JMLR: Workshop and Conference Proceedings 1:1–11, 2018. Symposium on Advances in Approximate Bayesian Inference

Challenges in Computing and Optimizing Upper Bounds of Marginal Likelihood based on Chi-Square Divergences

Melanie F. Pradier FERNANDEZPRADIER@G.HARVARD.EDU
Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA USA

Michael C. Hughes MHUGHES@CS.TUFTS.EDU
Dept. of Computer Science, Tufts University, Medford, MA USA

Finale Doshi-Velez FINALE@SEAS.HARVARD.EDU
Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, MA USA

Editor: Editor's name

Abstract

Variational inference based on χ^2 divergence minimization (CHIVI) provides a way to approximate a model's posterior while obtaining an upper bound on the marginal likelihood. However, in practice CHIVI relies on Monte Carlo (MC) estimates of an upper bound objective that at modest sample sizes are not guaranteed to be true bounds on the marginal likelihood. This paper provides an empirical study of CHIVI performance on a series of synthetic inference tasks. We show that CHIVI is far more sensitive to initialization than classic VI based on KL minimization, often needs a very large number of samples (over a million), and may not be a reliable upper bound. We also suggest possible ways to detect and alleviate some of these pathologies, including diagnostic bounds and initialization strategies.

Keywords: Latent variable models, upper bound on marginal likelihood, χ^2 divergence

1. Introduction

Estimating the marginal likelihood in probabilistic models is the holy grail of Bayesian inference. Marginal likelihoods allow us to compute the posterior probability of model parameters or perform Bayesian model selection (Bishop et al., 1995). While exact computation of the marginal is not tractable for most models, variational inference (VI) (Jordan et al., 1999) offers a promising and scalable approximation. VI suggests choosing a simple family of approximate distributions q and then optimizing the parameters of q to minimize its divergence from the true (intractable) posterior. The canonical choice is the KL divergence, where minimizing corresponds to tightening a *lower* bound on the marginal likelihood. Recently, (Dieng et al., 2017a) showed that minimizing a χ^2 divergence leads to a chi-divergence upper bound ("CUBO"). Practitioners often wish to combine upper and lower bound estimates to "sandwich" the model evidence in a narrow range for later decision making, so the CUBO's flexible applicability to all latent variable models is appealing.

However, both the estimation of the upper bound and computing its gradient for minimization require Monte Carlo estimators to approximate tough integrals. These estimators may have large variance even at modest number of samples. A natural question is then how reliable CUBO minimization is in practice. In this paper, we provide empirical evidence that CUBO optimization is often tricky, and the bound itself ends up being too loose even

