Fully Automatic Variational Inference of Differentiable Probability Models

Alp Kucukelbir
Data Science Institute
Department of Computer Science
Columbia University
alp@cs.columbia.edu

Rajesh Ranganath
Department of Computer Science
Princeton University
rajeshr@cs.princeton.edu

Andrew Gelman
Data Science Institute
Departments of Statistics, Political Science
Columbia University
gelman@stat.columbia.edu

David M. Blei
Data Science Institute
Departments of Computer Science, Statistics
Columbia University
david.blei@columbia.edu

Abstract

We describe an automatic variational inference method for approximating the posterior of differentiable probability models. Automatic means that the statistician only needs to define a model; the method forms a variational approximation, computes gradients using automatic differentiation and approximates expectations via Monte Carlo integration. Stochastic gradient ascent optimizes the variational objective function to a local maximum. We present an empirical study that applies hierarchical linear and logistic regression models to simulated and real data.

1 Introduction

Statistical inference studies the mechanism that gives rise to a set of random variable observations $\mathbf{X}$. The mechanism is unknown, so we propose a probabilistic model $p(\mathbf{X}, \mathbf{Z})$; it describes the data with latent variables $\mathbf{Z}$. The core computational challenge of inference is computing the posterior distribution $p(\mathbf{Z} \mid \mathbf{X})$ of the latent variables conditioned on the observed dataset.

Complex models present intractable posterior densities. Variational inference approximates the posterior with a simpler parameterized class of functions $q(\mathbf{Z}; \boldsymbol{\phi})$. Inference becomes an optimization problem that requires computing expectations of the model under the variational family. Analytic forms for these expectations are only available for a small class of models.

We propose a new variational inference algorithm for models with differentiable likelihoods. These are the models supported by the Stan probabilistic modeling language (Stan Development Team, 2014). We call our method variational Bayes in Stan (VBSTAN).[1] In VBSTAN, the statistician writes a model. The Stan compiler transforms any constrained variables (e.g., a positive variance term) into an unconstrained space, where we posit a Gaussian variational family. We approximate expectations via Monte Carlo integration and use automatic differentiation (AD) to compute gradients of the model. Stochastic gradient ascent maximizes the variational objective function using these noisy gradients.

We present a preliminary study using two hierarchical regression models: Bayesian linear regression with automatic relevance determination (ARD) (Murphy, 2012) and multi-level logistic regression (Gelman and Hill, 2006). We study the convergence of VBSTAN on the former model, and accuracy and speed on the latter, applied to polling data from the 1988 presidential election.

[1] VBSTAN is in active development at `https://github.com/stan-dev/stan/tree/feature/bbvb`.

Related work. Titsias and Lázaro-Gredilla (2014) propose a variational inference algorithm which include the Gaussian variational family, but cannot deal with constrained latent variables. Ranganath et al. (2014) develop a more general technique, but posit specific variational forms for different models. Salimans and Knowles (2014) present a stochastic linear regression perspective, but also rely on specifying a variational approximation. Kingma and Welling (2013); Rezende et al. (2014) derive variational inference algorithms based on gradients of the likelihood, but do not provide a generic way to determine the variational approximation. Wingate and Weber (2013) cast these ideas to probabilistic programs.

2 Automatic Variational Inference in Differentiable Models

Let $p(\mathbf{X}, \mathbf{Z})$ be a differentiable joint density with respect to $\mathbf{Z}$; the observations are $\mathbf{X}$, the latent variables $\mathbf{Z}$ are of dimension K. The posterior $p(\mathbf{Z} \mid \mathbf{X})$ describes the latent variables conditioned on the data. Variational inference minimizes the Kullback-Leibler (KL) divergence from an approximating variational family $q(\mathbf{Z}; \phi)$ with parameters ϕ to the posterior density $p(\mathbf{Z} \mid \mathbf{X})$.

