Variational Boosting: Iteratively Refining Posterior Approximations

Andrew C. Miller
Harvard University
Cambridge, MA 02143
acm@seas.harvard.edu

Nicholas Foti
University of Washington
Seattle, WA 98105
nfoti@uw.edu

Ryan P. Adams
Harvard University
Cambridge, MA 02143
rpa@seas.harvard.edu

Abstract

We present a black-box variational inference (BBVI) method to approximate intractable posterior distributions with an increasingly rich approximating class. Using mixture distributions as the approximating class, we first describe how to apply the re-parameterization trick and existing BBVI methods to mixtures. We then describe a method, termed *Variational Boosting*, that iteratively refines an existing approximation by defining and solving a sequence of optimization problems, allowing the practitioner to trade computation time for increased accuracy.

1 Introduction

Variational inference (VI) [7, 14, 1] is a family of methods designed to approximate an intractable target distribution (typically known only up to a constant) with a *tractable surrogate distribution*. VI procedures typically minimize the Kullback-Leibler (KL) divergence of the approximation to the target by maximizing an appropriately defined tractable objective. Often, the class of approximating distributions is fixed, and typically excludes the target distribution (and its neighbors), which prevents the variational approximation from becoming arbitrarily close to the true posterior.

Markov chain Monte Carlo (MCMC), an alternative class of inference methods, approximate target distributions with samples drawn from a Markov chain constructed to admit the target distribution at each marginal. MCMC methods allow a user to trade computation time for increased accuracy — drawing more samples will make the approximation closer to the true target distribution. However, MCMC algorithms typically must be run iteratively, making them difficult to parallelize. Furthermore, correctly specifying MCMC moves can be more algorithmically restrictive than optimizing an objective (e.g. data subsampling in stochastic gradient methods).

We propose a variational inference method that gradually allows the approximation to become more and more complex, affording the practitioner a trade-off between time and accuracy. Our method builds on black-box variational inference methods using the *re-parameterization trick* [13, 8, 10], applicable to a very general class of target distributions.

Variational Inference Given a *target distribution* with density[1] $\pi(x)$ for a multivariate random variable $x \in \mathcal{X}$, variational inference approximates $\pi(x)$ with a *tractable approximate* distribution,[2] $q(x;\lambda)$, from which we can draw samples and form sample-based estimates of functions of x (e.g. posterior credible intervals, Bayesian predictions, etc.). Variational methods typically minimize $KL(q_\lambda||\pi)$ between the approximation and the target as a function of variational parameters λ. This is framed as an optimization problem — we optimize the KL objective (or an equivalent one) with

[1] We assume $\pi(x)$ is known up to a constant, $\tilde{\pi}(x) = \mathcal{C}\pi(x)$ for some constant $\mathcal{C}$; we may omit $\sim$ simplicity.

[2] We treat the density function as a synecdoche for the entire law, and use $q(x;\lambda)$ and $q_\lambda(x)$ interchangeably at the risk of slight notational abuse.

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

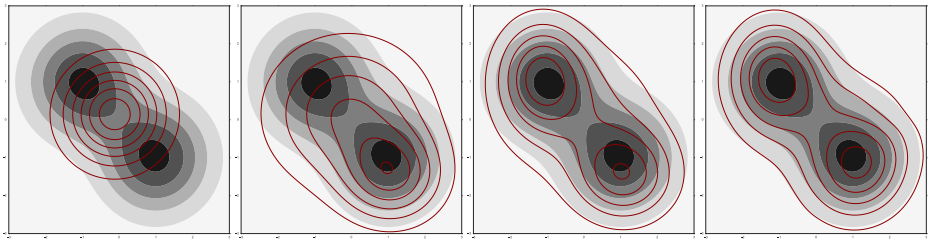

Figure 1: Variational Boosting applied to a 2-d target at steps $C = 1, 2, 3, 8$. The background (grey/black) contours depict the target distribution, and the foreground (red) contours depict the iterative approximations.

respect to λ. Directly minimizing $KL(q_\lambda||\pi)$ is often intractable, however, we can construct a tractable objective that, when maximized, corresponds to minimizing $KL(q_\lambda||\pi)$. This objective is often referred to as the *evidence lower bound* (ELBO), written

$$\begin{aligned}
\mathcal{L}(\lambda) &= \mathbb{E}_{q_\lambda}\left[\ln \tilde{\pi}(x) - \ln q(x;\lambda)\right] && \text{ELBO} && (1)\\
&= \mathbb{E}_{q_\lambda}\left[\ln \pi(x) - \ln q(x;\lambda)\right] + \ln \mathcal{C} && \tilde{\pi}(x) = \mathcal{C}\pi(x)\text{(unknown const.)} && (2)\\
&= \ln \mathcal{C} - KL(q_\lambda||\pi) && && (3)\\
&\leq \ln \mathcal{C} = \ln \int \tilde{\pi}(x)dx && \text{marginal likelihood} && (4)
\end{aligned}$$

which, due to the non-negativity of $KL(q||\pi)$, is a lower bound on the normalization constant[3] of $\tilde{\pi}(x)$.

