Statistical Science
2024, Vol. 39, No. 1, 62–89
https://doi.org/10.1214/23-STS919

Emerging Directions in Bayesian Computation

Steven Winter, Trevor Campbell, Lizhen Lin, Sanvesh Srivastava and David B. Dunson

Abstract. Bayesian models are powerful tools for studying complex data, allowing the analyst to encode rich hierarchical dependencies and leverage prior information. Most importantly, they facilitate a complete characterization of uncertainty through the posterior distribution. Practical posterior computation is commonly performed via MCMC, which can be computationally infeasible for high-dimensional models with many observations. In this article, we discuss the potential to improve posterior computation using ideas from machine learning. Concrete directions are explored in vignettes on normalizing flows, statistical properties of variational approximations, Bayesian coresets and distributed Bayesian inference.

Key words and phrases: Coresets, federated learning, machine learning, normalizing flows, posterior computation, variational Bayes.

1. INTRODUCTION

There is immense interest in performing inference and prediction for complicated real-world processes within science, industry and policy. Bayesian models are appealing because they allow specification of rich generative models encompassing hierarchical structures in the data, natural inclusion of information from experts and/or previous research via priors and a complete characterization of uncertainty in learning/inference/prediction through posterior and predictive distributions. The primary hurdle in applying Bayesian statistics to complex real-world data is posterior computation. In practice, posterior computation—evaluating posterior probabilities/expectations, credible intervals for parameters, posterior inclusion probabilities for features, posterior predictive intervals, etc.—is typically based on posterior samples using the Markov chain Monte Carlo (MCMC). Standard MCMC approaches often fail to converge in practice when the posterior has complicated geometry, such as multiple modes or geometric/manifold constraints. Even sampling from simple posteriors can be challenging when the data has tens or hundreds of millions of observations. This article focuses on the future of Bayesian computation, with emphasis on posterior inference for high-dimensional, geometrically complicated targets with potentially millions or more datapoints.

The recent explosive success of machine learning is key to shaping our vision for the future of Bayesian computation. This paper consists of four vignettes covering recent work on cutting-edge computational techniques, all involving ideas from machine learning. The first vignette covers normalizing flows as a new tool for adaptive MCMC with complicated targets; the second covers recent progress on characterizing the theoretical properties of variational approximations; the third covers data compression via Bayesian coresets to improve the scalability of inference for large data sets and the fourth covers modern techniques in distributed Bayesian inference. All sections focus heavily on promising avenues for future research. For the purposes of this paper, we will generally omit technical details (spaces, measurability, dominating measures, etc.) in the interest of simplicity and brevity.

Steven Winter is Ph.D. Student, Department of Statistical Science, Duke University, Durham, North Carolina 27710, USA (e-mail: steven.winter@duke.edu). Trevor Campbell is Associate Professor, Department of Statistics, University of British Columbia, Vancouver, British Columbia, Canada V6T 1Z4 (e-mail: trevor@stat.ubc.ca). Lizhen Lin is Robert and Sara Lumpkins Associate Professor, Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, Notre Dame, Indiana 46556, USA (e-mail: lizhen.lin@nd.edu). Sanvesh Srivastava is Associate Professor, Department of Statistics and Actuarial Science, University of Iowa, Iowa City, Iowa 52242, USA (e-mail: sanvesh-srivastava@uiowa.edu). David B. Dunson is Arts and Sciences Distinguished Professor, Departments of Statistical Science and Mathematics, Duke University, Durham, North Carolina 27710, USA (e-mail: dunson@duke.edu).

2. MOTIVATION AND CHALLENGES

We consider the problem of estimating Bayesian posterior expectations of the form

$$\mathbb{E}[f] = \int f(\theta)\pi(\theta|x)\,d\theta \tag{1}$$

$$\pi(\theta|x) = \frac{1}{C}\exp\bigl(U(x,\theta)\bigr)\pi_0(\theta), \tag{2}$$

where π denotes the posterior density of the model parameter θ, π_0 denotes the prior density, x denotes the data, U denotes the potential (typically a log-likelihood) function and C denotes the unknown normalizing constant. By selecting the appropriate function f, equation (1) could correspond to the posterior mean, variance, credible intervals, etc. We consider two approaches to Bayesian computation: (1) (approximately) sampling from equation (2), and (2) replacing equation (2) with a tractable distribution.

2.1 Computation via Sampling

The Metropolis–Hastings (MH) algorithm, often combined with Gibbs sampling, underpins many popular methods for obtaining approximate draws from Bayesian posterior distributions [46]. MH is an iterative procedure that generates a sequence of Markovian samples, $\{\theta_t\}_{t=1}^T$, which after appropriate burn-in can be used for Monte Carlo estimation of equation (1). Each step of MH consists of drawing a new state θ' from a proposal kernel $g(\cdot|\theta_t)$, followed by setting the next state θ_{t+1} to θ' with probability

$$\mathrm{acc}(\theta_t,\theta') = \min\left\{1, \frac{\pi(\theta'|x)g(\theta_t|\theta')}{\pi(\theta_t|x)g(\theta'|\theta_t)}\right\}, \tag{3}$$

and setting $\theta_{t+1} = \theta_t$ otherwise. Two high level challenges with MH include (1) designing efficient proposals and (2) efficiently evaluating equation (3) when the potential is computationally expensive.

Proposal Design

Good mixing—roughly, how well $\{\theta_t\}_{t=1}^T$ approximates T i.i.d. draws from π—is critically dependent on the choice of the proposal distribution. Although modern high-dimensional targets with complicated geometry could benefit greatly from flexible, parametrized proposal distributions, these have traditionally been too difficult to tune to be practical. Consequently, it is routine to settle for simpler proposals, such as a multivariate Gaussian centered around the current point, perhaps perturbed by the log-density gradient [174], or trajectories generated by dynamics driven by the target [131, 76, 108]. Parameters of these proposals are then tuned to encourage efficient exploration, for example, by adaptively learning the geometry of the target [72, 171, 199], or by targeting a particular average acceptance probability [9], Section 4. A major limitation of such local methods is their practical inability to cross low-probability regions and adapt to locally varying target geometry, resulting in, for example, poor convergence rates for multimodal distributions [112]. Many solutions have been proposed, ranging from slightly modifying local kernels to encourage crossing low probability regions [95, 140, 107] to constructing entirely new kernels, which are mixtures of a local and global component [8, 3, 159]. Despite these advances, there is still no general solution for sampling complicated high-dimensional distributions.