Constrained latent variables. The latent variables may have constrained support. Denote the support of $\mathbf{z}$ as $\text{supp}(\mathbf{z})$. We first transform the support to the real coordinate space $\mathbb{R}^K$. Define a one-to-one function $f : \text{supp}(\mathbf{z}) \to \mathbb{R}^K$ such that the transformed variables $\widetilde{\mathbf{z}} = f(\mathbf{z})$ have support on $\mathbb{R}^K$. The unconstrained model becomes

$$p\left(\mathbf{X}, \mathbf{Z}\right) = p\left(\mathbf{X}, f^{-1}(\widetilde{\mathbf{Z}})\right) \left|\det J_{f^{-1}}(\widetilde{\mathbf{Z}})\right|,$$

where $J_{f^{-1}}(\widetilde{\mathbf{Z}})$ is the Jacobian of the inverse of f.

Variational family. We then posit a Gaussian variational family $q(\widetilde{\mathbf{Z}}; \mu, \Sigma)$ parameterized by mean vector $\mu \in \mathbb{R}^K$ and covariance matrix $\Sigma \in \mathbb{R}^{(K \times K)}$. The problem of minimizing the KL divergence is equivalent to maximizing the evidence lower bound (ELBO) (Jordan et al., 1999). We write the ELBO in the unconstrained space as

$$\begin{aligned}\mathscr{L}(\mu, \Sigma) &= \int q(\widetilde{\mathbf{Z}}; \mu, \Sigma) \log \left(\frac{p\left(\mathbf{X}, f^{-1}(\widetilde{\mathbf{Z}})\right) \left|\det J_{f^{-1}}(\widetilde{\mathbf{Z}})\right|}{q(\widetilde{\mathbf{Z}}; \mu, \Sigma)} \right) \mathrm{d}\widetilde{\mathbf{Z}} \\ &= \mathbb{E}_{q(\widetilde{\mathbf{Z}}; \mu, \Sigma)} \left[\log p\left(\mathbf{X}, f^{-1}(\widetilde{\mathbf{Z}})\right) + \log \left|\det J_{f^{-1}}(\widetilde{\mathbf{Z}})\right| \right] - \mathbb{E}_{q(\widetilde{\mathbf{Z}}; \mu, \Sigma)} \left[\log q(\widetilde{\mathbf{Z}}; \mu, \Sigma) \right]. \quad (1)\end{aligned}$$

The ELBO is a function of the variational parameters (μ, Σ). The second term is the entropy of a multivariate Gaussian, which has an analytic expression. This optimization problem is equivalent to proposing a transformed multivariate Gaussian as the variational family. To maximize the ELBO we use its gradients with respect to (μ, Σ). These gradients depend on the form of the variational family. We consider two cases for $q(\widetilde{\mathbf{Z}}; \mu, \Sigma)$: full-covariance ($\Sigma$ is full-rank) and mean-field (Σ is diagonal).

Full-covariance approximation. Define the following affine transformation of the variational family $\check{\mathbf{z}} = L^{-1}(\widetilde{\mathbf{z}} - \mu)$ where L is the lower-triangular Cholesky factor of the covariance matrix $\Sigma = LL^\top$. This standardizes the variational distribution as $\check{\mathbf{z}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The first expectation in Equation (1) discards its dependency on the variational parameters. Thus, the ELBO becomes

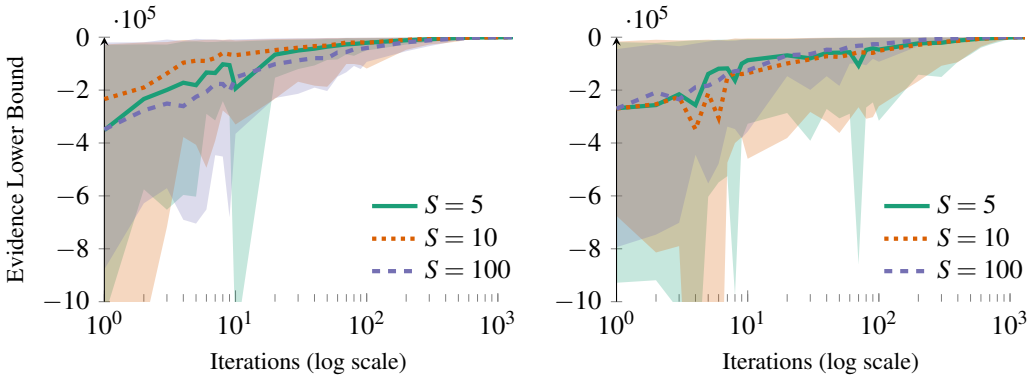
$$\begin{aligned}\mathscr{L}(\mu, L) = \mathbb{E}_{q(\check{\mathbf{Z}}; \mathbf{0}, \mathbf{I})} &\left[\log p\left(\mathbf{X}, f^{-1}(L\check{\mathbf{Z}} + \mu)\right) + \log \left|\det J_{f^{-1}}(L\check{\mathbf{Z}} + \mu)\right| \right] \\ &+ \frac{1}{2} K(1 + \log 2\pi) + \sum_k \log \left|L_{kk}\right|,\end{aligned}$$

