is the set of valid parameters $\boldsymbol{\lambda}$.

To optimize $\boldsymbol{\lambda}$, one of the seemingly-simplest approaches is gradient descent: $\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \beta_k \nabla \underline{\mathcal{L}}(\boldsymbol{\lambda}_k)$, which can be viewed as optimizing a quadratic approximation of $\underline{\mathcal{L}}$,

$$\boldsymbol{\lambda}_{k+1} = \operatorname*{argmin}_{\boldsymbol{\lambda} \in \mathcal{S}} \left[-\boldsymbol{\lambda}^T \nabla \underline{\mathcal{L}}(\boldsymbol{\lambda}_k) + \frac{1}{2\beta_k} \|\boldsymbol{\lambda} - \boldsymbol{\lambda}_k\|_2^2 \right]. \tag{2}$$

While we can often choose the family q so that it has convenient computational properties, it might be impractical to apply gradient descent in this context when we have a very large dataset or when some terms in the lower bound are intractable. Recently, SG methods have been proposed to deal with these issues (Ranganath et al., 2013; Titsias & Lázaro-Gredilla, 2014): they allow large datasets by using random subsets (mini-batches) and can approximate intractable integrals using Monte Carlo methods that draw samples from $q(\mathbf{z}|\boldsymbol{\lambda})$.

A second drawback of applying gradient descent to variational inference is that it uses the Euclidean distance and thus ignores the *geometry of the variational-parameter space*, which often results in slow convergence. Intuitively, (2) implies that we should move in the direction of the gradient, but not move $\boldsymbol{\lambda}_{k+1}$ too far away from $\boldsymbol{\lambda}_k$ in terms of the Euclidean distance. However, the Euclidean distance is not appropriate for variational inference because $\boldsymbol{\lambda}$ is the parameter vector of a distribution; the Euclidean distance is often a poor measure of dissimilarity between distributions. The following example from Hoffman et al. (2013) illustrates this point: the two normal distributions $\mathcal{N}(0, 10000)$ and $\mathcal{N}(10, 10000)$ are almost indistinguishable, yet the Euclidean distance between their parameter vectors is 10, whereas the distributions $\mathcal{N}(0, 0.01)$ and $\mathcal{N}(0.1, 0.01)$ barely overlap, but their Euclidean distance between parameters is only 0.1.

Natural-Gradient Methods: The canonical way to address the problem above is by replacing the Euclidean distance in (2) with another divergence function. For example, the *natural gradient* method defines the iteration by using the symmetric Kullback-Leibler (KL) divergence (Hoffman et al., 2013; Pascanu & Bengio, 2013; Amari, 1998),

$$\boldsymbol{\lambda}_{k+1} = \operatorname*{argmin}_{\boldsymbol{\lambda} \in \mathcal{S}} \left[-\boldsymbol{\lambda}^T \nabla \underline{\mathcal{L}}(\boldsymbol{\lambda}_k) + \frac{1}{\beta_k} \mathbb{D}_{KL}^{sym}[q(\mathbf{z}|\boldsymbol{\lambda}) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda}_k)] \right]. \tag{3}$$

This leads to the update

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \beta_k \left[\nabla^2 \mathbf{G}(\boldsymbol{\lambda}_k) \right]^{-1} \nabla \underline{\mathcal{L}}(\boldsymbol{\lambda}_k), \tag{4}$$

where $\mathbf{G}(\boldsymbol{\lambda})$ is the Fisher information-matrix,

$$\mathbf{G}(\boldsymbol{\lambda}) := \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\lambda})} \left\{ [\nabla \log q(\mathbf{z}|\boldsymbol{\lambda})] \, [\nabla \log q(\mathbf{z}|\boldsymbol{\lambda})]^T \right\}.$$

Hoffman et al. (2013) show that the natural-gradient update can be computationally simpler than gradient descent for conditionally-conjugate exponential family models. In this family, we assume that the distribution of $\mathbf{z}$ factorizes as $\prod_i p(\mathbf{z}^i|\text{pa}^i)$ where $\mathbf{z}^i$ are disjoint subsets of $\mathbf{z}$ and pa^i are the parents of the $\mathbf{z}^i$ in a directed acyclic graph. This family also assumes that each conditional distribution is in the exponential family,

$$p(\mathbf{z}^i|\text{pa}^i) := h^i(\mathbf{z}^i) \exp\left[[\boldsymbol{\eta}^i(\text{pa}^i)]^T \mathbf{T}^i(\mathbf{z}^i) - A^i(\boldsymbol{\eta}^i) \right],$$

where $\boldsymbol{\eta}^i$ are the natural parameters, $\mathbf{T}^i(\mathbf{z}^i)$ are the sufficient statistics, $A^i(\boldsymbol{\eta}^i)$ is the partition function, and $h^i(\mathbf{z}^i)$ is the base measure. Hoffman et al. (2013) consider a mean-field approximation $q(\mathbf{z}|\boldsymbol{\lambda}) = \prod_i q^i(\mathbf{z}^i|\boldsymbol{\lambda}^i)$ where

each q^i belongs to the same exponential-family distribution as the joint distribution,

$$q^i(\mathbf{z}^i) := h^i(\mathbf{z}^i) \exp\left[(\boldsymbol{\lambda}^i)^T \mathbf{T}^i(\mathbf{z}^i) - A^i(\boldsymbol{\lambda}^i)\right].$$

The parameters of this distribution are denoted by $\boldsymbol{\lambda}^i$ to differentiate them from the joint-distribution parameters $\boldsymbol{\eta}^i$.

As shown by Hoffman et al. (2013), the Fisher matrix for this problem is equal to $\nabla^2 A^i(\boldsymbol{\lambda}^i)$ and the gradient of the lower bound with respect to $\boldsymbol{\lambda}^i$ is equal to $\nabla^2 A^i(\boldsymbol{\lambda}^i)(\boldsymbol{\lambda}^i - \boldsymbol{\lambda}^i_*)$ where $\boldsymbol{\lambda}^i_*$ are the mean-field parameters (see Paquet, 2014). Therefore, when computing the natural-gradient, the $\nabla^2 A^i(\boldsymbol{\lambda}^i)$ terms cancel out and the natural-gradient is simply $\boldsymbol{\lambda}^i - \boldsymbol{\lambda}^i_*$ which is much easier to compute than the actual gradient. Unfortunately, for non-conjugate models this cancellation does not happen and the simplicity of the update is lost. The Riemannian conjugate-gradient method of Honkela et al. (2011) has similar issues, in that computing $\nabla^2 A(\boldsymbol{\lambda})$ is typically very costly.