Variational methods typically define (or derive) a family of distributions $Q = \{q(\cdot;\lambda) : \lambda \in \Lambda\}$ parameterized by λ, and maximize the ELBO with respect to $\lambda \in \Lambda$. Most commonly the class Q is fixed, and there exists some (possibly non-unique) $\lambda^* \in \Lambda$ for which $KL(q_\lambda||\pi)$ is minimized. When the family Q does not include π, there will be a non-zero KL gap between $q(\cdot;\lambda^*)$ and π, and that discrepancy will induce bias in posterior summaries and predictions.

In the following section, we propose an algorithm that iteratively grows the approximating class Q, and reframes the VI procedure as a series of optimization problems.

2 Method

We define our approximate distribution to be a mixture of C simpler component distributions

$$q^{(C)}(x;\lambda) = \sum_{c=1}^{C} \rho_c q_c(x;\lambda_c) \quad \text{s.t.} \sum_c \rho_c = 1 \tag{5}$$

where we have defined component distributions q_c,[4] mixture component parameters $\lambda = (\lambda_1, \ldots, \lambda_C)$, and mixing proportion parameters $\rho = (\rho_1, \ldots, \rho_C)$. The component distributions can be any distribution over x from which we can draw samples using a continuous map (e.g. multivariate normals [6], or a composition of invertible maps [11]).

Our method begins with a single mixture component, $C = 1$. We use existing VI methods to fit the first component parameter, λ_1, and ρ_1 is fixed to 1 by definition. At the next iteration, we fix λ_1 and introduce a new component into the mixture, $q_2(x;\lambda_2)$. We then introduce a new ELBO objective as a function of new component parameters, λ_2, and a new mixture weight, ρ_2. We then optimize this objective until convergence. At each subsequent iteration, we introduce new component parameters and a mixing weight, and then we optimize the new objective. We refer to this procedure as *variational boosting*, inspired by methods for learning strong classifiers by weighting an ensemble of weak classifiers.

[3] Often referred to as the *marginal likelihood*, $p(\text{data})$, in Bayesian inference.

[4] We denote full mixtures with parenthetical superscripts, $q^{(C)}$, and components with naked subscripts, q_c.

In order for our method to be applicable to a general class of target distributions, we use black-box variational inference methods and the *re-parameterization trick* [13, 8, 10] to fit each component and mixture weights. The re-parameterization trick is a method for obtaining unbiased estimates of the gradient of the ELBO. These gradient estimates can then be used to optimize the ELBO objective using a stochastic gradient optimization method. However, using mixtures as the variational approximation complicates the use of the re-parameterization trick. Details on the re-parameterization trick and its use in mixtures are in Appendix A.

2.1 Variational Boosting

Fitting the first component The procedure starts by fitting an approximation to $\pi(x)$ with a distribution that consists of a single component. We do this by maximizing the first ELBO objective

$$\mathcal{L}^{(1)}(\lambda_1) = \mathbb{E}_q \left[\ln \pi(x) - \ln q_1(x; \lambda_1)\right] \tag{6}$$

$$\lambda_1^* = \arg\max_{\lambda_1} \mathcal{L}^{(1)}(\lambda_1)\,. \tag{7}$$

Depending on the forms of π and q_1, optimizing the ELBO can be accomplished by various methods. One general method for fitting a continuous valued component is to compute stochastic, unbiased gradients of $\mathcal{L}(\lambda_1)$, and use stochastic gradient optimization. See Appendix A for details. After convergence (or close to it) we fix λ_1 to be λ_1^*.