We believe deep learning ill play an integral role in developing better general solutions. Deep generative models have demonstrated remarkable success in estimating and approximately sampling complicated, high-dimensional distributions, achieving state-of-the-art performance in image/audio/video synthesis, computer graphics, physical/engineering simulations, drug discovery and other domains [74, 93]. Section 3 explores the use of deep generative models to design better proposal distributions for use in MH.

Large-Scale Data

Modern statistical inference problems have another key challenge: large sample sizes. Even the simplest MH algorithms must evaluate the posterior density (cf. equation (3)) at each step, which generally requires a pass over the full data. This requirement makes inference, iterative model development, tuning and verification arduous and error-prone in settings with massive amounts of data. Uniformly subsampling the data at each iteration, as in subsampling MCMC, [13, 94, 110, 217, 4], improves computation speed with the potential cost of slower mixing [81, 128, 14, 160, 161]. In certain situations, it is possible to mitigate these drawbacks by designing an effective control variate for the log-likelihood [160, 133]. Alternatively, one might hope a Gaussian approximation to equation (2) [181, 83, 204, 73, 16] would suffice in the large-data limit. But this is not necessarily the case in modern high-dimensional models with local latent variables, weak or nonidentifiability, discrete variables, low-dimensional manifold structure, model selection priors, etc. We require general-purpose, easy-to-use and statistically rigorous approaches for dealing with large-scale data.

In this work, we consider two generic approaches to the large data problem: (1) compressing redundant data prior to expensive computation and (2) splitting the data, performing potentially expensive operations on smaller subsets of data, and then combining the results. The first approach is formalized by Bayesian coresets [78], which represent the large-scale data by a small, weighted subset that can be passed to any standard inference algorithm, providing posterior inference at a reduced cost. Section 5 introduces coresets and discusses recent advantages as well as important open challenges. The second approach we consider is distributed Bayesian inference, which takes three general forms: (1) one-shot methods that run MCMC in parallel on disjoint subsets of data, and combine the results on a central machine (e.g., [213, 123, 132, 185, 214, 178]); (2) distributed extensions of stochastic gradient MCMC that involve several rounds of communication (e.g., [4, 94, 32, 48]) and (3) variable augmentation approaches (e.g., [20, 157]) based on stochastic extensions of distributed consensus methods, (e.g., [17, 150]). Section 6 formally introduces the three paradigms and discusses future research directions.

2.2 Computation via Approximation

It may not be feasible to sample from equation (2) for complicated potentials, regardless of proposal kernel or sample size. Variational inference is a popular alternative approach, which involves selecting a family of probability distributions $\mathcal{Q}$ on θ, finding the "closest" distribution $q^\star \in \mathcal{Q}$ to π, and using $q^\star$ for downstream inference. This has the major computational advantage of replacing sampling with an optimization problem. Most commonly, the family $\mathcal{Q}$ is both tractable—that is, enables i.i.d. draws and pointwise density evaluation—and parametric—that is, can be written $\mathcal{Q} = \{q_\lambda : \lambda \in \Lambda\}$ for a finite-dimensional parameter space Λ. In addition the "closest" distribution $q^\star$ is usually taken to be the minimizer of the Kullback–Leibler divergence,

$$q^\star = \arg\min_{\lambda \in \Lambda} \mathrm{D}_{\mathrm{KL}}(q_\lambda \| \pi). \tag{4}$$

The optimization in equation (4) is written as a maximization of the *evidence lower bound* (see equation (20)) as is commonly done in the literature; note that the unknown C is removed as it is a constant that does not influence the optimization. Further, note that there are many examples of exceptions to this setup (f-divergences instead of KL [206, 42, 102], nonparametric families [120, 71], families that do not enable density evaluation [193, 225], etc.); we focus on equation (4) as it constitutes the vast majority of the literature.

Most of the variational literature is dedicated to two primary challenges. The first is designing a family $\mathcal{Q}$ that is flexible enough to enable a good approximation $q^\star \approx \pi$, but simple enough to enable fast i.i.d. draws and pointwise density evaluation. Historically, mean-field approximations have been the most popular class (i.e., $q_\lambda(\theta) = \prod_{i=1}^d q_{\lambda_i}(\theta_i)$ for multivariate parameter $\theta = (\theta_1, \dots, \theta_d)$) (e.g., [83, 204]). Modern research considers more flexible families, including normalizing flows, for which q_λ is the pushforward of a base distribution q_0 through a parametrized diffeomorphism T_λ [149]; variational autoencoders [91] and annealing families, which augment both the variational family $q_\lambda(\theta, z)$ and target $\pi(\theta, z)$ [176, 228, 191] and amortized families that specify $\lambda = \lambda_\beta(x)$ as a function of the data x with its own parameters β [91]. The second challenge is designing an optimization method for equation (4). Generally stochastic first-order methods are used (e.g., ADAM [89]) relying on unbiased stochastic gradient estimates based on Monte Carlo approximations of the objective in equation (4).

Theoretical Guarantees

Two important questions for analyzing variational procedures are (1) how accurate the approximation is in terms of properties of the *optimal* distribution in a family $\mathcal{Q}$ for a particular target π and (2) how quickly and reliably does a computational method find that optimum (or at the very least a local optimum). These two questions align with the two primary methodological challenges mentioned above.

Methodology for variational inference has developed rapidly over the past decade in both variational family and optimization algorithm design. However, the literature on theoretical analysis has only recently started making progress. The reason is that even for very simple families $\mathcal{Q}$ (e.g., multivariate Gaussians), the problem in equation (4) is a difficult-to-analyze nonconvex stochastic optimization. For modern flexible families, the challenge is even greater. In situations where there is an interpretable, empirical objective of downstream performance (e.g., predictive accuracy), the lack of theory is not a major barrier to use of variational inference but in the typical Bayesian inferential setting, convergence guarantees (both statistical and optimization-related) are crucial for reliable statistical inference. In Section 4, we discuss the theory of variational Bayes.

The paper now proceeds in four vignettes—on MH proposal design via deep learning in Section 3, theory of variational inference in Section 4, Bayesian coresets in Section 5 and distributed Bayesian inference in Section 6. Emphasis is placed on influential recent advances and important future work.