where the last two terms are from the entropy. The index $k \in \{1, \ldots, K\}$ goes along the diagonal of L.

To compute the gradient of the ELBO, we exchange derivatives and expectations. We use the AD library in Stan and Monte Carlo integration to compute the model-specific gradient. The gradient with respect to μ is

$$\nabla_\mu \mathscr{L}(\mu, L) = \mathbb{E}_{q(\check{\mathbf{Z}}; \mathbf{0}, \mathbf{I})} \left[\texttt{stan::model::gradient}(L\check{\mathbf{Z}}_s + \mu) \right]. \quad (2)$$

We use S samples $\check{\mathbf{Z}}_s \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ to compute a noisy estimate of this expectation. The Stan function `stan::model::gradient` accounts for the Jacobian term via the chain rule of differentiation.



(a) Full-covariance **(b)** Mean-field

Figure 1: Convergence of VBSTAN using Bayesian linear regression with ARD. Lines and shaded areas represent the mean and min/max of 10 runs per configuration. ELBO computed using $S = 1000$.

The noisy gradient for an entry in L is

$$\nabla_{L_{ij}} \mathscr{L}(\mu, L) \approx \frac{1}{S} \sum_{s=1}^{S} [\texttt{stan::model::gradient}(\widetilde{\mathbf{Z}}_s)_i \cdot (\check{\mathbf{Z}}_s)_j] + L_{ij}^{-1} \mathbf{1}_{i=j}, \tag{3}$$

where the indices (i, j) traverse the rows and columns of a lower triangular square matrix of size K. The entropy term only contributes along the diagonal, as denoted by the indicator function $\mathbf{1}_{i=j}$.

The Cholesky decomposition is not unique for positive semidefinite matrices. (Consider sign permutations.) Requiring the diagonal of L to be positive would enforce uniqueness. However, the standard decomposition yields a simpler optimization routine.

Mean-field approximation. Define the following affine transformation of the variational family $\check{\mathbf{z}} = \text{diag}(\sigma^{-1})(\widetilde{\mathbf{z}} - \mu)$ where σ is the vector of standard deviations such that $\Sigma = \text{diag}(\sigma^2)$. This similarly standardizes the variational distribution; the entropy term in the ELBO becomes $\sum_k \log \sigma_k$.

The gradient with respect to μ is the same as in Equation (2). The gradient with respect to the standard deviations has a subtlety: we must ensure positivity. To that end, define $\widetilde{\sigma} = \log \sigma$, applied element-wise. The support of $\widetilde{\sigma}_k$ is now the real line, and we can write the standardized latent variables as $\check{\mathbf{z}} = \text{diag}(\exp(\widetilde{\sigma})^{-1})(\widetilde{\mathbf{z}} - \mu)$. The gradient for an entry of $\widetilde{\sigma}$ is similar to Equation (3),

$$\nabla_{\widetilde{\sigma}_k} \mathscr{L}(\mu, \widetilde{\sigma}) \approx \frac{1}{S} \sum_{s=1}^{S} [\texttt{stan::model::gradient}(\widetilde{\mathbf{Z}}_s)_k \cdot (\check{\mathbf{Z}}_s)_k \cdot \exp(\widetilde{\sigma})_k] + 1. \tag{4}$$

Stochastic gradient ascent. The gradients in Equations (2) to (4) are all noisy approximations of the true gradients of the ELBO. We use these gradients in a stochastic gradient ascent algorithm with an adaptive learning rate (Tieleman and Hinton, 2012).