KL-Divergence Based Methods: Rather than using the symmetric-KL, Theis & Hoffman (2015) consider using the KL divergence $\mathbb{D}_{KL}[q(\mathbf{z}|\boldsymbol{\lambda}) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda}_k)]$ within a stochastic proximal-point method:

$$\boldsymbol{\lambda}_{k+1} = \underset{\boldsymbol{\lambda}\in\mathcal{S}}{\operatorname{argmin}} \left[-\underline{\mathcal{L}}(\boldsymbol{\lambda}) + \frac{1}{\beta_k}\mathbb{D}_{KL}[q(\mathbf{z}|\boldsymbol{\lambda}) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda}_k)]\right]. \tag{5}$$

This method yields better convergence properties, but requires numerical optimization to implement the update even for conditionally-conjugate models. Khan et al. (2015) considers a deterministic proximal-gradient variant of this method by splitting the lower bound into $-\underline{\mathcal{L}} := f + h$, where f contains all the "easy" terms and h contains all the "difficult" terms. By linearizing the "difficult" terms, this leads to a closed-form update even for non-conjugate models. The update is given by:

$$\boldsymbol{\lambda}_{k+1} = \underset{\boldsymbol{\lambda}\in\mathcal{S}}{\operatorname{argmin}} \Big[\boldsymbol{\lambda}^T[\nabla f(\boldsymbol{\lambda}_k)] + h(\boldsymbol{\lambda}) + \frac{1}{\beta_k}\mathbb{D}_{KL}[q(\mathbf{z}|\boldsymbol{\lambda}) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda}_k)]\Big]. \tag{6}$$

However, this method requires the exact gradients which is usually not feasible for large dataset and/or complex models.

Mirror Descent Methods: In the optimization literature, *mirror descent* (and stochastic mirror descent) algorithms are a generalization of (2) where the squared-Euclidean distance can be replaced by any Bregman divergence $\mathbb{D}_F(\boldsymbol{\lambda}\|\boldsymbol{\lambda}_k)$ generated from a strongly-convex function $F(\boldsymbol{\lambda})$ (Beck & Teboulle, 2003),

$$\boldsymbol{\lambda}_{k+1} = \underset{\boldsymbol{\lambda}\in\mathcal{S}}{\operatorname{argmin}} \left\{-\boldsymbol{\lambda}^T \nabla\underline{\mathcal{L}}(\boldsymbol{\lambda}_k) + \frac{1}{\beta_k}\mathbb{D}_F(\boldsymbol{\lambda}\|\boldsymbol{\lambda}_k)\right\}. \tag{7}$$

The convergence rate of mirror descent algorithm has been analyzed in convex (Duchi et al., 2010) and more recently in non-convex (Ghadimi et al., 2014) settings. However, mirror descent does not cover the cases described above in (5) and (6) when a KL divergence between two exponential-family distributions is used with $\boldsymbol{\lambda}$ as the natural-parameter. For such cases, the Bregman divergence corresponds to a KL divergence with swapped parameters (see Nielsen & Garcia, 2009, Equation 29),

$$\begin{aligned}\mathbb{D}_A(\boldsymbol{\lambda}\|\boldsymbol{\lambda}_k) &:= A(\boldsymbol{\lambda}) - A(\boldsymbol{\lambda}_k) - [\bigtriangledown A(\boldsymbol{\lambda}_k)]^T(\boldsymbol{\lambda} - \boldsymbol{\lambda}_k) \\ &= \mathbb{D}_{KL}[q(\mathbf{z}|\boldsymbol{\lambda}_k) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda})].\end{aligned} \tag{8}$$

where $A(\boldsymbol{\lambda})$ is the partition function of q. Because (5) and (6) both use a KL divergence where the second argument is fixed to $\boldsymbol{\lambda}_k$, instead of the first argument, they are not covered under the mirror-descent framework. In addition, even though mirror-descent has been used for variational inference (Ravikumar et al., 2010), Bregman divergences do not yield an efficient update in many scenarios.

3 PROXIMAL-GRADIENT SVI

Our proximal-gradient stochastic variational inference (PG-SVI) method extends (6) to allow stochastic gradients $\widehat{\nabla} f(\boldsymbol{\lambda}_k)$ and general divergence functions $\mathbb{D}(\boldsymbol{\lambda}\|\boldsymbol{\lambda}_k)$ by using the iteration

$$\boldsymbol{\lambda}_{k+1} = \underset{\boldsymbol{\lambda}\in\mathcal{S}}{\operatorname{argmin}} \left\{\boldsymbol{\lambda}^T\left[\widehat{\bigtriangledown} f(\boldsymbol{\lambda}_k)\right] + h(\boldsymbol{\lambda}) + \frac{1}{\beta_k}\mathbb{D}(\boldsymbol{\lambda} \,\|\, \boldsymbol{\lambda}_k)\right\}. \tag{9}$$

This unifies a variety of existing approaches since it allows:

1. Splitting of $\underline{\mathcal{L}}$ into a difficult term f and a simple term h, similar to the method of Khan et al. (2015).
2. A stochastic approximation $\widehat{\nabla} f$ of the gradient of the difficult term, similar to SG methods.
3. Divergence functions $\mathbb{D}$ that incorporate the geometry of the parameter space, similar to methods discussed in Section 2 (see (3), (5), (6), and (7)).

Below, we describe each feature in detail, along with the precise assumptions used in our analysis.

3.1 SPLITTING

Following Khan et al. (2015), we split the lower bound into a sum of a "difficult" term f and an "easy" term h, enabling a closed-form solution for (9). Specifically, we split using $p(\mathbf{y}, \mathbf{z})/q(\mathbf{z}|\boldsymbol{\lambda}) = c\,\tilde{p}_d(\mathbf{z}|\boldsymbol{\lambda})\tilde{p}_e(\mathbf{z}|\boldsymbol{\lambda})$, where $\tilde{p}_d$ contains all factors that make the optimization difficult, and $\tilde{p}_e$ contains the rest (while c is a constant). By substituting in (1), we

get the following split of the lower bound:

$$\underline{\mathcal{L}}(\boldsymbol{\lambda}) = \underbrace{\mathbb{E}_q[\log \tilde{p}_d(\mathbf{z}|\boldsymbol{\lambda})]}_{-f(\boldsymbol{\lambda})} + \underbrace{\mathbb{E}_q[\log \tilde{p}_e(\mathbf{z}|\boldsymbol{\lambda})]}_{-h(\boldsymbol{\lambda})} + \log c.$$

Note that $\tilde{p}_d$ and $\tilde{p}_e$ need not be probability distributions.

We make the following assumptions about f and h:

(A1) The function f is differentiable and its gradient is $L-$Lipschitz-continuous, i.e. $\forall \boldsymbol{\lambda}$ and $\boldsymbol{\lambda}' \in \mathcal{S}$ we have

$$\|\nabla f(\boldsymbol{\lambda}) - \nabla f(\boldsymbol{\lambda}')\| \leq L\|\boldsymbol{\lambda} - \boldsymbol{\lambda}'\|.$$

(A2) The function h can be a general convex function.

These assumptions are very weak. The function f can be non-convex and the Lipschitz-continuity assumption is typically satisfied in practice (and indeed the analysis can be generalized to only require this assumption on a smaller set containing the iterations). The assumption that h is convex seems strong, but note that we can always take $h = 0$ in the split if the function has no "nice" convex part. Below, we give several illustrative examples of such splits for variational-Gaussian inference with $q(\mathbf{z}|\boldsymbol{\lambda}) := \mathcal{N}(\mathbf{z}|\mathbf{m}, \mathbf{V})$, so that $\boldsymbol{\lambda} = \{\mathbf{m}, \mathbf{V}\}$ with $\mathbf{m}$ being the mean and $\mathbf{V}$ being the covariance matrix.