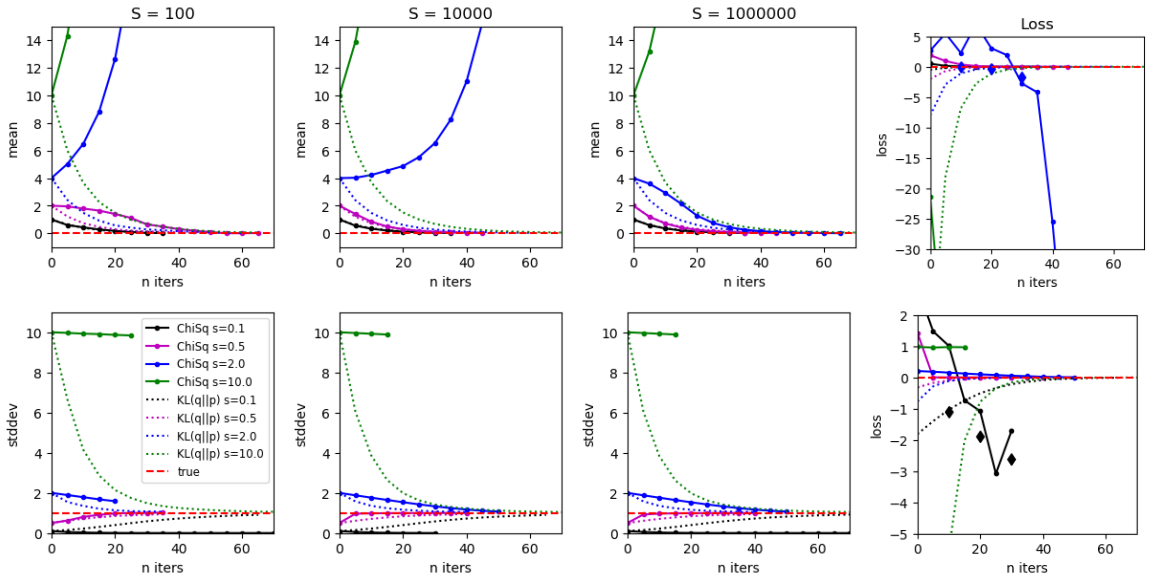

Figure 1: Minimizing χ^2 divergence using MC gradient estimates via the reparametrization trick can be challenging even with simple univariate Gaussian distributions. Each column shows results under a different number of MC samples. The last column compares ELBO and CUBO traces for $S = 10^4$; diamonds correspond to sanity-check estimator from Eq. (2). *Top row*: variational parameter traces with fixed true variance but changing starting mean locations. *Bottom row:* same, but with fixed true mean and changing start variance values.

using hundreds of samples. Our contributions include: i) *evaluation* of the CUBO in two simple scenarios, and comparison to other bounds to gauge its utility; ii) empirical analysis of CUBO *optimization* in both scenarios, in terms of convergence rate and sensitivity to the number of samples; iii) review of alternative upper bounds and best practices for diagnosing and testing new bounds.

2. χ-Divergence Variational Inference via Monte Carlo Gradient Descent

Let $p(\mathbf{x}, \mathbf{z})$ be the joint distribution of observed variables $\mathbf{x}$ and latent variables $\mathbf{z}$. Variational inference (VI) approximates the posterior distribution $p(\mathbf{z}|\mathbf{x})$ through optimization. The idea is to posit a family of variational distributions and find the member distribution $q(\mathbf{z}; \boldsymbol{\lambda})$ which is as close as possible to the true posterior. Standard VI minimizes the KL divergence $D_{\mathrm{KL}}\big(q(\mathbf{z}; \boldsymbol{\lambda})||p(\mathbf{z}|\mathbf{x})\big)$. Minimizing the KL divergence is equivalent to maximizing the evidence lower bound (ELBO) on the model evidence $\log p(\mathbf{x})$. Alternatively, χ^2 variational inference (Dieng et al., 2017b) minimizes the χ^2 divergence $D_{\mathrm{KL}}\big(p(\mathbf{z}|\mathbf{x})||q(\mathbf{z}; \boldsymbol{\lambda})\big)$. This is equivalent to minimizing the following upper bound (CUBO):

$$\mathcal{L}_{\mathrm{CUBO}}(\boldsymbol{\lambda}) = \frac{1}{2} \log \mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})} \left[\left(\frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z}; \boldsymbol{\lambda})} \right)^2 \right]. \tag{1}$$

The expectation in the CUBO is usually intractable, so we use Monte Carlo samples to construct a biased estimate $L(\boldsymbol{\lambda}) = \frac{1}{2} \log \frac{1}{S} \sum_{s=1}^{S} \left(\frac{p(\mathbf{x}, \mathbf{z}^{(s)})}{q(\mathbf{z}^{(s)}; \boldsymbol{\lambda})} \right)^2$ where $z^{(1)}, \ldots, z^{(S)} \sim q(\mathbf{z}; \boldsymbol{\lambda})$.

In this paper, we consider two optimization strategies, both relying on the reparametrization trick (Kingma and Welling, 2013; Rezende et al., 2014; Titsias and Lázaro-Gredilla, 2014): i) optimizing the CUBO directly in Eq. (1) using biased gradient estimators; ii) optimizing the exponentiated CUBO defined as $\mathcal{L}_{\text{EXPCUBO}}(\boldsymbol{\lambda}) = \exp(2\ \mathcal{L}_{\text{CUBO}}(\boldsymbol{\lambda}))$, whose gradients are unbiased but might suffer from higher variance.

3. Case Study: CUBO Optimization for Univariate Gaussians

We consider a simple inference scenario: minimizing the divergence between two univariate Gaussian distributions. We assume no data $\mathbf{x}$, such that the true posterior is just the prior fixed at $p(\mathbf{z}) \doteq \mathcal{N}(0, 1)$. We consider two cases: a variational distribution $q(\mathbf{z}; \tilde{\mu}, \tilde{\sigma}^2)$ with fixed $\tilde{\sigma} = 1.0$ and varying mean $\tilde{\mu} = \{1, 2, 4, 10\}$, or the other way around, fixed $\tilde{\mu} = 0.0$ and varying $\tilde{\sigma} = \{0.1, 0.5, 2.0, 10.0\}$. All experiments were performed using stochastic gradient descent (Bottou, 2010) and grid-searching the learning rate for each different bound independently in a fine grid between 10^{-4} and 1.0.