Fitting component $C+1$ After iteration C, our current approximation to $\pi(x)$ is a mixture distribution with C components

$$q^{(C)}(x; \lambda) = \sum_{c=1}^{C} \rho_c q_c(x; \lambda_c) \tag{8}$$

where $\lambda = (\{\rho_c, \lambda_c\}_c)$ is a list of component parameters and mixing weights, and $q_c(x; \lambda_c)$ is the component distribution parameterized by λ_c. Adding a new component introduces a new component parameter, λ_{C+1}, and a new mixing weight, ρ_{C+1}. In this section, the mixing parameter $\rho_{C+1} \in [0, 1]$ mixes between the new component, $q_{C+1}(\cdot; \lambda_{C+1})$ and the existing distribution, $q^{(C)}$. The new approximate distribution is

$$q^{(C+1)}(x; \rho_{C+1}, \lambda_{C+1}) = (1 - \rho_{C+1}) q^{(C)}(x) + \rho_{C+1} q_{C+1}(x; \lambda_{C+1})$$

The new optimization objective, as a function of ρ_{C+1} and λ_{C+1} is

$$\begin{aligned}
\mathcal{L}^{(C+1)}(\rho_{C+1}, \lambda_{C+1}) &= \mathbb{E}_{x \sim q^{(C+1)}} \left[\ln \pi(x) - \ln q^{(C+1)}(x; \lambda_{C+1}, \rho_{C+1})\right] \\
&= (1 - \rho_{C+1}) \mathbb{E}_{q^{(C)}} \left[\ln \pi(x) - \ln q^{(C+1)}(x; \lambda_{C+1}, \rho_{C+1})\right] \\
&\quad + \rho_{C+1} \mathbb{E}_{q_{C+1}} \left[\ln \pi(x) - \ln q^{(C+1)}(x; \lambda_{C+1}, \rho_{C+1})\right]
\end{aligned}$$

Above we have separated out two expectations — one with respect to the existing approximation (which is fixed), and the other with respect to the new component distribution. Because we fix the existing component distributions we only need to optimize the new component parameters $\lambda_{C+1}, \rho_{C+1}$, allowing us to use the re-parameterization trick and Monte Carlo gradients to optimize $\mathcal{L}^{(C+1)}$. The details of component initialization are in Appendix B. Figure 1 depicts this procedure on a simple, two-dimensional target distribution.

2.2 Related Work

Using a mixture model as an approximating distribution in variational inference is a well-studied idea. Mixtures of mean field approximations [6] introduced mean field-like updates for a mixture approximation using a bound on the entropy term and model-specific parameter updates. Nonparametric variational inference [3] is a black-box variational inference algorithm that approximates a target distribution with a mixture of equally-weighted isotropic normals. The authors use a lower bound on the entropy term in the ELBO to make the optimization procedure tractable. Similarly,

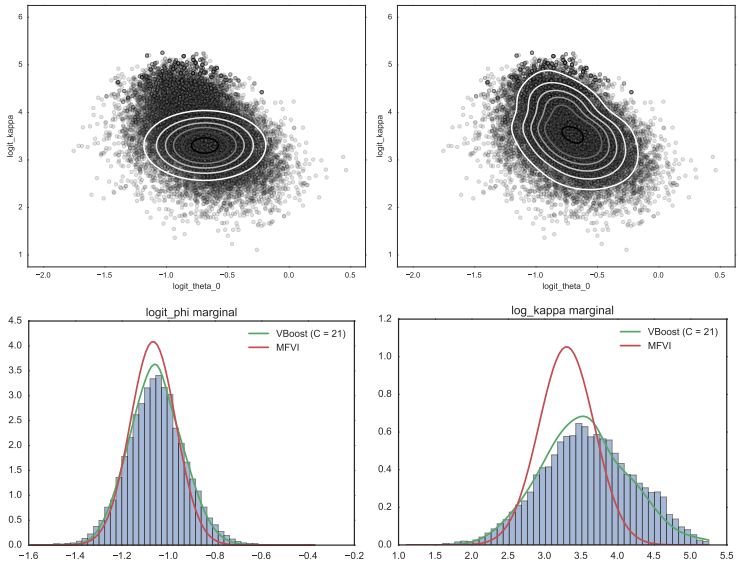

Figure 2: Comparison of univariate and bivariate marginals for the binomial hierarchical model. Each histogram/scatterplot results from 20,000 NUTS samples. Top left: Bivariate marginal (κ, θ_0) HMC samples and a mean field approximation (MFVI). Top Right, the same bivariate marginal, and the Variational Boosting approximation. Bottom: comparison of NUTS, MFVI, and VBoost on univariate marginals (global parameters).

[13] present a method for fitting mixture distributions as an approximation. However, their method is restricted to mixture component distributions within the exponential family, and a joint optimization procedure. Another related thread of research is boosting density estimation [12], which iteratively improves unsupervised models of data. Finally, we note that [4] independently and in parallel propose a closely-related idea for an iterative "boosted" construction of a variational approximation.