3 Empirical Study

First, we investigate Bayesian linear regression with ARD. We simulate a dataset with 10 regressors and 100 observations. Figure 1 shows the convergence of VBSTAN as the number of Monte Carlo integration terms S vary. Both algorithms succeed at optimizing the ELBO to a local maximum.

Second, we study hierarchical logistic regression with polling data. We estimate a single outcome (probability of voting Republican) from a CBS news dataset, which has 11,566 responses from the week before the 1988 presidential election. There are two predictors for gender and race. Each state receives its own intercept, distributed according to a Gaussian with mean μ_α and standard deviation σ_α. (See Chapter 14.1 of Gelman and Hill, 2006, for more details.)

Table 1 shows the posterior means and standard deviations of the latent variables. All algorithms estimate similar values for the regressors β, but mean-field VBSTAN reports notably larger standard

Table 1: Accuracy (means and standard deviations) and speed of VBSTAN using hierarchical logistic regression. "Sampling" refers to Stan's Hamiltonian Monte Carlo algorithm. Both VBSTAN algorithms use $S = 10$ samples.

	Sampling	**Full-covariance**	**Mean-field**
β^{female}	-1.8 (8.6×10^{-2})	-1.8 (1.9×10^{-2})	-1.8 (8.1×10^{-2})
β^{black}	-0.1 (4×10^{-2})	-0.1 (6.3×10^{-2})	-0.1 (2.5×10^{-2})
μ_α	0.4 (7.3×10^{-2})	0.4 (2.7×10^{-2})	0.4 (1.6×10^{-1})
σ_α	0.4 (6.1×10^{-2})	0.5 (2.3×10^{-2})	0.6 (4.7×10^{-1})
runtime	$\sim$ 30s (default params)	$\sim$ 15s (1,000 iters)	$\sim$ 12s (1,000 iters)

deviations for the mean and standard deviation of the states (μ_α, σ_α). Preliminary timing measurements indicate that VBSTAN is faster than sampling for a dataset of this size; both algorithms converged before 1,000 iterations, by visual inspection of the ELBO (not shown).

4 Conclusion and Future Work

VBSTAN is an automatic variational algorithm for statistical inference of differentiable probability models. The variational approximation is a transformed Gaussian family implicitly defined on the constrained space of latent variables. Stochastic optimization with Monte Carlo integration circumvents the need to derive any of the expressions typically required for variational inference.

There are theoretical and practical directions for future work. Natural and higher-order gradients should lead to faster algorithms. Alternatives to the Gaussian approximation could increase VBSTAN's accuracy with heavy- or light-tailed likelihoods. High-dimensional models with many latent variables could benefit from sparse covariance estimation. Assessing convergence using noisy estimates of the ELBO will be important in practice. Data sub-sampling should scale VBSTAN to massive datasets.

References

Gelman, A. and Hill, J. (2006). *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Murphy, K. P. (2012). *Machine Learning: a Probabilistic Perspective*. MIT Press.

Ranganath, R., Gerrish, S., and Blei, D. (2014). Black box variational inference. In *Artificial Intelligence and Statistics*, pages 814–822.

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, pages 1278–1286.

Salimans, T. and Knowles, D. A. (2014). On using control variates with stochastic approximation for variational Bayes and its connection to stochastic linear regression. *arXiv preprint arXiv:1401.1022*.

Stan Development Team (2014). Stan: A C++ library for probability and sampling, version 2.5.0.

Tieleman, T. and Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4.

Titsias, M. and Lázaro-Gredilla, M. (2014). Doubly stochastic variational Bayes for non-conjugate inference. In *International Conference on Machine Learning (ICML-14)*, pages 1971–1979.

Wingate, D. and Weber, T. (2013). Automated variational inference in probabilistic programming. *arXiv preprint arXiv:1301.1299*.