Fig. 1 shows the evolution of the variational parameters over time when minimizing the χ^2 divergence (ChiSq) or maximizing the KL divergence (KL) from different initialization points. While the KL trajectories always converge to the true values, the ChiSq variational parameters fail to converge for 5 out of the 8 cases when the number of MC samples $S = 100$. If we increase the number of samples S to 1M, 3 out of 8 cases still fail to find the true values. Most alarming, in several cases, e.g., fixed mean and varying $\tilde{\sigma}$ initialized at 0.1, the CUBO MC estimator present values below 0 (the true marginal likelihood value), so it is not an upper bound anymore, even with 1M samples. Appendix A show similar pathological behaviors for the exponentiated CUBO case.

To assess CUBO correctness, consider an alternative MC estimator that samples from the prior p, rather than from q:

$$\mathcal{L}_{\text{CUBO}}(\boldsymbol{\lambda}) = \frac{1}{2}\log \mathbb{E}_{p(\mathbf{z})}\left[p(\mathbf{x}|\mathbf{z})^2\left(\frac{p(\mathbf{z})}{q(\mathbf{z};\boldsymbol{\lambda})}\right)\right] \approx \frac{1}{2}\log \frac{1}{S}\sum_{s=1}^{S} p(\mathbf{x}|\mathbf{z}^{(s)})^2\left(\frac{p(\mathbf{z}^{(s)})}{q(\mathbf{z}^{(s)};\boldsymbol{\lambda})}\right), \tag{2}$$

where $z^{(1)}, \ldots, z^{(S)} \sim p(\mathbf{z})$. In general, since CUBO optimization is sensitive to initialization, it is a good practice to do warm initializations, either with MAP estimation or by performing KL optimization first during a few iterations.

4. Case Study: CUBO Optimization for Topic Models

We consider applying the CUBO training objective to the Latent Dirichlet Allocation (LDA) topic model (Blei et al., 2003). We focus on single-document inference, where the *length* of the document should directly impact posterior uncertainty about which topics are used. We assume that there are $K = 3$ topics and $V = 3$ vocabulary words. We are given a set of topic-word probabilities ϕ where ϕ_{kv} is the probability of word v under topic k. Each document d is represented by counts of V discrete words or features, $x_d \in \mathbb{Z}_+^V$. These counts are generated via a document-specific mixture of K topics, $x_d \sim \text{Mult}(x_d \mid \sum_{k=1}^K \pi_{dk}\phi_k, N_d)$. The probabilities π_d, where $\sum_{k=1}^K \pi_{dk} = 1$, have a conjugate Dirichlet prior with hyperparameter α: $\pi_d \sim \text{Dir}(\pi_d \mid \alpha_1, \ldots \alpha_K)$. Given a specific document d, our goal is to estimate an approximate posterior $q(\pi_d)$ over the document-topic probabilities via

	method	$\log p(x)$ (5%, 95%)	
$q(\pi)$: Dirichlet		10^2 samples	10^5 samples
warm init optimized ELBO	UB KLpq	-25.06 (-25.07,-25.04)	-25.06 (-25.06,-25.06)
$q(\pi)$: LogisticNorm		10^2 samples	10^5 samples
cold init optimized ELBO	CUBO	-24.59 (-25.18,-22.99)	-23.15 (-23.42,-20.20)
	UB KLpq	-23.33 (-23.84,-22.73)	-23.39 (-23.41,-23.37)
warm init optimized CUBO	CUBO	-24.67 (-24.87,-24.19)	-24.37 (-24.46,-24.08)
	UB KLpq	-24.42 (-24.55,-24.25)	-24.39 (-24.40,-24.39)
cold init optimized CUBO	N/A: Optimizer diverged		

Table 1: Bounds on marginal likelihood for a "long" toy document under an LDA topic model. We infer an approximate posterior for a single document with 100 words, using either a Dirichlet q (top row) or MC gradient updates to fit a LogisticNormal q (bottom rows) with 100 samples per gradient step. We evaluate CUBO and KLpq, see Appendix B.

variational inference. In particular, we explore two tasks: (i) estimating upper bounds on the marginal likelihood given a fixed q, and (ii) optimizing q to try to improve such bounds.