3. SAMPLING USING DEEP GENERATIVE MODELS

In this vignette, we discuss the use of deep generative models to design better proposal distributions for use in MH, both by augmenting existing kernels and by constructing entirely new distributions. Most deep generative models use a neural network (NN) to transform a simple-based distribution to closely match a prespecified empirical distribution. The setting of posterior computation via MH introduces two practical problems. First, samples from the target are not available prior to sampling, complicating the process of training the NN. Second, each iteration of MH requires computing the acceptance probability, hence evaluating the proposal density. If the proposal is a simple distribution transformed by an NN, then this requires inverting an NN, which is generally impossible, and computing the Jacobian, which can be numerically intractable in high dimensions. We focus on adaptively tuning normalizing flow (NF) proposals as a means of resolving these challenges. Section 3.1 introduces NFs; Sections 3.2–3.3 cover applications to MH and straightforward generalizations, Section 3.4 covers alternative uses of NFs, and Section 3.5 discusses exciting future research.

3.1 Introduction to Normalizing Flows

In this section, we provide a brief introduction to NFs and highlight their useful properties. One method for generating a flexible class of proposal distributions is to transform a simple D-dimensional random variable Z (e.g., $Z \sim N(0, I_D)$) with a diffeomorphism f parameterized by an NN. Carefully tuning f can result in proposals $Y = f(Z)$ that closely conform to the target. Computing the acceptance probability in each iteration of MH requires evaluating the proposal density,

$$\pi_Y(y) = \pi_Z(f^{-1}(y))|J_{f^{-1}}(f^{-1}(y))|, \tag{5}$$

where π_Z is the density of Z and $J_{f^{-1}}$ is the Jacobian of f^{-1}. Inverting NNs is generally intractable, and evaluating Jacobians is $O(D^3)$ in the worst case.

NFs impose additional structure on f to resolve these problems. Specifically, discrete NFs decompose f as the composition of K simple component functions:

$$f = f_K \circ \cdots \circ f_1. \tag{6}$$

Component functions are constructed to facilitate fast inversion (either exactly or approximately) and fast Jacobian calculations (e.g., by ensuring Jacobians are upper/lower triangular). The change of variables rule becomes

$$\pi_Y(y) = \pi_Z(f^{-1}(y)) \prod_{i=1}^{K} |J_{f_i^{-1}}(z_i)|, \tag{7}$$

where $f^{-1} = f_1^{-1} \circ \cdots \circ f_K^{-1}$ and $z_i = f_{i+1}^{-1} \circ \cdots \circ f_K^{-1}(y)$ with $z_K = y$. By the inverse function theorem, $J_{f_i^{-1}} = J_{f_i}^{-1}$, so it is sufficient to compute the Jacobian of f_i or f_i^{-1}. For example, a *planar* normalizing flow [168] uses component functions

$$f_i(z) = z + a_i h(w_i^T z + b_i), \tag{8}$$

where $a_i, w_i \in \mathbb{R}^D$, $b_i \in \mathbb{R}$ are parameters to be tuned and h is an invertible, differentiable nonlinearity applied elementwise. The matrix determinant lemma allows one to express the Jacobian as

$$|J_{f_i}(z)| = 1 + h'(w_i^T z + b)a_i^T w_i, \tag{9}$$

which is $O(D)$ to compute. Planar flows are not invertible for all choices of parameters and nonlinearities, however, efficient constrained optimization algorithms are available, which ensure invertibility [168]. Planar flows have relatively limited expressivity, and many layers may be needed to construct suitably complicated high-dimensional proposals. Improved component functions have been proposed, including radial [168], spline [47], coupling [43], autoregressive [90], etc. See [93] for a review of NFs and [149] for theory on the expressively of discrete flows.

Continuous normalizing flows [35] are an extension of the discrete framework, potentially enhancing expressivity while requiring fewer parameters and lower memory complexity. The key insight is to reconceptualize discrete NFs as a method for computing the path $x(t)$ of a particle at discrete times $t \in \{0, 1/K, 2/K..., 1\}$. The initial location $x(0)$ is drawn from Z. At time $1/K$, the location is updated to $x(1/K) = f_1(x(0))$. This is repeated iteratively, moving from $x(i/K)$ at time i/K to $x((i+1)/K) = f_i(x(i/K))$ at time $i + 1$. The result is a path $(x(0), \dots, x(1))$ where the final location is a sample from Y. Continuous NFs consider the limit $K \to \infty$, with the intuition that one can obtain a more flexible distribution for Y by flowing samples of Z through continuous paths instead of discrete paths. This can be formalized as the initial value problem

$$\frac{dx(t)}{dt} = f(x(t), t), \tag{10}$$

where f is a function parameterized by an NN and $x(0)$ is a sample from Z. In practice, equation (10) cannot be solved analytically, however, approximate samples of Y can be generated using an ODE solver. Euler's method with a step size of $1/K$ exactly recovers a discrete NF with K layers, but greater expressivity can be obtained using higher order solvers. This framework has a number of surprising technical advantages; see [35] for an exposition.

3.2 Normalizing Flow Proposals

In this section, we outline methods for constructing proposals with NFs. A NF with parameters ϕ will be denoted $f_\phi : \mathbb{R}^D \to \mathbb{R}^D$; this yields a new density $\hat{\pi}_\phi$ by pushing forward a simple random variable Z with density π_Z.

Independent proposals. The simplest approach is to use an NF to generate proposals in independent MH [19]. At each iteration, a proposed state θ' is generated by pushing a sample of Z through the NF. This state is accepted with probability (3) where $g(\cdot|\theta_t) = \hat{\pi}_\phi(\cdot)$. In high dimensions, almost all choices of ϕ will result in low overlap between $\hat{\pi}_\phi$ and π; hence small acceptance ratios and poor mixing. Consequently, we focus our discussion on more elaborate proposals, which result in better practical performance.

Dependent proposals. A more practical approach is to let proposals depend on the current state. This can be achieved by using a larger NF $f_\phi : \mathbb{R}^D \times \mathbb{R}^M \to \mathbb{R}^D \times \mathbb{R}^M$, which maps the current state x and M-dimensional noise z to a proposal θ' and transformed noise z'. The M-dimensional noise can be thought of as an auxiliary parameter, such as momentum or temperature in dynamics-based MCMC. Song, Zhao and Ermon [184] construct a dependent proposal that is symmetric, thus eliminating the ratio of proposal densities in equation (3) and reducing the problem of extremely low early acceptance rates. The proposal is constructed in two stages: first, sample $u \sim \text{Uniform}[0, 1]$ and z from Z. If $u > 0.5$, propose θ' using $(\theta', z') = f_\phi(x, z)$. Otherwise propose θ' using $(\theta', z') = f_\phi^{-1}(\theta, z)$. Using a mixture of f_ϕ and f_ϕ^{-1} ensures that θ' is as likely to be proposed when starting at θ as θ is to be proposed when starting at θ'. Key to the proof of symmetry is the assumption that the NF is volume-preserving. This is a restrictive assumption: current volume-preserving architectures are outperformed by nonvolume-preserving architectures.