3 Experiments and Analysis

Hierarchical Binomial Regression We test out our posterior approximation on a hierarchical binomial regression model.[5] We borrow an example from [2], and estimate the the binomial rates of success (batting averages) of baseball players using a hierarchical model — details of the model are in Appendix C.

To highlight the fidelity of our method, we compare Variational Boosting to mean field VI and the No-U-Turn Sampler (NUTS) [5]. The empirical distribution resulting from 20k NUTS samples is considered the "ground truth" posterior in this example. Figure 2 compares a selection of univariate and bivariate posterior marginals. We see that Variational Boosting is able to closely match the NUTS posterior estimate, allowing the user to improve upon the MFVI approximation.

4 Discussion and Conclusion

We have proposed a variational inference method that iteratively incorporates new components into the approximation. We see multiple directions for future work. The Variational Boosting framework allows for more flexible component distributions than diagonal Gaussians. For instance, compositions of invertible maps have been used to enrich variational families [11], as well as auxiliary variable variational models [9], both of which could be used as component distributions in a larger mixture. We also plan to explore alternative optimization procedures. While this work uses first-order stochastic gradient methods to fit variational approximations, natural gradient and second-order optimization methods have been shown to be effective and more efficient.

[5] Model and data from the `mc-stan` case studies, `http://mc-stan.org/documentation/case-studies/pool-binary-trials.html`

Acknowledgments

The authors would like to acknowledge Arjumand Masood, Mike Hughes, and Finale Doshi-Velez for helpful conversations.

References

[1] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *arXiv preprint arXiv:1601.00670*, 2016.

[2] Bradley Efron and Carl Morris. Data analysis using stein's estimator and its generalizations. *Journal of the American Statistical Association*, 70(350):311–319, 1975.

[3] Samuel Gershman, Matt Hoffman, and David M Blei. Nonparametric variational inference. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 663–670, 2012.

[4] Fangjian Guo, Xiangyu Wang, Kai Fan, Tamara Broderick, and David B. Dunson. Boosting variational inference. *arXiv preprint arXiv:1611.05559*, 2016.

[5] Matthew D Hoffman and Andrew Gelman. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014.

[6] Tommi S Jaakkola and Michael I Jordan. Improving the mean field approximation via the use of mixture distributions. In *Learning in graphical models*, pages 163–173. Springer, 1998.

[7] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[8] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[9] Lars Maaløe, Casper Kaae Sønderby, Søren Kaae Sønderby, and Ole Winther. Auxiliary deep generative models. *arXiv preprint arXiv:1602.05473*, 2016.

[10] Rajesh Ranganath, Sean Gerrish, and David M Blei. Black box variational inference. In *AISTATS*, pages 814–822, 2014.

[11] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 1530–1538, 2015.

[12] Saharon Rosset and Eran Segal. Boosting density estimation. In *Advances in Neural Information Processing Systems*, pages 641–648, 2002.

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

[14] Martin J Wainwright and Michael I Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1-2):1–305, 2008.

A The re-parameterization trick

The re-parameterization trick is a method for computing low-variance estimates of the gradient of an objective for which we only have an unbiased estimator

$$\begin{aligned}\mathcal{L}(\lambda) &= \mathbb{E}_q\left[\ln \pi(x) - \ln q(x;\lambda)\right] \\ &\approx \frac{1}{L}\sum_{\ell=1}^{L}\left[\ln \pi(x^{(\ell)}) - \ln q(x^{(\ell)};\lambda)\right] \quad x^{(\ell)} \sim q(x;\lambda)\end{aligned}$$

To obtain a Monte Carlo gradient of $\mathcal{L}(\lambda)$ using the re-parameterization trick, we first separate the randomness needed to generate $x^{(\ell)}$ from the parameters λ, by defining a deterministic map $f_q(x_0;\lambda) = x^{(\ell)}$ such that $x_0 \sim q_0$[6] implies $x^{(\ell)} \sim q(x;\lambda)$. Then, we can differentiate through f_q with respect to λ to obtain a gradient estimator. This takes advantage of the following equivalence

$$\mathcal{L}(\lambda) = \mathbb{E}_{x\sim q_\lambda}\left[\ln \pi(x) - \ln q(x;\lambda)\right] \tag{9}$$

$$= \mathbb{E}_{x_0\sim q_0}\left[\ln \pi(f_q(x_0;\lambda)) - \ln q(f_q(x_0;\lambda);\lambda)\right] \tag{10}$$