Experiment: Reliability of Upper Bound Estimation To assess the reliability of upper bound estimation using approximate distributions, we fit four possible q: one Dirichlet via closed-form updates optimizing the ELBO, and 3 separate Logistic Normal (LN) distributions fit via Monte-Carlo gradient descent steps (see details for each q in the appendix). The 3 LNs are respectively a cold-started optimization of the ELBO, a warm-started optimization of the CUBO, and a cold-started optimization of the CUBO. Warm-starting here means that the mean of q is set to the maximum likelihood estimator of the document-topic vector π_d, while cold-starting has random parameters not informed by the data. We hope that these detailed experiments tease apart the impact of initialization and optimization.

In Tab. 1 and Tab. 2, for each q described above, we compare CUBO to an alternative upper bound KLpq, detailed in Appendix B. For each stochastic upper bound estimator, we compute 20 replicates using each 100 samples and 100,000 samples, then report the median of these samples as well as 5-th and 95-th percentile value intervals. Our conclusions are:

CHIVI parameter estimation often diverges for cold initializations. We replicated this issue across many settings, as reported in Tab. 1.

CUBO estimators are overconfident. Increasing sample size *widens* confidence intervals. KLpq estimators are better behaved. Consider Tab. 2's warm-init CUBO row (in Appendix A): At 100 samples the CUBO seems to be within (-1.03, 0.77), but at many more samples, the CUBO interval drops to (-0.86, -0.64), with a new median that is just barely contained in the previous interval. In contrast, the 100 sample KLpq bound has an interval that shrinks.

ELBO optimization followed by CUBO computation may be enough. The Dirichlet q optimized for the ELBO but then fitted into a CUBO estimator produces competitive bounds. This suggests that it may not always be necessary to optimize the CUBO directly.

References

J. Aitchison and S. M. Shen. Logistic-normal distributions: Some properties and uses. *Biometrika*, 67(2):261–272, 1980. URL http://www.jstor.org/stable/2335470.

Christopher M Bishop et al. *Neural networks for pattern recognition*. Oxford university press, 1995.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. 3: 993–1022, 2003.

Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

Adji B Dieng, Dustin Tran, Rajesh Ranganath, John Paisley, and David M Blei. The χ-divergence for approximate inference. 2017a.

Adji Bousso Dieng, Dustin Tran, Rajesh Ranganath, John Paisley, and David Blei. Variational inference via χ upper bound minimization. In *Advances in Neural Information Processing Systems*, pages 2732–2741, 2017b.

Andrew Gelman, Daniel Lee, and Jiqiang Guo. Stan: A probabilistic programming language for bayesian inference and optimization. *Journal of Educational and Behavioral Statistics*, 40(5):530–543, 2015.

Chunlin Ji, Haige Shen, and Mike West. Bounded approximations for marginal likelihoods. Technical Report 10–05, Duke University Dept. of Statistics, 2010.

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.

Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv:1312.6114 [cs, stat]*, December 2013. arXiv: 1312.6114.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. *arXiv:1401.4082 [cs, stat]*, January 2014. arXiv: 1401.4082.

Michalis Titsias and Miguel Lázaro-Gredilla. Doubly stochastic variational bayes for non-conjugate inference. In *International conference on machine learning*, pages 1971–1979, 2014.