Gaussian Process (GP) Models: Consider GP models (Kuss & Rasmussen, 2005) for N input-output pairs $\{y_n, \mathbf{x}_n\}$ indexed by n. Let $z_n := f(\mathbf{x}_n)$ be the latent function drawn from a GP with mean 0 and covariance $\mathbf{K}$. We use a non-Gaussian likelihood $p(y_n|z_n)$ to model the output. We can then use the following split, where the non-Gaussian terms are in $\tilde{p}_d$ and the Gaussian terms are in $\tilde{p}_e$:

$$\frac{p(\mathbf{y}, \mathbf{z})}{q(\mathbf{z}|\boldsymbol{\lambda})} = \underbrace{\prod_{n=1}^{N} p(y_n|z_n)}_{\tilde{p}_d(\mathbf{z}|\boldsymbol{\lambda})} \underbrace{\frac{\mathcal{N}(\mathbf{z}|0, \mathbf{K})}{\mathcal{N}(\mathbf{z}|\mathbf{m}, \mathbf{V})}}_{\tilde{p}_e(\mathbf{z}|\boldsymbol{\lambda})}. \tag{10}$$

The detailed derivation is in the appendix. By substituting in (1), we obtain the lower bound $\underline{\mathcal{L}}(\boldsymbol{\lambda})$ shown below along with its split:

$$\underbrace{\sum_n \mathbb{E}_q[\log p(y_n|z_n)]}_{-f(\boldsymbol{\lambda})} - \underbrace{\mathbb{D}_{KL}[\mathcal{N}(\mathbf{z}|\mathbf{m}, \mathbf{V}) \,\|\, \mathcal{N}(\mathbf{z}|0, \mathbf{K})]}_{h(\boldsymbol{\lambda})}. \tag{11}$$

A1 is satisfied for common likelihoods, while it is easy to establish that h is convex. We show in Section 6 that this split leads to a closed-form update for iteration (9).

Generalized Linear Models (GLMs): A similar split can be obtained for GLMs (Nelder & Baker, 1972), where the non-conjugate terms are in $\tilde{p}_d$ and the rest are in $\tilde{p}_e$. Denoting the weights by $\mathbf{z}$ and assuming a standard Gaussian prior over it, we can use the following split:

$$\frac{p(\mathbf{y}, \mathbf{z})}{q(\mathbf{z}|\boldsymbol{\lambda})} = \underbrace{\prod_{n=1}^{N} p(y_n|\mathbf{x}_n^T\mathbf{z})}_{\tilde{p}_d(\mathbf{z}|\boldsymbol{\lambda})} \underbrace{\frac{\mathcal{N}(\mathbf{z}|0, \mathbf{I})}{\mathcal{N}(\mathbf{z}|\mathbf{m}, \mathbf{V})}}_{\tilde{p}_e(\mathbf{z}|\boldsymbol{\lambda})}.$$

We give further details about the bound for this case in the appendix.

Correlated Topic Model (CTM): Given a text document with a vocabulary of N words, denote its word-count vector by $\mathbf{y}$. Let K be the number of topics and $\mathbf{z}$ be the vector of topic-proportions. We can then use the following split:

$$\frac{p(\mathbf{y}, \mathbf{z})}{q(\mathbf{z}|\boldsymbol{\lambda})} = \underbrace{\prod_{n=1}^{N} \left[\sum_{k=1}^{K} \beta_{n,k} \frac{e^{z_k}}{\sum_j e^{z_j}}\right]^{y_n}}_{\tilde{p}_d(\mathbf{z}|\boldsymbol{\lambda})} \underbrace{\frac{\mathcal{N}(\mathbf{z}|\boldsymbol{\mu}, \boldsymbol{\Sigma})}{\mathcal{N}(\mathbf{z}|\mathbf{m}, \mathbf{V})}}_{\tilde{p}_e(\mathbf{z}|\boldsymbol{\lambda})},$$

where $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ are parameters of the Gaussian prior and $\beta_{n,k}$ are parameters of K multinomials. We give further details about the bound in the appendix.

3.2 STOCHASTIC-APPROXIMATION

The approach of Khan et al. (2015) considers (9) in the special case of (6) where we use the exact gradient $\nabla f(\boldsymbol{\lambda}_k)$ in the first term. But in practice this gradient is often difficult to compute. In our framework, we allow a stochastic approximation of $\nabla f(\boldsymbol{\lambda})$ which we denote by $\widehat{\nabla} f(\boldsymbol{\lambda}_k)$.

As shown in the previous section, f might take a form $f(\boldsymbol{\lambda}) := \Sigma_{n=1}^{N} \mathbb{E}_q[\tilde{f}_n(\mathbf{z})]$ for a set of functions $\tilde{f}_n$ as in the GP model (11). In some situations, $\mathbb{E}_q[\tilde{f}_n(\mathbf{z})]$ is computationally expensive or intractable. For example, in GP models the expectation is equal to $\mathbb{E}_q[\log p(y_n|z_n)]$, which is intractable for most non-Gaussian likelihoods. In such cases, we can form a stochastic approximation by using a few samples $\mathbf{z}^{(s)}$ from $q(\mathbf{z}|\boldsymbol{\lambda})$, as shown below:

$$\nabla \mathbb{E}_q[\tilde{f}_n(\mathbf{z})] \approx \widehat{\mathbf{g}}(\boldsymbol{\lambda}, \boldsymbol{\xi}_n) := \frac{1}{S} \sum_{s=1}^{S} \tilde{f}_n(\mathbf{z}^{(s)}) \nabla [\log q(\mathbf{z}^{(s)}|\boldsymbol{\lambda})]$$

where $\boldsymbol{\xi}_n$ represents the noise in the stochastic approximation $\widehat{\mathbf{g}}$ and we use the identity $\nabla q(\mathbf{z}|\boldsymbol{\lambda}) = q(\mathbf{z}|\boldsymbol{\lambda}) \nabla [\log q(\mathbf{z}|\boldsymbol{\lambda})]$ to derive the expression (Ranganath et al., 2013). We can then form a stochastic-gradient by randomly selecting a mini-batch of M functions $\tilde{f}_{n_i}(\mathbf{z})$ and employing the estimate

$$\widehat{\nabla} f(\boldsymbol{\lambda}) = \frac{N}{M} \sum_{i=1}^{M} \widehat{\mathbf{g}}(\boldsymbol{\lambda}, \boldsymbol{\xi}_{n_i}). \tag{12}$$

In our analysis we make the following two assumptions regarding the stochastic approximation of the gradient:

(A3) The estimate is unbiased: $\mathbb{E}[\widehat{\mathbf{g}}(\boldsymbol{\lambda}, \boldsymbol{\xi}_n)] = \bigtriangledown f(\boldsymbol{\lambda})$.