Mixture kernels. Higher initial acceptance rates can be obtained by combining NF proposals with classical kernels, for example, by alternating proposing samples with Hamiltonian Monte Carlo (HMC) and a conditional flow. Samples from the classical kernel provide data with which to tune the NF. Eventually, the NF becomes a good approximation to the posterior, proposing efficient global moves and resulting in better mixing than the classical kernel alone. Gabrié, Rotskoff and Vanden-Eijnden [53] construct a proposal, which deterministically alternates between ten MALA proposals and one independent NF proposal. The resulting sampler efficiently explores multimodal distributions: MALA locally explores each mode, and NF teleports the chain between modes. It is critical to initialize the sampler with at least one particle in each mode, as the local dynamics are unlikely to discover new modes on their own. The algorithm is shown to converge with an exponential rate in the continuous time limit. Partial ergodic theory is available when the flow is adaptively learned by minimizing the KL divergence, although other loss functions remain unstudied.

Augmenting existing kernels. The previously discussed mixtures rely on classical kernels for local exploration until there is sufficient data to train the NF. An alternate approach is to use NFs to augment classical kernels, that is, to improve the classical kernel as the chain runs instead of tuning a separate auxiliary kernel. We use HMC as an example, wherein a new state θ' is proposed by drawing a momentum $\nu \sim N(0, I_D)$ and approximating the resulting Hamiltonian dynamics (usually) with the leapfrog integrator. One time step of the approximation proceeds by taking a half-step of the momentum

$$\nu_{1/2} = \nu - \frac{\varepsilon}{2}\nabla U(x), \tag{11}$$

which is then used to update the momentum,

$$\nu' = \nu - \frac{\varepsilon}{2}\nabla U(x'). \tag{12}$$

The process is repeated a prespecified number of times to generate a final proposal; the final momentum is disregarded. The resulting proposal is symmetric and volume-preserving, resulting in a simple acceptance ratio. Crossing low-probability regions requires a large velocity, which is unlikely if the momentum is sampled from a Gaussian. Levy, Hoffman and Sohl-Dickstein [98] use NFs to learn a collection of maps that dynamically rescale the momentum and position to encourage exploration across low-probability regions. Specifically, the momentum half-step is replaced by

$$\begin{aligned}\nu_{1/2} = {} & \exp(S_\nu(\theta)) \odot \nu \\ & - \frac{\varepsilon}{2}\exp(Q_\nu(\theta)) \odot \nabla U(\theta) + T_\nu(\theta),\end{aligned} \tag{13}$$

where $\odot$ is the elementwise product, S_ν is a NF that rescales the momentum, Q_ν is a NF that rescales the gradient and T_ν is a NF that translates the momentum. Similarly, the position update is replaced with

$$\begin{aligned}\theta' = {} & \exp(S_\theta(\nu_{1/2})) \odot \theta \\ & + \varepsilon \exp(Q_\theta(\nu_{1/2})) \odot \nu_{1/2} + T_\theta(\nu_{1/2}),\end{aligned} \tag{14}$$

where S_θ, Q_θ and T_θ are NFs. The momentum is updated again with equation (13) using θ' instead of θ, and the entire procedure is iterated. When all of these NFs are zero, we recover HMC exactly. Allowing the NFs to be nonzero results in a very flexible family of proposal distributions, which can be adaptively tuned to propel the sampler out of low probability regions by rescaling and translating the momentum/position. The invertibility and tractable Jacobians allow efficient calculation of the proposal density. This presentation has been simplified from [98], which also includes random directions, random masking and conditions NFs on the leapfrog iteration. So far, the above augmentation technique has only been applied to HMC. However, there is a broad class of dynamical systems that can be used to generate proposals, including Langevin dynamics, relativistic dynamics, Nose–Hoover thermostats and others [108]. NFs can be used to augment all of these algorithms using the same recipe as above.

3.3 Tuning Proposals

Appropriately tuning NF parameters is critical for good mixing. In practice, tuning is often performed by adaptively minimizing a loss. In this section, we cover a variety of candidate loss functions, including measure-theoretic losses, summary statistics and adversarial approaches.

Statistical deviance. The simplest approach is to define a function d measuring how close the proposal is to the target and then to find NF parameters minimizing $d(\hat{\pi}_\phi, \pi)$. Let $\mathcal{G}$ be a space of probability densities and $d : \mathcal{G} \times \mathcal{G} \to \mathbb{R}$ be any function measuring the distance/discrepancy/deviance between two probability measures. We assume:

1. $d(\rho, \rho) = 0$ for all $\rho \in \mathcal{G}$.
2. $d(\rho, \rho') > 0$ for $\rho \neq \rho' \in \mathcal{G}$, with equality interpreted as equality almost everywhere.
3. $d(\hat{\pi}_\phi, \pi)$ has a gradient with respect to ϕ, $\nabla_\phi d(\pi_\phi, \pi)$.

Conditions (1) and (2) ensure $d(\hat{\pi}_\phi, \pi) = 0$ if and only if $\hat{\pi}_\phi = \pi$, hence minimizing d is a reasonable way to approximate the target. Condition (3) allows optimization with gradient-based methods. Weaker notions of differentiability are sufficient, such as having a tractable subgradient.

For example, d may be the forward KL divergence,

$$\text{(15)} \qquad \mathrm{D}_{\mathrm{KL}}(\pi \| \hat{\pi}_\phi) = \int_{\mathbb{R}^D} \pi(\theta) \log\left(\frac{\pi(\theta)}{\hat{\pi}_\phi(\theta)}\right) d\theta$$

Adaptive estimation can be performed by alternating between generating a sample via MH and updating NF parameters using the gradient of (15) [19]. The gradient can be estimated via Monte Carlo using previous samples. Under technical assumptions on the NF and the target, the resulting Markov chain is ergodic with the correct limiting distribution [19].

Other viable choices for d include the Hellinger distance, the (sliced) Wasserstein distance, the total variation distance, etc. Many of these are as-of-yet unexplored as a means of adaptively estimating flows, and it is unclear which will result in the best performance. The main limitation with approaches in this class is that minimizing a difference only indirectly targets good mixing; in the following, we consider directly targeting good mixing with MCMC diagnostics.