Appendix A. Extra Experiments

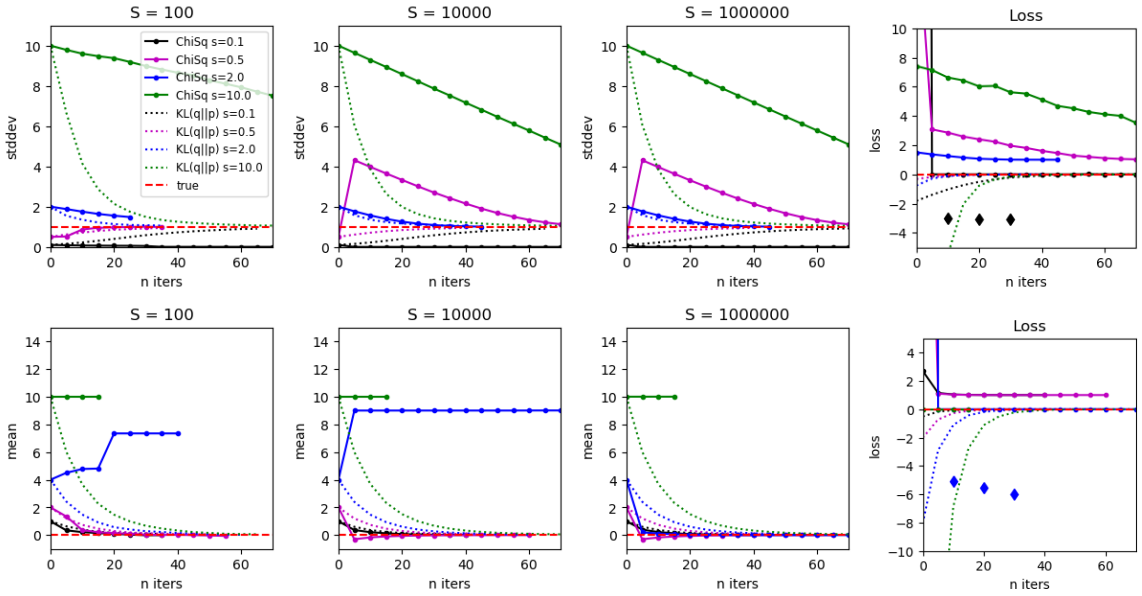

Figure 2: **Univariate Gaussian case study.** Minimizing the exponentiated CUBO using MC gradient estimates via the reparameterization trick can be challenging even with simple univariate Gaussian distributions. Each column shows results under a different number of MC samples. Las columns shows ELBO and CUBO traces for $S = 10^4$ samples. *Top row*: Comparison of variational parameter traces, while minimizing ELBO and CUBO, with fixed true variance but changing starting locations of the mean (farther and farther from true mean). *Bottom row:* Comparison of ELBO and CUBO traces, with fixed true mean but changing start variance values (some larger, some smaller).

method	$\log p(x)$ (5%, 95%)
exact	-1.0986

$q(\pi)$: Dirichlet	10^2 samples	10^5 samples
warm init optimized ELBO		
UB KLpq	-1.08 (-1.12, -1.04)	-1.08 (-1.08, -1.07)

$q(\pi)$=LogisticNorm	10^2 samples	10^5 samples
warm init optimized ELBO		
CUBO (q)	-0.89 (-1.19, -0.16)	-0.31 (-0.51, 0.13)
UB KLpq	-0.67 (-0.90, -0.46)	-0.70 (-0.71, -0.69)
warm init optimized CUBO		
CUBO (q)	-0.89 (-1.03, -0.77)	-0.82 (-0.86, -0.64)
UB KLpq	-0.81 (-0.87, -0.69)	-0.79 (-0.80, -0.79)
cold init optimized CUBO		
CUBO (q)	-0.88 (-0.98, -0.79)	-0.80 (-0.86, -0.67)
UB KLpq	-0.80 (-0.88, -0.67)	-0.79 (-0.79, -0.78)

Table 2: **Topic model case study.** Bounds on marginal likelihood for a "short" toy document under an LDA topic model. We infer an approximate posterior over doc-topic probabilities for a single document with just 1 word, using either closed-form coordinate ascent updates to fit a Dirichlet q (top row) or MC gradient updates to fit a LogisticNormal q (bottom rows) with 100 samples per gradient step. Using the final fitted q, we then compute 20 replicates of our stochastic upper bounds on marginal likelihood using either the CUBO or the KLpq estimator (see Appendix B, using $S = 10^2$ or 10^5 samples for each. We show the median value and the (5%, 95%) interval.

Appendix B. The "KLpq" bound : reliable but expensive.

Given any approximate posterior $q(\pi_d)$ parameterized by $\hat{v}_d \in \mathcal{V}$, the following is an upper bound on the marginal likelihood:

$$\text{UBx\_KLpq}(x_d, \phi, \hat{v}_d) \triangleq \mathbb{E}_{\pi_d \sim p(\pi_d|x_d)}\Big[\log \frac{p(\pi_d, x_d|\phi)}{q(\pi_d|\hat{v}_d)}\Big], \approx \frac{1}{S}\sum_{s=1}^{S}\Big[\log \frac{p(\pi_d^s, x_d|\phi)}{q(\pi_d^s)}\Big] \quad (3)$$

Ji et al. (2010) show that minimizing this bound is equivalent to minimizing $\text{KL}(p||q)$, which computes the asymmetric KL divergence in the *opposite* direction of typical variational methods, which minimize $\text{KL}(q||p)$. We suggest that this bound is a useful comparison point for the CUBO bound.