(A4) Its variance is upper bounded: $\text{Var}[\widehat{\mathbf{g}}(\boldsymbol{\lambda}, \boldsymbol{\xi}_n)] \leq \sigma^2$.

In both the assumptions, the expectation is taken with respect to the noise $\boldsymbol{\xi}_n$. The first assumption is true for the stochastic approximations of (12). The second assumption is stronger, but only needs to hold for all $\boldsymbol{\lambda}_k$ so is almost always satisfied in practice.

3.3 DIVERGENCE FUNCTIONS

To incorporate the geometry of q we incorporate a divergence function $\mathbb{D}$ between $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}_k$. The set of divergence functions need to satisfy two assumptions:

(A5) $\mathbb{D}(\boldsymbol{\lambda} \,\|\, \boldsymbol{\lambda}') > 0$, for all $\boldsymbol{\lambda} \neq \boldsymbol{\lambda}'$.

(A6) There exist an $\alpha > 0$ such that for all $\boldsymbol{\lambda}, \boldsymbol{\lambda}'$ generated by (9) we have:

$$(\boldsymbol{\lambda} - \boldsymbol{\lambda}')^T \nabla_{\boldsymbol{\lambda}} \mathbb{D}(\boldsymbol{\lambda} \,\|\, \boldsymbol{\lambda}') \geq \alpha \|\boldsymbol{\lambda} - \boldsymbol{\lambda}'\|^2. \qquad (13)$$

The first assumption is reasonable and is satisfied by typical divergence functions like the squared Euclidean distance and variants of the KL divergence. In the next section we show that, whenever the iteration (9) is defined and all $\boldsymbol{\lambda}_k$ stay within a compact set, the second assumption is satisfied for all divergence functions considered in Section 2.

4 SPECIAL CASES

Most methods discussed in Section 2 are special cases of the proposed iteration (9). We obtain gradient descent if $h = 0$, $f = -\underline{\mathcal{L}}$, $\widehat{\nabla} f = \nabla f$, and $\mathbb{D}(\boldsymbol{\lambda} \| \boldsymbol{\lambda}_k) = (1/2)\|\boldsymbol{\lambda} - \boldsymbol{\lambda}_k\|^2$ (in this case A6 is satisfied with $\alpha = 1$). From here, there are three standard generalizations in the optimization literature: SG methods do not require that $\widehat{\nabla} f = \nabla f$, proximal-gradient methods do not require that $h = 0$, and mirror descent allows $\mathbb{D}$ to be a different Bregman divergence generated by a strongly-convex function. Our analysis applies to all these variations on existing optimization algorithms because A1 to A5 are standard assumptions (Ghadimi et al., 2014) and, as we now show, A6 is satisfied for this class of Bregman divergences. In particular, consider the generic Bregman divergence shown in the left side of (8) for some strongly-convex function $A(\boldsymbol{\lambda})$. By taking the gradient with respect to $\boldsymbol{\lambda}$ and substituting in (13), we obtain that A6 is equivalent to

$$(\boldsymbol{\lambda} - \boldsymbol{\lambda}_k)^T [\bigtriangledown A(\boldsymbol{\lambda}) - \bigtriangledown A(\boldsymbol{\lambda}_k)] \geq \alpha \|\boldsymbol{\lambda} - \boldsymbol{\lambda}_k\|^2,$$

which is equivalent to strong-convexity of the function $A(\boldsymbol{\lambda})$ (Nesterov, 2004, Theorem 2.1.9).

The method of Theis & Hoffman (2015) corresponds to choosing $h = -\underline{\mathcal{L}}$, $f = 0$, and $\mathbb{D}(\boldsymbol{\lambda} || \boldsymbol{\lambda}_k) := \mathbb{D}_{KL}[q(\mathbf{z}|\boldsymbol{\lambda}) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda}_k)]$ where q is an exponential family distribution with natural parameters $\boldsymbol{\lambda}$. Since we assume h to be convex, only limited cases of their approach are covered under our framework. The method of Khan et al. (2015) also uses the KL divergence and focuses on the deterministic case where $\widehat{\nabla} f(\boldsymbol{\lambda}) = \nabla f(\boldsymbol{\lambda})$, but uses the split $-\underline{\mathcal{L}} = f + h$ to allow for non-conjugate models. In both of these models, A6 is satisfied when the Fisher matrix $\bigtriangledown^2 A(\boldsymbol{\lambda})$ is positive-definite. This can be shown by using the definition of the KL divergence for exponential families (Nielsen & Garcia, 2009):

$$\begin{aligned} &\mathbb{D}_{KL}[q(\mathbf{z}|\boldsymbol{\lambda}) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda}_k)] \\ &\quad := A(\boldsymbol{\lambda}_k) - A(\boldsymbol{\lambda}) - [\bigtriangledown A(\boldsymbol{\lambda})]^T (\boldsymbol{\lambda}_k - \boldsymbol{\lambda}). \end{aligned} \qquad (14)$$

Taking the derivative with respect to $\boldsymbol{\lambda}$ and substituting in (13) with $\boldsymbol{\lambda}' = \boldsymbol{\lambda}_k$, we get the condition

$$(\boldsymbol{\lambda} - \boldsymbol{\lambda}_k)^T [\bigtriangledown^2 A(\boldsymbol{\lambda})](\boldsymbol{\lambda} - \boldsymbol{\lambda}_k) \geq \alpha \|\boldsymbol{\lambda} - \boldsymbol{\lambda}_k\|^2,$$

which is satisfied when $\bigtriangledown^2 A(\boldsymbol{\lambda})$ is positive-definite over a compact set for α equal to its lowest eigenvalue on the set.

Methods based on natural-gradient using iteration (3) (like SVI) correspond to using $h = 0$, $f = -\underline{\mathcal{L}}$, and the symmetric KL divergence. Assumption A1 to A5 are usually assumed for these methods and, as we show next, A6 is also satisfied. In particular, when q is an exponential family distribution the symmetric KL divergence can be written as the sum of the Bregman divergence shown in (8) and the KL divergence shown in (14),

$$\begin{aligned} &\mathbb{D}^{sym}_{KL}[q(\mathbf{z}|\boldsymbol{\lambda}) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda}_k)] \\ &:= \mathbb{D}_{KL}[q(\mathbf{z}|\boldsymbol{\lambda}_k) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda})] + \mathbb{D}_{KL}[q(\mathbf{z}|\boldsymbol{\lambda}) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda}_k)] \\ &= \mathbb{D}_A(\boldsymbol{\lambda} \| \boldsymbol{\lambda}_k) + \mathbb{D}_{KL}[q(\mathbf{z}|\boldsymbol{\lambda}) \,\|\, q(\mathbf{z}|\boldsymbol{\lambda}_k)] \end{aligned}$$

where the first equality follows from the definition of the symmetric KL divergence and the second one follows from (8). Since the two divergences in the sum satisfy A6, the symmetric KL divergence also satisfies the assumption.