Mixing summary statistics. A high quality global approximation of the target may not be required for sufficiently good mixing, especially if NFs are used in conjunction with local kernels such as HMC. Using distance-based losses in these situations are unnecessarily ambitious and better practical performance may be attained by switching to a loss function, which directly targets good mixing. Ideally, one would maximize the effective sample size, but this depends on the entire history of the chain and is in general slow to compute. Instead, [98] propose minimizing the lag-1 autocorrelation, which is equivalent to maximizing the expected squared jump distance [151]:

$$\text{(16)} \qquad \mathrm{lag}(\hat{\pi}_\phi, \pi) = E\big[\|\theta - \theta'\|_2^2 \mathrm{acc}(x, x')\big],$$

where the expectation is over the target and any auxiliary variables used to sample θ'. This can be estimated using samples θ_t, $t = 1, \dots, T$ from the first S iterations of the chain by generating a proposal θ_t' starting at each θ_t and averaging

$$\text{(17)} \qquad \mathrm{lag}(\hat{\pi}_\phi, \pi) \approx \frac{1}{T} \sum_{t=1}^{T} \|\theta_t - \theta_t'\|_2^2 \mathrm{acc}(\theta_t, \theta_t').$$

This loss depends on ϕ implicitly through the θ_t'. Naively optimizing this loss does not guarantee good mixing across the entire space, for example, the chain may bounce between two distant modes. To solve these problems, [98] add a reciprocal term and instead optimize

$$\text{(18)} \qquad \ell_\lambda(\hat{\pi}_\phi, \pi) = \frac{\lambda}{\mathrm{lag}(\hat{\pi}_\phi, \pi)} - \frac{\mathrm{lag}(\hat{\pi}_\phi, \pi)}{\lambda},$$

where $\lambda > 0$ is a tuning parameter. The reciprocal term penalizes states where the expected squared jump distance is small. Levy, Hoffman and Sohl-Dickstein [98] add a term of the same form to encourage faster burn-in. The composite loss is used to train an augmented variant of HMC and results in a sampler, which efficiently moves between well-separated modes.

Other summary statistics can be integrated into this framework, possibly considering lag-k autocorrelations or multiple chain summaries such as the Gelman–Rubin statistic [57]. One concern with this class of loss functions is that no single summary statistic can detect when a chain has mixed, and naively optimizing one statistic may result in pathological behavior that is hard to detect. In Section 3.5, we discuss different strategies that may strike a middle ground between ambitious distance-based methods and narrow summary statistic-based methods.

3.4 Alternatives to MCMC

Normalizing flows are useful proposal distributions in MH because they are flexible, allow exact evaluation of the likelihood and can be automatically tuned to have high overlap with the posterior. This also makes them useful as proposal distributions in alternative approaches for evaluating equation (1), such as importance sampling (IS) and sequential Monte Carlo (SMC) [192]. IS rewrites equation (1) as

$$E[f] = \int w(\theta) f(\theta) \hat{\pi}_\phi(\theta)\, d\theta,$$

where $w(\theta) = \pi(\theta|x)/\hat{\pi}_\phi(\theta)$ are importance weights; this allows estimation of $E[f]$ by sampling $\{\theta_t\}_{t=1}^T \sim \hat{\pi}_\phi$ and averaging $\{w(\theta_t) f(\theta_t)\}_{t=1}^T$. The weights can be self-normalized, that is, divided by $w(\theta_1) + \cdots + w(\theta_T)$ – to eliminate dependence on the normalizing constant of π. IS may be sensitive to the proposal distribution, with high variance weights when $\hat{\pi}$ proposes values, which are unlikely under π. SMC improves upon importance sampling by alternating resampling/selection and mutation steps: unlikely proposals are thrown away during the selection step, and then new proposals are generated using the survivors in the mutation step. SMC is a broad class of algorithms, which also frequently leverages other techniques, such as tempering [130]. Recently, NFs have been used to enhance these algorithms, leveraging many of the same fundamental tools discussed above [127, 54, 10, 36, 117]. In particular, SMC may be more efficient for sequential problems, such as those involving streaming data.

3.5 Open Questions and Future Directions

We have introduced several different kernel structures and losses, which can be combined to develop new adaptive MCMC algorithms. In this section, we discuss shortcomings of these approaches, as well as avenues for exciting long-term research.

Theoretical guarantees. So far, partial ergodic theory is only available in the simplest case of tuning an independent NF proposal by adaptively minimizing the KL divergence [19]. Dependent/conditional proposals and augmented kernels are not well studied, and no guarantees are available when adaptively minimizing summary statistic or adversarial-based losses. This is particularly concerning for summary statistic-based losses, as it is not clear that minimizing (e.g., lag-1 autocorrelation) is enough to guarantee ergodic averages converge to the correct values. Precise theoretical results will provide insights into when/why these methods succeed/fail, and are a necessary precursor to widespread adoption of NF sampling.

Adversarial training. Generative adversarial networks (GANs) [63, 69] pit two NNs against each other in a minimax game. The first player is a generator, which transforms noise into samples that look like real data; the second player is a discriminator, which tries to determine whether an arbitrary sample is synthetic or real. GANs may be applied to MCMC by taking the proposal distribution to be the generator and training a discriminator to distinguish between proposals and previous samples of the target. Song, Zhao and Ermon [184] use this idea to adaptively train a NF proposal, which dramatically outperforms HMC on multimodal distributions. Many improvements are possible by leveraging modern ideas from the GAN literature. Conditional GANs [124] allow the discriminator and generator to condition on external variables. For example, one could construct a tempered adversarial algorithm by conditioning on a temperature variable, possibly accelerating the mixing of annealed MCMC. Complicated GAN structures are prone to mode collapse, hence these generalizations will likely require modified loss functions [196, 115, 82, 212] and regularization [70, 154, 175, 125].

Constrained posteriors. In this vignette, we only consider the case where the target is supported over Euclidean space; however, in some applications the target is supported over a Riemannian manifold (e.g., the sphere or positive semidefinite matrices). Most manifold sampling algorithms rely on approximating dynamics defined either intrinsically on the manifold or induced by projecting from ambient space. These dynamics-based methods may be inferior to NF kernels for multimodal distributions. Recent work has successfully generalized NFs to Riemannian manifolds, although these constructions typically place significant restrictions on geometry (e.g., diffeomorphic to a cross-product of spheres [169]) or rely on high-variance estimates of Jacobian terms [116]. Loss functions measuring the distance between a proposal and the target may be harder to define and compute over manifolds. New architectures for manifold valued NFs and improved estimation techniques could facilitate efficient sampling in a wide class of models with non-Euclidean supports.