Now that the stochasticity has been separated from parameters λ, we can move the gradient operator into the expectation

$$\mathcal{L}(\lambda) = \nabla_\lambda \mathbb{E}_{x_0\sim q_0}\left[\ln \pi(f_q(x_0;\lambda)) - \ln q(f_q(x_0;\lambda);\lambda)\right] \tag{11}$$

$$= \mathbb{E}_{x_0\sim q_0}\nabla_\lambda\left[\ln \pi(f_q(x_0;\lambda)) - \ln q(f_q(x_0;\lambda);\lambda)\right] \tag{12}$$

which directly translates the Monte Carlo objective estimator into a Monte Carlo objective gradient estimator.

A.1 The re-parameterization trick for mixtures

The re-parameterization trick when q is a mixture, however, is less straightforward. The sampling procedure for a mixture model typically contains a discrete component (i.e. sampling component identities), which is a process that cannot be differentiated through. We circumvent this complication by re-writing the variational objective as a weighted combination of expectations with respect to individual mixture components. Because of the form of the mixture, we can write the ELBO as

$$\begin{aligned}\mathcal{L}(\lambda,\rho) &= \mathbb{E}_q\left[\ln \pi(x) - \ln q(x;\lambda)\right] \\ &= \int\left(\sum_{c=1}^{C}\rho_c q_c(x;\lambda_c)\right)\left[\ln \pi(x) - \ln q(x;\lambda)\right]dx \\ &= \sum_{c=1}^{C}\rho_c\int q_c(x;\lambda_c)\left[\ln \pi(x) - \ln q(x;\lambda)\right]dx \\ &= \sum_{c=1}^{C}\rho_c\mathbb{E}_{q_c}\left[\ln \pi(x) - \ln q(x;\lambda)\right]\end{aligned}$$

which is a function of expectations with respect to *mixture components*. If these distributions are continuous, and there exists some function $f_c(x_0;\lambda)$ such that $f_c(x_0;\lambda) \sim q_c(\cdot;\lambda)$ when $x_0 \sim q_0$, then we can apply the re-parameterization trick to each component to obtain gradients of the ELBO

$$\begin{aligned}\nabla_{\lambda_c}\mathcal{L}(\lambda,\rho) &= \nabla_{\lambda_c}\sum_{c=1}^{C}\rho_c\mathbb{E}_{x\sim q(x;\lambda)}\left[\ln \pi(x) - \ln q(x;\lambda)\right] \\ &= \sum_{c=1}^{C}\rho_c\mathbb{E}_{x_0\sim q_0}\left[\nabla_{\lambda_c}\ln \pi(f_c(x_0;\lambda_c)) - \nabla_{\lambda_c}\ln q(f_c(x_0;\lambda_c))\right].\end{aligned}$$

Variational Boosting uses the above fact to use the re-parameterization trick in a component-by-component manner, allowing us to improve the variational approximation as we incorporate and fit new components.

B Component Initialization

Initializing Components Introducing a new component requires initialization of component parameters. When our component distributions are mixtures of Gaussians, we found that the optimization procedure is sensitive to initialization. We found that the following importance-weighted scheme improves the optimization objective. To initialize a new mean, μ_{C+1}, we first draw L samples from the existing distribution, $x^\ell \sim q^{(C)}$. For each sample, we compute an importance weight, $\ln w^\ell = \ln \pi(x) - \ln q^{(C)}$. We initialize μ_{C+1} to the sample with the largest importance weight. We initialize this component to be small, and the new mixing weight to be small (around .01).

[6] Here, q_0 is some base distribution that is, importantly, *not* a function of λ.

C Example Hierarchical Model

The model of the data is

$$\phi \sim \texttt{Unif} \qquad \text{hyper prior} \tag{13}$$

$$\kappa \sim \texttt{Pareto}(1, 1.5) \qquad \text{hyper prior} \tag{14}$$

$$\theta_j \sim \texttt{Beta}(\phi \cdot \kappa, (1 - \phi) \cdot \kappa) \qquad \text{player } j \text{ prior} \tag{15}$$

$$y_j \sim \texttt{Binomial}(K_j, \theta_j) \qquad \text{player } j \text{ hits} \tag{16}$$

where y_j is the number of successes (hits) player j has attempted in K_j attempts (at bats). Each player has a latent success rate θ_j, which is governed by two global variables κ and ϕ. There are 18 players in this small example, with a total of $D = 20$ parameters. This model is not conjugate, and requires approximate Bayesian inference.