5 CONVERGENCE OF PG-SVI

We first analyze the convergence rate of deterministic methods where the gradient is exact, $\widehat{\nabla} f(\boldsymbol{\lambda}) = \nabla f(\boldsymbol{\lambda})$. This yields a simplified result that applies to a wide variety of existing variational methods. Subsequently, we consider the more general case where a stochastic approximation of the gradient is used.

5.1 DETERMINISTIC METHODS

The following theorem establishes the convergence under a fixed step-size. We use $C_0 = \underline{\mathcal{L}}^* - \underline{\mathcal{L}}(\boldsymbol{\lambda}_0)$ as the initial (constant) sub-optimality, and express our result in terms of the quantity $\|\boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k\|$.

Proposition 1. *Let A1, A2, A5, and A6 be satisfied. If we run t iterations of (9) with a fixed step-size $\beta_k = \alpha/L$ for all k and an exact gradient $\nabla f(\boldsymbol{\lambda})$, then we have*

$$\min_{k\in\{0,1,\ldots,t-1\}} \|\boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k\|^2 \leq \frac{2C_0}{\alpha t} \tag{15}$$

We give a proof in the appendix. Roughly, the theorem states that the minimum distance moved across all iterations must be in $O(1/t)$. If the objective is bounded below (C_0 is finite), then this result implies that the algorithm converges to a stationary point and also gives a rate of convergence.

Stating the result in terms of $\|\boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k\|$ may appear to be unconventional, but this quantity is useful since it characterizes a fixed point of the algorithm. For example, consider the special case of gradient descent where $h = 0$ and $\mathbb{D}(\boldsymbol{\lambda}, \boldsymbol{\lambda}_k) = \frac{1}{2}\|\boldsymbol{\lambda} - \boldsymbol{\lambda}_k\|^2$. In this case, $\alpha = 1$ and $\beta_k = 1/L$, therefore we have $\|\boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k\| = \|\nabla f(\boldsymbol{\lambda}_k)\|^2/L$ and Proposition 1 implies that $\min_k \|\nabla f(\boldsymbol{\lambda}_k)\|^2$ has a convergence rate of $O(1/t)$. This in turn shows that the method converges at a sublinear rate to an approximate stationary point, which would be a global minimum in the special case where f is convex.

If we use a divergence with $\alpha > 1$ then we can use a step-size larger than $1/L$ and the error will decrease faster than gradient-descent. To our knowledge, this is the first result that formally shows that natural-gradient methods can achieve faster convergence rates. The splitting of the objective into f and h functions is also likely to improve the step-size. Since L only depends on f, sometimes it might be possible to reduce the Lipschitz constant by choosing an appropriate split.

We next give a more general result that allows a per-iteration step size.

Proposition 2. *If we choose the step-sizes β_k to be such that $0 < \beta_k \leq 2\alpha/L$ with $\beta_k < 2\alpha/L$ for at least one k, then,*

$$\min_{k\in\{0,1\ldots t-1\}} \frac{1}{\beta_k}\|\boldsymbol{\lambda}_{k+1} - \boldsymbol{\lambda}_k\|^2 \leq \frac{C_0}{\sum_{k=0}^{t-1}(\alpha\beta_k - L\beta_k^2/2)} \tag{16}$$

We give a proof in the appendix. For gradient-descent, the above result implies that we can use any step-size less than $2/L$, which agrees with the classical step-size choices for gradient and proximal-gradient methods.

5.2 STOCHASTIC METHODS

We now give a bound for the more general case where we use a stochastic approximation of the gradient.

Proposition 3. *Let A1-A6 be satisfied. If we run t iterations of (9) for a fixed step-size $\beta_k = \gamma\alpha_*/L$ (where $0 < \gamma < 2$ is a scalar) and fixed batch-size $M_k = M$ for all k with a stochastic gradient $\widehat{\nabla} f(\boldsymbol{\lambda})$, then we have*

$$\mathbb{E}_{R,\boldsymbol{\xi}}(\|\boldsymbol{\lambda}_{R+1} - \boldsymbol{\lambda}_R\|^2) \leq \frac{1}{2-\gamma}\left[\frac{2C_0}{\alpha_* t} + \frac{\gamma c\sigma^2}{ML}\right].$$

where c is a constant such that $c > 1/(2\alpha)$ and $\alpha_* := \alpha - 1/(2c)$. The expectation is taken with respect to the noise $\boldsymbol{\xi} := \{\boldsymbol{\xi}_0, \boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{t-1}\}$, and a random variable R which follows the uniform distribution $Prob(R = k) = 1/t, \forall k \in \{0, 1, 2, \ldots, t-1\}$.

Unlike the bound of Proposition 1, this bound depends on the noise variance σ^2 as well the mini-batch size M. In particular, as we would expect, the bound gets tighter as the variance gets smaller and as the size of our mini-batch grows. Notice that we can also make the second term smaller by decreasing the value of γ and the first term smaller by increasing the number of iterations. Therefore, this bound indicates that a small enough constant step-size γ (or a sufficiently-large batch-size M) can be used to reach any target level of accuracy. In the appendix, we give a more general result that allows a per-iteration step-size which can used to give an "anytime" algorithm that is able to converge to an arbitrary level of accuracy by using a decreasing sequence of step sizes (but we found that constant step-sizes work better empirically). Note that while stating the result in terms of a randomized iteration might seem strange, in practice we typically just take the last iteration as the minimizer.

6 CLOSED-FORM UPDATES FOR NON-CONJUGATE MODELS

We now give an example where iteration (9) attains a closed-form solution. We expect such closed-form solution to exist for a large class of problems, including models where q is an exponential-family distribution, but here we focus on the GP model discussed in Section 3.1.

For the GP model, we rewrite the lower bound (11) as

$$-\underline{\mathcal{L}}(\mathbf{m}, \mathbf{V}) := \underbrace{\sum_{n=1}^{N} f_n(m_n, v_n)}_{f(\boldsymbol{m},\boldsymbol{V})} + \underbrace{\mathbb{D}_{KL}[q \,\|\, p]}_{h(\boldsymbol{m},\boldsymbol{V})} \tag{17}$$

where we've used $q := \mathcal{N}(\mathbf{z}|\mathbf{m}, \mathbf{V})$, $p := \mathcal{N}(\mathbf{z}|0, \mathbf{K})$, and $f_n(m_n, v_n) := -\mathbb{E}_q[\log p(y_n|z_n)]$ with m_n being the entry n of $\mathbf{m}$ and v_n being the diagonal entry n of $\mathbf{V}$. We can compute a stochastic approximation of f using (12) by randomly selecting an example n_k (choosing $M = 1$) and using a Monte Carlo gradient approximation of f_{n_k}. Using this approximation, the linearized term in (9) can be

simplified to the following:

$$\boldsymbol{\lambda}^T \left[\widehat{\nabla} f(\boldsymbol{\lambda}_k)\right] = m_n \underbrace{N[\nabla_{m_n} f_{n_k}(m_{n_k,k}, v_{n_k,k})]}_{:=\alpha_{n_k,k}} + v_n \underbrace{N[\nabla_{v_n} f_{n_k}(m_{n_k,k}, v_{n_k,k})]}_{:=2\,\gamma_{n_k,k}} = m_n \alpha_{n_k,k} + \tfrac{1}{2} v_n \gamma_{n_k,k} \quad (18)$$