Our discussion also neglected to mention discrete parameters. Discrete parameters occur routinely in Bayesian applications, including clustering/discrete mixture models, latent class models and variable selection. Specific NF architectures have been constructed to handle discrete data [194, 233], but current approaches are relatively inflexible and cannot be made more flexible by naively adding more NN layers, limiting their utility within MH. A more promising direction is to leverage the flexibility of continuous NFs by embedding discrete parameters in Euclidean space and sampling from an augmented posterior. Several variants of HMC have been proposed to accommodate piecewise discontinuous potential functions [148, 126, 44], with recent implementations such as discontinuous HMC [141] achieving excellent practical performance sampling ordinal variables. However, embedding-based methods struggle to sample unordered variables—here the embedding order is arbitrary, with most embeddings introducing multimodality in the augmented posterior. NFs have successfully augmented continuous HMC [98] to handle multimodal distributions; the same strategy is promising for improving discontinuous HMC.

Automated proposal selection. A priori it is unclear which NF architecture, kernel structure and loss will result in the most efficient mixing for sampling a given posterior. Running many Markov chains with different choices can be time consuming, and a large amount of computational effort may be wasted if some chains mix poorly. Tools for automatic architecture/kernel/loss selection would greatly improve the accessibility of the proposed methodology. This goal is difficult in general given (1) the space of possible samplers is huge, (2) different architectures and kernels are not always comparable and (3) good mixing is impossible to quantify with a single numerical summary.

Ideas from reinforcement learning, sequential decision making and control theory could provide principled algorithms for exploring the space of possible samplers. One could define a state space of kernel/loss pairs, $(\hat{\pi}_\phi, L)$, which an agent interacts with by running adaptive MCMC. After each action, the agent observes sampler outputs such as trace plots and summary statistics. The goal is to develop a policy for choosing the next kernel/loss pair to run while maximizing some cumulative reward, such as cumulative effective sample sizes across all chains. As an initial attempt, one could restrict kernels to all have the same structure, such as HMC/NF mixtures where only the NF architecture is changing, and the loss function to be a simple parametric family, such as the lag-1 loss with different tuning parameters. This facilitates a parameterization of the state-space and allows application of existing continuous-armed bandit algorithms [2, 215]. Constructing a sequential decision-making algorithm that can efficiently explore kernel/loss pairs with fundamentally different kernel structures and loss functions is an open challenge, which will likely require better understanding of the theoretical relationships between the different proposed kernel structures as well as the dynamics, which result from minimizing different types of losses.

We expect broad patterns to emerge with increasing use of NF, with certain architectures/kernels/losses performing consistently well in specific classes of problems. For example, the authors have observed that discrete spline flows work very well for sampling from Gaussian mixture models. These heuristics could be collected in a community reference manual, allowing statisticians to quickly find promising candidate algorithms for their model class, dimension, features of the data, etc. Crowd-sourcing the construction and maintenance of this manual could enable statisticians to stay up-to-date with NFs, despite the rapid pace of ML research.

Accelerated tuning. The recipe presented in this vignette is to (1) choose a NF kernel structure, (2) choose a loss and (3) adaptively estimate parameters starting from a random initialization. Starting from a random initialization in step (3) is inefficient. Transfer/meta learning may provide tools for accelerating tuning by avoiding random initialization. For example, iterative model development and sensitivity analysis often involve repeating the same inferences with slightly different prior specifications. NF parameters estimated for one prior specification could be used to initialize the sampler for other prior specifications, potentially eliminating the need for adaptive tuning.

A more difficult task is handling targets with similar structures, but different dimensions. For example, consider a Bayesian sparse logistic regression model classifying Alzheimer's disease status using vectorized images of brains. Interest is in sampling coefficients β_I from the posterior $\pi(\beta_I|A, I)$ where $A = (A_1, \dots, A_n)$ is a set of disease indicators and $I = (I_1, \dots, I_n)$ is a set of brain images. Perhaps additional covariates for each subject are collected at a later stage, such as gene expression vectors $G = (G_1, \dots, G_n)$. Intuitively, there should be strong similarities between the updated posterior $\pi(\beta_I, \beta_G|A, I, G)$ and the original posterior $\pi(\beta_I|A, I)$; however, this is difficult to formalize because the posteriors have different dimensions.

A promising approach is to parameterize the initial sampler in a dimension-free manner, for example, by defining a kernel, which proposes an update for the ith coefficient depending only on the potential $U(\beta)$, the gradient in that direction $\partial_{\beta_i} U(\beta)$ and auxiliary variables in that direction. This kernel can be tuned while sampling $\pi(\beta_I|A, I)$ with any of the aforementioned loss functions, and then automatically applied to sample $\pi(\beta_I, \beta_G|A, I, G)$. Gong, Li and Hernández-Lobato [62] introduce a related idea for stochastic gradient sampling of Bayesian neural networks with different activation functions. The general methodology remains unstudied for exact sampling. The proposed coordinate-wise strategy cannot leverage correlation between pairs of parameters to propose efficient block updates; solutions to this problem constitute ongoing research.

4. THEORY OF VARIATIONAL BAYES

In this vignette, we discuss the theory of variational Bayesian (VB) methods. This section focuses specifically on (1) characterization of the frequentist properties of the VB posterior in terms of posterior contraction rates, uncertainty quantification and others and (2) convergence guarantees of VI algorithms.

Denote $x_n = x_{1:n}$ as the sample of the data. Recall that the *variational posterior* distribution is typically defined as

$$\widehat{Q} = \underset{Q \in \mathcal{Q}}{\operatorname{argmin}} \, \mathrm{D_{KL}}\big(Q \,\|\, \Pi(\cdot|x_n)\big). \tag{19}$$

In solving the optimization problem (19), one first writes $\Pi(d\theta|x_n) = p_\theta(x_n)\Pi(d\theta)/p(x_n)$, where $p(x_n) := \int p_\theta(x_n)\Pi(d\theta)$ is the joint density of x_n. The KL-divergence above can be written as

$$\begin{aligned} &\mathrm{D_{KL}}\big(Q \,\|\, \Pi(\cdot|x_n)\big) \\ &\quad = \int \log\Big(\frac{p(x_n)Q(d\theta)}{p_\theta(x_n)\Pi(d\theta)}\Big) Q(d\theta) \\ &\quad = \underbrace{-\int \log p_\theta(x_n) Q(d\theta) + \mathrm{D_{KL}}(Q\|\Pi)}_{=:\Psi(Q,\Pi,x_n):=-ELBO} \\ &\qquad + \log p(x_n). \end{aligned} \tag{20}$$

In the above, we let

$$\Psi(Q, \Pi, x_n) = -\int \log p_\theta(x_n) Q(d\theta) + \mathrm{D_{KL}}(Q\|\Pi),$$

which we call the *variational objective function.* This is also the negative of the *evidence lower bound (ELBO)*, where the ELBO is $\int \log p_\theta(x)Q(d\theta) - \mathrm{D_{KL}}(Q\|\Pi)$ which provides a lower bound of the "evidence" or the marginal likelihood $\log p(x)$ as seen from (20).