The "KLpq" upper bound can be approximated using S samples from the posterior $\pi_d^s \sim p(\pi_d|x_d, \phi)$. For our LDA model, we compute S samples from a Hamiltonian Monte Carlo posterior using Stan (Gelman et al., 2015).

Appendix C. Details on Variational Approximate Posteriors for Topic Models

C.1. LDA Optimization #1: Dir-Cat + ELBO + CoordAscent

Consider the approximate posterior $q(\pi_d, z_d)$:

$$q(\pi_d|\hat{\theta}_d) = \text{Dir}(\pi_d \mid \hat{\theta}_{d1} \dots \hat{\theta}_{dK}), \tag{4}$$
$$q(z_d|\hat{r}_d) = \prod_{u=1}^{U_d} \text{Cat}(z_{du}|\hat{r}_{du1} \dots \hat{r}_{duK})$$

Objective expression for Dir-Cat

$$\mathcal{L}(x_d, \hat{\theta}_d, \hat{r}_d) \triangleq \mathbb{E}_q\Big[\log p(x_d, \pi_d, z_d) - \log q(\pi_d, z_d|\hat{\theta}_d, \hat{r}_d)\Big] \tag{5}$$

This expression has the following closed-form:

$$\mathcal{L}(x_d, \hat{\theta}_d, \hat{r}_d) = \sum_{k=1}^{K}\sum_{u=1}^{U_d} c_{du}^x \hat{r}_{duk} \log \phi_{k,i_{du}^x} \qquad \textit{from } \mathbb{E}_q[\log p(x_d|z_d)] \tag{6}$$
$$- \sum_{k=1}^{K}\sum_{u=1}^{U_d} c_{du}^x \hat{r}_{duk} \log \hat{r}_{duk} \qquad \textit{from } -\mathbb{E}_q[\log q(z_d)]$$
$$c_K(\alpha) - c_K(\hat{\theta}) + \sum_{k=1}^{K}(N_{dk} + \alpha - \hat{\theta}_{dk})(\psi(\hat{\theta}_{dk}) - \psi(\hat{\theta}_{d\cdot})) \qquad \textit{from } \mathbb{E}_q[\log \frac{p(\pi_d, z_d)}{q(\pi_d)}]$$

where the log cumulant $c_K(\cdot)$ of the K-dimensional Dirichlet log pdf is a log ratio of Gamma functions:

$$c_K(a_1, \dots a_K) = \log \Gamma(\textstyle\sum_{k=1}^K a_k) - \sum_{k=1}^K \log \Gamma(a_k) \tag{7}$$

C.2. LDA Optimization #2: Logistic Normal + MonteCarloGD

Approximate posterior: Logistic Normal. Alternatively, we consider another approximating posterior family which treats the vector π_d as a logistic normal (LN) random variable (Aitchison and Shen, 1980) and marginalizes away $q(z_d)$. We will call this the LN-Marg family for short.

$$q(\pi_d|\hat{m}_d, \hat{s}_d) = \text{LN}(\pi_d \mid \hat{m}_d, \text{diag}(\hat{s}_d^2)), \qquad \hat{m}_d \in \mathbb{R}^{K-1}, \hat{s}_d \in \mathbb{R}_+^{K-1} \tag{8}$$

Here, $\hat{m}_d$ is a vector of mean parameters, and $\hat{s}_d$ a vector of standard deviation parameters. Each has length $K-1$, which is a *minimal* representation.

Because LN random variables are not very common, we write the log probability density function of the approximate posterior here, using results from Aitchison and Shen (1980, Eq. 1.3):

$$\log q(\pi_d) = -\sum_{k=1}^{K} \log \pi_{dk} - \frac{K-1}{2}\log[2\pi] - \sum_{k=1}^{K-1} \log \hat{s}_{dk} - \frac{1}{2}\sum_{k=1}^{K-1} \frac{1}{\hat{s}_{dk}^2}(\log \frac{\pi_{dk}}{\pi_{dK}} - \hat{m}_{dk})^2 \quad (9)$$