where $m_{n_k,k}$ and $v_{n_k,k}$ denote the value of m_n and v_n in the k'th iteration for $n = n_k$. By using the KL divergence as our divergence function in iteration (9), and by denoting $\mathcal{N}(\mathbf{z}|\mathbf{m}_k, \mathbf{V}_k)$ by q_k, we can express the two last two terms in (9) as a single KL divergence function as shown below:

$$\begin{aligned}
&\boldsymbol{\lambda}^T \left[\widehat{\nabla} f(\boldsymbol{\lambda}_k)\right] + h(\boldsymbol{\lambda}) + \frac{1}{\beta_k}\mathbb{D}(\boldsymbol{\lambda}\|\boldsymbol{\lambda}_k),\\
&= (m_n \alpha_{n,k} + \tfrac{1}{2} v_n \gamma_{n,k}) + \mathbb{D}_{KL}[q \,\|\, p] + \frac{1}{\beta_k}\mathbb{D}_{KL}[q \,\|\, q_k],\\
&= (m_n \alpha_{n,k} + \tfrac{1}{2} v_n \gamma_{n,k}) + \frac{1}{1-r_k}\mathbb{D}_{KL}[q \,\|\, p^{1-r_k} q_k^{r_k}],
\end{aligned}$$

where $r_k := 1/(1+\beta_k)$. Comparing this to (17), we see that this objective is similar to that of a GP model with a Gaussian prior[1] $p^{1-r_k} q_k^{r_k}$ and a linear Gaussian-like log-likelihood. Therefore, we can obtain closed-form updates for its minimization.

The updates are shown below and a detailed derivation is given in the appendix.

$$\begin{aligned}
\widetilde{\boldsymbol{\gamma}}_k &= r_k \widetilde{\boldsymbol{\gamma}}_{k-1} + (1-r_k)\gamma_{n_k,k}\mathbf{1}_{n_k},\\
\mathbf{m}_{k+1} &= \mathbf{m}_k - (1-r_k)(\mathbf{I} - \mathbf{K}\mathbf{A}_k^{-1})(\mathbf{m}_k + \alpha_{n_k,k}\boldsymbol{\kappa}_{n_k}),\\
v_{n_{k+1},k+1} &= \kappa_{n_{k+1},n_{k+1}} - \boldsymbol{\kappa}_{n_{k+1}}^T \mathbf{A}_k^{-1} \boldsymbol{\kappa}_{n_{k+1}}, \qquad (19)
\end{aligned}$$

where $\widetilde{\boldsymbol{\gamma}}_0$ is initialized to a small positive constant to avoid numerical issues, $\mathbf{1}_{n_k}$ is a vector with all zero entries except n_k'th entry which is equal to 1, $\boldsymbol{\kappa}_k$ is n_k'th column of $\mathbf{K}$, and $\mathbf{A}_k := \mathbf{K} + [\text{diag}(\widetilde{\boldsymbol{\gamma}}_k)]^{-1}$. For iteration $k+1$, we use $m_{n_{k+1},k+1}$ and $v_{n_{k+1},k+1}$ to compute the gradients $\alpha_{n_{k+1},k+1}$ and $\gamma_{n_{k+1},k+1}$, and run the above updates again. We continue until a convergence criteria is reached.

There are numerous advantages of these updates. First, We do not need to store the full covariance matrix $\mathbf{V}$. The updates avoid forming the matrix and only update $\mathbf{m}$. This works because we only need one diagonal element in each iteration to compute the stochastic gradient $\gamma_{n_k,k}$. For large N this is a clear advantage since the memory cost is $O(N)$ rather than $O(N^2)$. Second, computation of the mean vector $\mathbf{m}$ and a diagonal entry of $\mathbf{V}$ only require solving two linear equations, as shown in the second and third line of (19). In general, for a mini-batch of size M, we need a total of $2M$ linear equations, which is a lot cheaper than an explicit inversion. Finally, the linear equations at iteration $k+1$ are very similar to those at iteration k, since $\mathbf{A}_k$ differ only at one entry from $\mathbf{A}_{k+1}$. Therefore, we can reuse computations from the previous iteration to improve the computational efficiency of the updates.

[1] Since p and q are Gaussian, the product is a Gaussian.

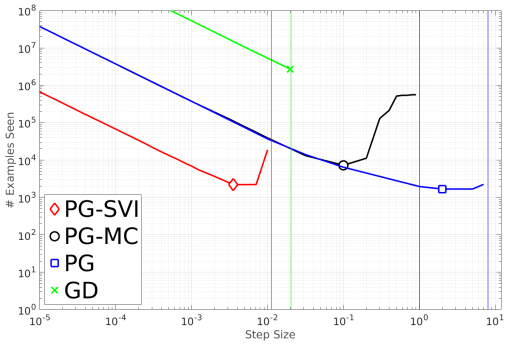

Figure 1: We show the number of examples required for convergence versus fixed step-sizes for binary GP classification. Methods based on proximal-gradient require fewer number of examples compared to gradient descent (GD). Step-size is upper bounded for all methods (upper bound shown with vertical lines). PG-SVI surprisingly converges at the same rate as PG.

7 EXPERIMENTAL RESULTS

In this section, we compare our method to many existing approaches such as SGD and four adaptive gradient-methods (ADAGRAD, ADADELTA, RMSprop, ADAM), as well as two variational inference methods for non-conjugate models (the Delta method and Laplace method). We show results on Gaussian process classification (Kuss & Rasmussen, 2005) and correlated topic models (Blei & Lafferty, 2007). The code to reproduce these experiments can be found at this link[2].

7.1 GAUSSIAN PROCESS CLASSIFICATION

We consider binary classification by using a GP model with Bernoulli-logit likelihood on three datasets: Sonar, Ionosphere, and USPS-3vs5. These datasets can be found at the UCI data repository[3] and their details are discussed in Kuss & Rasmussen (2005). For the GP prior, we use the zero mean-function, and a squared-exponential covariance function with hyperparameters σ and l as defined in Kuss & Rasmussen (2005) (see Eq. 33). We set the values of the hyperparameters using cross-validation. For the three datasets, the hyperparameters $(\log l, \log \sigma)$ are set to $(-1, 6)$, $(1, 2.5)$, and $(2.5, 5)$, respectively.