Since $p(x_n)$ is a constant with respect to Q, one has

$$\begin{aligned}\widehat{Q} &= \operatorname*{argmin}_{Q\in\mathcal{Q}} \Psi(Q,\Pi,x_n) \\ &= \operatorname*{argmin}_{Q\in\mathcal{Q}} \mathrm{D}_{\mathrm{KL}}\big(Q\|\Pi(\cdot|x_n)\big).\end{aligned} \tag{21}$$

Hence, minimizing the KL divergence between the variational family and the exact posterior distribution is equivalent to minimizing the variational objective $\Psi(Q,\Pi,x)$ or maximizing the ELBO.

4.1 Statistical Theory of Variational Bayes

As outlined in Section 2.2, it is important to understand properties of the variational posterior, including critically the quality of the approximation. One approach to studying the VB posterior is through investigating frequentist properties, including contraction rates, model selection consistency, and asymptotic normality (known as Bernstein–von Mises (BvM) theorems) of VB posteriors.

In the asymptotic regime, we assume data x_n are generated from $\mathsf{P}_{\theta^\star}^{(n)}$ and $n\to\infty$. The variational posterior

$$\widehat{Q}_n \in \operatorname*{argmin}_{Q\in\mathcal{Q}} \Psi(Q,\Pi,x_n),$$

is said to have the contraction rate ϵ_n if

$$\mathsf{E}_{\theta^\star}^{(n)}\big[\widehat{Q}_n\big(d(\theta,\theta^\star)\le A_n\epsilon_n\big)\big]\to 1 \tag{22}$$

as $n\to\infty$ for any diverging sequence $A_n\to\infty$. Here, $\mathsf{E}_{\theta^\star}^{(n)}$ denotes the expectation with respect to the true likelihood function, $\widehat{Q}_n(d(\theta,\theta^\star)\le A_n\epsilon_n)$ measures the variational posterior probability over the neighborhood $d(\theta,\theta^\star)\le A_n\epsilon_n$ of the true parameter θ^* with radius $A_n\epsilon_n$. If the contraction rate ϵ_n matches the *minimax optimal rate*, we say that the variational posterior distribution is optimal.

Recent work [6, 227, 224] provided theoretical conditions under which the variational posterior is optimal. These conditions imply that when the model is appropriately complex and the prior is sufficiently diffuse, which are standard conditions for establishing posterior contraction rates for the original posterior [58], then together with an assumption on the variational gap, the variational posterior distribution also has optimal contraction rates. The variational gap condition assumes there is $Q\in\mathcal{Q}$ such that

$$\int \mathrm{D}_{\mathrm{KL}}\big(\mathsf{P}_\theta^{(n)}\|\mathsf{P}_{\theta^\star}^{(n)}\big)Q(\mathrm{d}\theta)+\mathrm{D}_{\mathrm{KL}}(Q\|\Pi)\lesssim n\epsilon_n^2. \tag{23}$$

The left-hand side of (23) is an upper bound on the variational gap $\mathrm{D}_{\mathrm{KL}}(\hat{Q}\|\Pi(\theta|x_n)$. This upper bound is verified by ensuring that each term on the left is of order $O(n\epsilon_n^2)$. Alquier and Ridgway [6] formulate this variational gap condition as an extension of prior mass conditions. If one restricts the VB family to be in the same class as the prior and the parameters to lie in a neighborhood of the true parameter, this condition reduces to the standard prior mass condition.

In addition, [152] and [224] developed variational Bayes' theoretic frameworks that can deal with latent variable models. Alquier and Ridgway [6] investigated the contraction properties of variational fractional posteriors with the likelihood raised to a fractional power. There are several studies that derived contraction rates of variational posteriors for specific statistical models, for example, mixture models [37], sparse (Gaussian) linear regression [165, 223], sparse logistic linear regression [166] and sparse factor models [139].

4.2 Adaptive Variational Bayes

A novel and general variational framework for adaptive statistical inference on a collection of model spaces has been proposed [142]. The framework yields an *adaptive variational posterior* that has optimal theoretical properties in terms of posterior contraction and model selection while enjoying tractable computation.

In general, when performing statistical inference the "regularity" of the true parameter is unknown and adaptive inference aims to construct estimation procedures that are optimal with respect to the unknown true regularity. To do this, one typically prepares *multiple models* with different complexities, for example, sparse linear regression models with different sparsity, neural networks with different numbers of neurons or mixture models with different numbers of components and then selects among them. To achieve adaptivity, frequentists usually conduct (fully data-dependent) model selection before parameter estimation, for example, via cross-validation or penalization. There is some work on *Bayesian adaptation* by imposing hierarchical priors on a collection of model spaces [59].

Let $\mathcal{M}$ denote a set of model indices and $\{\Theta_m\}_{m\in\mathcal{M}}$ multiple disjoint (sub)models with different complexities. Let $\Theta_{\mathcal{M}}:=\bigcup_{m\in\mathcal{M}}\Theta_m$ be an encompassing model. A (hierarchical) prior is given as

$$\Pi=\sum_{m\in\mathcal{M}}\alpha_m\Pi_m,$$

where α_m is the prior probability of model Θ_m, $\sum_{m\in\mathcal{M}}\alpha_m=1$, and Π_m is the prior distribution of θ within model Θ_m.

Ohn and Lin [142] considers *variational Bayes adaptation* by approximating the posterior $\Pi(\cdot|x_n)$ under the above hierarchical prior, using a variational Bayes family over the encompassing model parameter space, using disjoint variational families $\{\mathcal{Q}_m\}_{m\in\mathcal{M}}$ over individual models with $\mathcal{Q}_m\subset\mathcal{P}(\Theta_m)$:

$$\mathcal{Q}_{\mathcal{M}}:=\Big\{\sum_{m\in\mathcal{M}}\gamma_m Q_m\Big|Q_m\in\mathcal{Q}_m\Big\}.$$

Algorithm 1 Adaptive variational Bayes

Input: data x_n, prior $\Pi = \sum_{m \in \mathcal{M}} \alpha_m \Pi_m$, variational families $\{\mathcal{Q}_m\}_{m \in \mathcal{M}}$.