The entropy of the distribution is then:

$$\mathbb{E}_q[-\log q(\pi_d)] = \sum_{k=1}^{K} \mathbb{E}_q[\log \pi_{dk}] + \frac{K-1}{2}\log[2\pi] + \sum_{k=1}^{K-1} \log \hat{s}_{dk} + \frac{K-1}{2} \quad (10)$$

where we have used standard results to simplify that last term:

$$\frac{1}{2}\sum_{k=1}^{K-1} \frac{1}{\hat{s}_{dk}^2}\mathbb{E}_{q(\pi)}[(\log \frac{\pi_{dk}}{\pi_{dK}} - \hat{m}_{dk})^2] = \frac{1}{2}\sum_{k=1}^{K-1} \frac{1}{\hat{s}_{dk}^2}\mathbb{E}_{u_k \sim \mathcal{N}(m_k, s_k^2)}[(u_k - \hat{m}_{dk})^2] \quad = \frac{K-1}{2} \quad (11)$$

This expectation $\mathbb{E}_{\pi_d \sim q}[\log \pi_{dk}]$ unfortunately has no closed form.

Reparameterization trick. We can write the random variable π_d as a deterministic transform of a standard normal random variable u_d.

First, recall we can map any $K-1$-length real vector $u \in \mathbb{R}^{K-1}$ to the K-dimensional simplex Δ^K via the *softmax* transformation:

$$\text{smax}([u_1, \dots u_{K-1}]) = \Big[\frac{e^{u_1}}{1 + \sum_{\ell=1}^{K-1} e^{u_\ell}}, \dots, \frac{e^{u_{K-1}}}{1 + \sum_{\ell=1}^{K-1} e^{u_\ell}}, \frac{1}{1 + \sum_{\ell=1}^{K-1} e^{u_\ell}}\Big] \quad (12)$$

This transformation is one-to-one invertible, and also differentiable w.r.t. its input vector.

Now, to generate $\pi_d \in \Delta^K$, we can draw π_d in three steps: (1) draw u_d from a standard normal, (2) scale it with the appropriate mean and standard deviation parameters, and (3) apply the softmax transformation,

$$\pi_d \leftarrow \text{smax}(\hat{s}_d \cdot u_d + \hat{m}_d), \qquad u_d \sim \mathcal{N}(0, I_{K-1}) \quad (13)$$

C.3. LDA Optimization #3: Overcomplete-Logistic-Normal + MonteCarloGD

Transformation between overcomplete simplex and the reals We now consider an *overcomplete* representation of the K-dimensional simplex. Rather than the minimal $K-1$ parameters in the LN-Marg approximation above, let's look at transformations that use K free parameters. In this overcomplete space, we must *augment* our probability vector $\pi_d \in \Delta^K$ (which has only $K-1$ degrees of freedom) with an additional scalar real random variable $w_d \in \mathbb{R}$, so the combined vector $[\pi_{d1} \dots \pi_{d,K-1}\ w_d]$ has the required K linearly-independent dimensions. Now, we can create an *invertible* transformation between two

K-length vectors: a vector u of real values, and the augmented pair π, w:

$$
\begin{aligned}
u_1(\pi, w) &= \log \pi_1 + w & \pi_1(u) &= \frac{e^{u_1}}{\sum_{\ell=1}^{K} e^{u_\ell}} \\
u_2(\pi, w) &= \log \pi_2 + w & \pi_2(u) &= \frac{e^{u_2}}{\sum_{\ell=1}^{K} e^{u_\ell}} \dots \\
u_{K-1}(\pi, w) &= \log \pi_{K-1} + w & \pi_{K-1}(u) &= \frac{e^{u_{K-1}}}{\sum_{\ell=1}^{K} e^{u_\ell}} \\
u_K(\pi, w) &= \log(1 - \sum_{\ell=1}^{K-1} \pi_\ell) + w & w(u) &= \log \sum_{\ell=1}^{K} e^{u_\ell}
\end{aligned}
\tag{14}
$$

Because this is an invertible transformation, we can compute the Jacobian:

$$
J(\pi, w) = \begin{bmatrix}
\frac{\partial u_1}{\partial \pi_1} & \frac{\partial u_1}{\partial \pi_2} & \dots & \frac{\partial u_1}{\partial w} \\
\frac{\partial u_2}{\partial \pi_1} & \frac{\partial u_2}{\partial \pi_2} & \dots & \frac{\partial u_2}{\partial w} \\
\vdots & & \ddots & \\
\frac{\partial u_K}{\partial \pi_1} & \frac{\partial u_K}{\partial \pi_2} & \dots & \frac{\partial u_K}{\partial w}
\end{bmatrix} =
\begin{bmatrix}
\frac{1}{\pi_1} & 0 & 0 & \dots & 0 & 1 \\
0 & \frac{1}{\pi_2} & 0 & \dots & 0 & 1 \\
0 & 0 & \frac{1}{\pi_3} & \dots & 0 & 1 \\
\vdots & & & \ddots & & \\
0 & 0 & 0 & \dots & \frac{1}{\pi_{K-1}} & 1 \\
\frac{-1}{1-\sum_{\ell=1}^{K-1} \pi_\ell} & \frac{-1}{1-\sum_{\ell=1}^{K-1} \pi_\ell} & \frac{-1}{1-\sum_{\ell=1}^{K-1} \pi_\ell} & \dots & \frac{-1}{1-\sum_{\ell=1}^{K-1} \pi_\ell} & 1
\end{bmatrix}
$$

Next, we wish to compute the determinant of this Jacobian, as a function of π and w. First, we perform row and column swaps until only the first column and first row have non-diagonal entries, like this:

$$
J' = \begin{bmatrix}
1 & \frac{-1}{\text{rem}(\pi)} & \frac{-1}{\text{rem}(\pi)} & \dots & \frac{-1}{\text{rem}(\pi)} & \\
1 & \frac{1}{\pi_1} & 0 & 0 & \dots & 0 \\
1 & 0 & \frac{1}{\pi_2} & 0 & \dots & 0 \\
1 & 0 & \frac{1}{\pi_3} & \dots & 0 & 0 \\
\vdots & & & \ddots & & \\
1 & 0 & 0 & \dots & \frac{1}{\pi_{K-1}} &
\end{bmatrix}
\tag{15}
$$

Here, we have defined the remaining mass beyond the $K-1$ independent entries of the vector π as $\text{rem}(\pi) = 1 - \sum_{k=1}^{K-1} \pi_k$ for simplicity. The number of swaps needed to create J' from J is always an even number (there will be the some a swaps needed to fix the rows, and then the same number a swaps for the columns, so $2a$ swaps total). Each single row or column swap changes the sign of the determinant but not the value. An even number of swaps thus leaves the determinant unchanged: $|J'| = |J|$. We can then apply the Schur determinant formula, which says, for any square matrix, we can compute its determinant by manipulating its subcomponent blocks:

$$
\det \begin{vmatrix} A & B \\ C & D \end{vmatrix} = \det|D| \det|A - BD^{-1}C| \tag{16}
$$

Let us choose D as the diagonal block of J': $D = \text{diag}(\frac{1}{\pi_1} \dots \frac{1}{\pi_{K-1}})$. Then we have:

$$\det\ J(\pi, w) = \det\ J' = \left[\prod_{k=1}^{K-1} \frac{1}{\pi_k}\right]\left(1 - \frac{-1}{\text{rem}(\pi)} \sum_{k=1}^{K-1} \pi_k\right) = \left[\prod_{k=1}^{K-1} \frac{1}{\pi_k}\right]\left[\frac{1}{1 - \sum_{\ell=1}^{K-1} \pi_\ell}\right] \tag{17}$$

The simplification arises via algebra after plugging in the definition of $\text{rem}(\pi)$. Armed with the Jacobian and its determinant, we have all the tools needed to perform variational inference in this representation.

Approximate posterior: Overcomplete LN. Returning to our topic modeling task, we consider again the LDA generative model for a document as a given, and wish to compute an approximate posterior for the document-topic vector π_d. We suggest an approximate posterior family based on the overcomplete logistic normal above. We can draw samples from this in two steps. First generate a vector of reals $u_d = [u_{d1} \dots u_{dK}]$ such that $u_{dk} \sim \mathcal{N}(\hat{m}_{dk}, \hat{s}^2_{dk})$. Second, transform this vector u_d to the simplex-plus-real vector $[\pi_{d1} \dots \pi_{dK-1}\ w_d]$ via Eq. (14).

This leads to the following log probability density function over the *joint* space of $\pi, w \in \Delta^K \times \mathbb{R}$:

$$\log q(\pi, w) = \log |\det\ J(\pi, w)| + \sum_{k=1}^{K} \log \mathcal{N}(u_k(\pi, w) | \hat{m}_{dk}, \hat{s}^2_{dk}) \tag{18}$$

Our generative model does not include the log-scale variable w_d, but we can easily just give it a $\mathcal{N}(0, 1)$ prior and keep it decoupled from the data.