[2] https://github.com/emtiyaz/prox-grad-svi

[3] https://archive.ics.uci.edu/ml/datasets.html

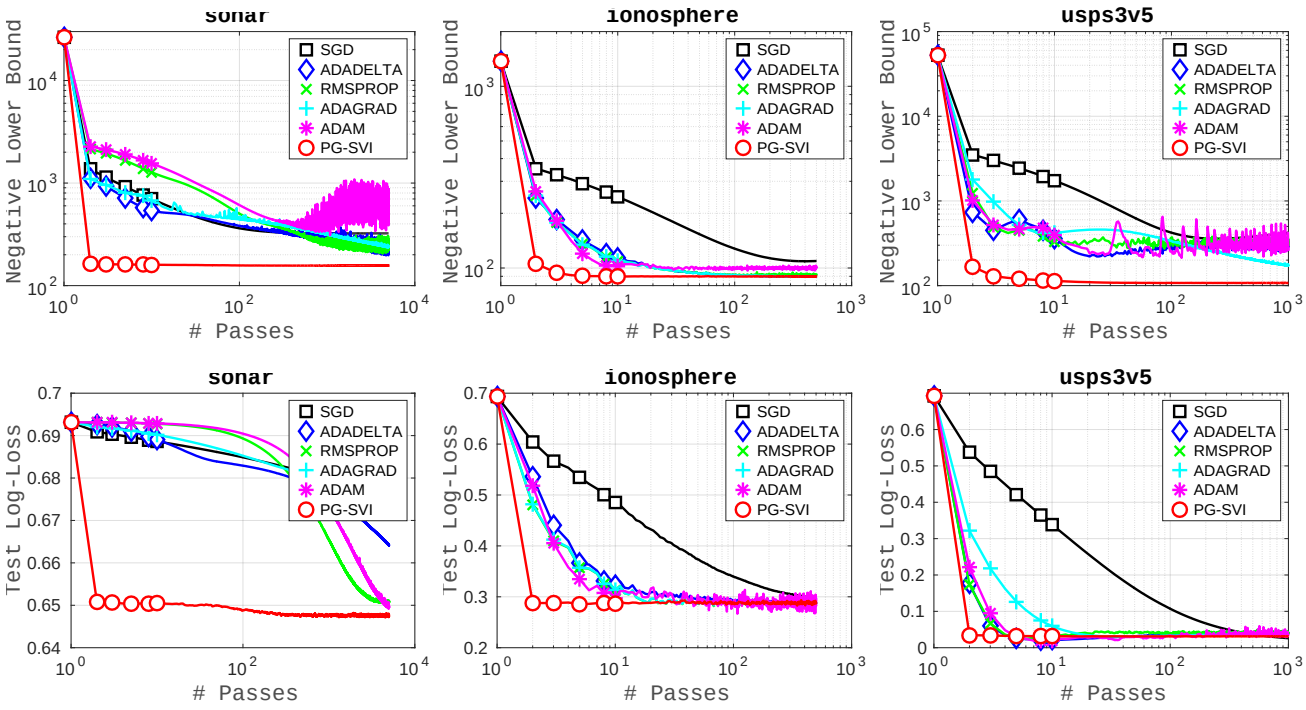

Figure 2: Comparison with adaptive gradient methods for binary classification using GP. We compare PG-SVI to SGD, ADADELTA, RMSprop, ADAGRAD, and ADAM on three datasets. Each column shows results for a dataset. Top row shows the negative of the lower bound, while the bottom row shows the test log-loss. In each plot, the X-axis shows the number of passes made through the data. Markers are shown at 0, 1, 2, 4, 7, and 9 passes through the data. Our method always converges within 10 passes through the data, while other methods more than 100 passes.

7.1.1 Performance Under a Fixed Step-Size

In our first experiment, we compare the performance under a fixed step-size. These results demonstrate that the step-size required for convergence is upper-bounded, as shown in our convergence analysis. The results also demonstrate the faster convergence of our method compared to gradient-descent methods. We compare the following four algorithms on the Ionosphere dataset: (1) batch gradient-descent (referred to as 'GD'), (2) batch proximal-gradient algorithm (referred to as 'PG'), (3) batch version of proximal-gradient algorithm with gradients approximated by using Monte-Carlo (referred to as 'PG-MC'), and (4) proposed proximal-gradient stochastic variational-inference (referred to as 'PG-SVI') where stochastic gradients are obtained using (12) with $M = 5$. For Monte Carlo approximation, we use $S = 500$ samples.

Figure 1 shows the number of examples required for convergence versus the step-size. A lower number implies faster convergence. The vertical lines show the step-size above which a method diverges. Convergence is assessed by monitoring the lower bound, and when the change in consecutive iterations do not exceed a certain threshold, we stop the algorithm.

We clearly see that GD requires many more passes through the data, and methods based on proximal-gradient method converge faster than GD. In addition, the upper bound on the step-size for PG is much larger than GD. This implies that PG can potentially take larger steps than the GD method. PG-SVI is surprisingly as fast as PG which shows the advantage of our approach over the approach of Khan et al. (2015).

7.1.2 Comparison with Adaptive Gradient Methods

We also compare PG-SVI to SGD and four adaptive methods, namely ADADELTA (Zeiler, 2012), RMSprop (Tieleman & Hinton, 2012), ADAGRAD (Duchi et al., 2011), and ADAM (Kingma & Ba, 2014). The implementation details of these algorithms are given in the appendix. We compare the value of the lower bound versus number of passes through the data. We also compare the average log-loss on the test data: $-\sum_n \log \hat{p}_n / N_*$ where $\hat{p}_n = p(y_n|\sigma, l, \mathcal{D}_t)$ is the predictive probabilities of the test point y_n given training data $\mathcal{D}_t$ and N_* is the total number of test-pairs. A lower value is better for the log-loss, and a value of 1 is equal to the performance of random coin-flipping.

Figure 2 summarizes the results. Each column shows results for a dataset. The top row shows the negative of the lower bound, while the bottom row shows the test log-loss.

Lower values are better for both. In all plots, the X-axis shows the number of passes made through the data. Markers are shown at 0, 1, 2, 4, 7, and 9 passes through the data (one pass means the number of randomly selected examples is equal to the total number of examples). Our method is much faster to converge than other methods, and it always converges within 10 passes through the data, while other methods requires more than 100 passes.

7.2 CORRELATED TOPIC MODEL

We now show results for correlated topic model on two collections of documents, namely NIPS dataset and Associated Press (AP) dataset. The NIPS[4] dataset contains 1500 documents from the NIPS conferences held between 1987 and 1999 (a vocabulary-size of 12,419 words and a total of around 1.9M words). The AP[5] collection contains 2,246 documents from the Associated Press (a vocabulary-size of 10,473 words and a total of 436K observed words). We use 50% of the documents for training and 50% for testing.

We compare to the Delta method and the Laplace method discussed in Wang & Blei (2013), and also to the original mean-field (MF) method of Blei & Lafferty (2007). For these methods, we use the implementation available at this link[6]. All of these methods approximate the lower bound by using approximations to the expectation of log-sum-exp functions (see Appendix for details). We compare these methods to the two versions of our algorithm which do not use such approximations, rather use a stochastic gradient as explain in Section 3.2. Specifically, we use the following two versions: one with full covariance (referred to as PG-SVI), and the other with diagonal covariance (referred to as PG-SVI-MF). For both of these algorithms, we use a fixed step-size of 0.001, and a mini-batch size of 2 documents.

Following Wang & Blei (2013), we compare the held-out log-likelihood, which is computed as follows: a new test document $\mathbf{y}$ is split into two halves $(\mathbf{y}^1, \mathbf{y}^2)$, then we compute the approximate posterior $q(\mathbf{z})$ to the posterior $p(\mathbf{z}|\mathbf{y}^1)$ using which we compute the held-out log-likelihood for each $y_n \in \mathbf{y}^2$ as follows:

$$\log p(y_n) \approx \log \int_z \left[\sum_{k=1}^{K} \beta_{n,k} \frac{e^{z_k}}{\sum_j e^{z_j}}\right]^{y_n} q(\mathbf{z}) d\mathbf{z} \quad (20)$$

We use Monte Carlo to approximate this quantity by using a large number of samples from q (unlike Wang & Blei (2013) who approximate it by using the Delta method). We report the average of this quantity over all words in $\mathbf{y}^2$.

Figure 3 shows the negative of the average held-out log-likelihood versus time for 10 topics. Lower values are better. Markers are shown at iterations after second and fifth passes through the data. We see that methods based on proximal-gradient algorithm converge a little bit faster than the existing methods. More importantly, they achieves better performance. This could be due to the fact that we do not approximate the expectation of the log-sum-exp function, unlike the Delta and Laplace method. We obtained similar results for different number of topics.

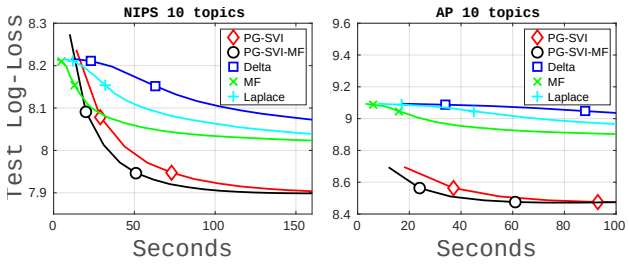

Figure 3: Results on NIPS (left) and AP (right) datasets using correlated topic model with 10 topics. We plot the negative of the average held-out log-likelihood versus time. Lower values are better. Methods based on proximal-gradient algorithm perform better.

8 DISCUSSION

This work has made two contributions. First, we proposed a new variational inference method that combines variable splitting, stochastic gradients, and general divergence functions. This method is well-suited for a huge variety of the variational inference problems that arise in practice, and we anticipate that it may improve over state of the art methods in a variety of settings. Our second contribution is a theoretical analysis of the convergence rate of this general method. Our analysis generalizes existing results for the mirror descent algorithm in optimization, and resolves the convergences of a variety of existing variational inference methods. Due to its generality we expect that this analysis could be useful to establish convergence of other algorithms that we have not thought of, perhaps beyond the variational inference settings we consider in this work.

One issue that we have not satisfactorily resolved is giving a theoretically-justified way to set the step-size in practice; our analysis only indicates that it must be sufficiently small. However, this problem is common in many methods in the literature and our analysis at least suggests the factors that should be taken into account. Another open issue is the applicability our method to many other latent variable models; in this paper we have shown applications to variational-Gaussian inference, but we expect that our method should result in simple updates for a larger class of latent variable models such as non-conjugate exponential family distribution models. Additional work on these issues will improve usability of our method.

[4] https://archive.ics.uci.edu/

[5] http://www.cs.columbia.edu/~blei/lda-c/index.html

[6] https://www.cs.princeton.edu/~chongw/resource.html

References

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

Beck, Amir and Teboulle, Marc. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

Beck, Amir and Teboulle, Marc. A fast iterative shrinkage-thresholding algorithm with application to wavelet-based image deblurring. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 693–696, 2009.

Blei, David M and Lafferty, John D. A correlated topic model of science. *The Annals of Applied Statistics*, pp. 17–35, 2007.

Duchi, John, Hazan, Elad, and Singer, Yoram. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159, 2011.

Duchi, John C, Shalev-Shwartz, Shai, Singer, Yoram, and Tewari, Ambuj. Composite objective mirror descent. *COLT*, 2010.

Ghadimi, Saeed, Lan, Guanghui, and Zhang, Hongchao. Mini-batch stochastic approximation methods for non-convex stochastic composite optimization. *Mathematical Programming*, pp. 1–39, 2014.

Hoffman, Matthew D, Blei, David M, Wang, Chong, and Paisley, John. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

Honkela, A., Raiko, T., Kuusela, M., Tornio, M., and Karhunen, J. Approximate Riemannian conjugate gradient learning for fixed-form variational Bayes. *Journal of Machine Learning Research*, 11:3235–3268, 2011.

Khan, Mohammad Emtiyaz, Baque, Pierre, Flueret, Francois, and Fua, Pascal. Kullback-Leibler Proximal Variational Inference. In *Advances in Neural Information Processing Systems*, 2015.

Kingma, Diederik and Ba, Jimmy. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kucukelbir, Alp, Ranganath, Rajesh, Gelman, Andrew, and Blei, David. Fully automatic variational inference of differentiable probability models. In *NIPS Workshop on Probabilistic Programming*, 2014.

Kuss, M. and Rasmussen, C. Assessing approximate inference for binary Gaussian process classification. *Journal of Machine Learning Research*, 6:1679–1704, 2005.

Mnih, Andriy and Gregor, Karol. Neural variational inference and learning in belief networks. *arXiv preprint arXiv:1402.0030*, 2014.

Nelder, John A and Baker, R Jacob. Generalized linear models. *Encyclopedia of Statistical Sciences*, 1972.

Nesterov, Y. *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2004.

Nielsen, Frank and Garcia, Vincent. Statistical exponential families: A digest with flash cards. *arXiv preprint arXiv:0911.4863*, 2009.

Paquet, Ulrich. On the convergence of stochastic variational inference in Bayesian networks. *NIPS Workshop on variational inference*, 2014.

Pascanu, Razvan and Bengio, Yoshua. Revisiting natural gradient for deep networks. *arXiv preprint arXiv:1301.3584*, 2013.

Ranganath, Rajesh, Gerrish, Sean, and Blei, David M. Black box variational inference. *arXiv preprint arXiv:1401.0118*, 2013.

Ravikumar, Pradeep, Agarwal, Alekh, and Wainwright, Martin J. Message-passing for graph-structured linear programs: Proximal methods and rounding schemes. *The Journal of Machine Learning Research*, 11:1043–1080, 2010.

Salimans, Tim, Knowles, David A, et al. Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Analysis*, 8(4):837–882, 2013.

Theis, Lucas and Hoffman, Matthew D. A trust-region method for stochastic variational inference with applications to streaming data. *arXiv preprint arXiv:1505.07649*, 2015.

Tieleman, Tijmen and Hinton, Geoffrey. Lecture 6.5-RMSprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning 4*, 2012.

Titsias, Michalis and Lázaro-Gredilla, Miguel. Doubly stochastic variational Bayes for non-conjugate inference. In *International Conference on Machine Learning*, 2014.

Wang, Chong and Blei, David M. Variational inference in nonconjugate models. *Journal of Machine Learning Research*, 14(1):1005–1031, 2013.

Zeiler, Matthew D. ADADELTA: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.