- For every $m \in \mathcal{M}$, compute the variational posterior of the submodel Θ_m:

$$\widehat{Q}_{n,m} \in \operatorname*{argmin}_{Q \in \mathcal{Q}_m} \Psi(Q, \Pi_m, x_n). \tag{25}$$

- Compute the "optimal model weight" as

$$\widehat{\gamma}_{n,m} \propto \underbrace{\alpha_m}_{\text{prior}} \times \underbrace{\exp(-\Psi(\widehat{Q}_{n,m}, \Pi_m, x_n))}_{\text{goodness-of-fit of } \widehat{Q}_{n,m}} \tag{26}$$

for $m \in \mathcal{M}$

Return: The adaptive variational posterior

$$\widehat{Q}_n = \sum_{m \in \mathcal{M}} \widehat{\gamma}_{n,m} \widehat{Q}_{n,m}. \tag{27}$$

They show that the variational posterior

$$\widehat{Q}_n \in \operatorname*{argmin}_{Q \in \mathcal{Q}_{\mathcal{M}}} \Psi(Q, \Pi, x_n)$$

is of the form

$$\widehat{Q}_n = \sum_{m \in \mathcal{M}} \widehat{\gamma}_{n,m} \widehat{Q}_{n,m} \in \mathcal{Q}_{\mathcal{M}} \tag{24}$$

for some "mixing weight" $(\widehat{\gamma}_{n,m})_{m \in \mathcal{M}}$ and "mixture components" $\widehat{Q}_{n,m} \in \mathcal{Q}_m$ for $m \in \mathcal{M}$. The adaptive variational Bayes framework is summarized in Algorithm 1.

Computation of the adaptive variational posterior reduces to computing variational approximations for each individual model. The framework is general and can be applied for adaptive inference in many statistical models where multiple submodels of different complexities are available. The adaptive variational posterior has optimal contraction rates and strong model selection consistency when the true model is in $\mathcal{M}$. This theory has been applied to show optimal contraction for a rich variety of models, including finite mixtures, sparse factor models, deep neural networks and stochastic block models.

The above adaptive variational framework is not the first approach for adaptation under the variational Bayes paradigm. For example, [227] propose to use the variational posterior corresponding to the submodel having the highest variational model probability. Their methods have been applied to the Gaussian sequence model and finite-mixture models for adaptive density estimation, among others. Zhang and Gao [227] show that an empirical Bayes posterior with hyperparameters chosen in a data-driven way, such as maximizing the marginal likelihood, can be regarded as an variational approximation to a posterior in a hierarchical model. This connection allows the derivation of posterior contraction rates for the general empirical Bayes posterior considered in [227]. Their framework has been applied to adaptive density estimation, adaptive sparse high-dimensional linear regression and so on.

4.3 Convergence of Variational Bayes Algorithms

The above subsections focus on obtaining theoretical guarantees for variational Bayes from the lens of frequentist statistical theory. An alternative theoretical direction focuses on showing convergence of VB algorithms to the optimum (or a local optimum) as measured by the KL divergence. This subsection surveys recent developments.

Computational algorithms for solving (21) depend on the original posterior distribution as well as the variational family. When the variational family has certain simple structure, in particular, the so-called *mean field class*, there are efficient computational algorithms for finding $\hat{Q}$, based on the well-known *CAVI (coordinate ascent variational inference)* algorithm [83, 219], which guarantees convergence to a local minimizer [16]. As mentioned earlier, the mean-field class imposes posterior independence as

$$Q(\theta_1, \dots, \theta_d) = \prod_{j=1}^{d} Q_j(\theta_j), \tag{28}$$

where Q_j is a distribution for θ_j. By taking the derivative of the ELBO with respect to each of the $Q_j(\theta_j)$, one can arrive at the following coordinate ascent update:

$$\begin{aligned} \widehat{Q}_j(\theta_j) &\propto \exp\bigl(E_{Q_{-j}}[\log p(\theta_j | \theta_{-j}, x_n)]\bigr) \\ &\propto \exp\bigl(E_{Q_{-j}}[\log p(\theta_j | \theta_{-j}, x_n)]\bigr), \end{aligned} \tag{29}$$

where $\theta_{-j} = (\theta_1, \dots, \theta_{j-1}, \theta_{j+1}, \dots, \theta_n)$, and the expectation $E_{Q_{-j}}$ is taken with respect to all variational distributions excluding the jth component.

When a statistical model has latent structures, such as finite mixture models, topic models and stochastic block models, there are typically latent variables for every sample. The CAVI algorithm is inefficient in such cases, as it requires sweeping through the whole data set before updating the variational parameters at each iteration. *Stochastic variational inference* [75] is a popular alternative in this setting. Stochastic variational inference employs stochastic gradient descent by computing the gradient of the ELBO based on mini batches, leading to efficiency gains.

The BBVI (black box variational inference) algorithm proposed in [162] is a general method, which requires very little model specific derivation or analysis. The key idea is to optimize the variational objective stochastically using noisy estimates of the gradient of the ELBO. These estimates are obtained by rewriting the gradient of the ELBO as an expectation with respect to the variational posterior, which can then be approximated via Monte Carlo. As with other stochastic methods, reducing variance is key for improving efficiency. Several strategies have been proposed in [162] for controlling the variance. One technique is based on Rao–Blackwellization [28], which computes the gradient of ELBO with respect to one variable using iterative conditional expectations. The second strategy for variance reduction utilizes control variates [173, 147]. A control variate of a function is a family of functions with equivalent expectation. The idea is to choose a control variate that has smaller variance. In tailoring the method for BBVI, one can use control variates to estimate each entry of the Rao–Blackwellized gradient, further reducing the variance.

There has been recent effort in establishing convergence guarantees for BBVI as well as general stochastic variational inference algorithms [87, 45]. Kim et al. [87] provides convergence guarantees for BBVI, which hold for log-smooth posterior densities with and without strong log concavity and location-scale variational families. They also provide convergence guarantees for proximal BBVI for strongly log concave posteriors. Domke, Garrigos and Gower [45] obtain convergence guarantees for BBVI with log concave and log Lipschitz smooth target densities providing analyses of complexity of a proximal stochastic gradient method and a projected stochastic gradient algorithm. They additionally show that gradient estimators based on reparameterization satisfy a quadratic noise bound for Gaussian families. Convergence analysis of BBVI is made challenging by gradient estimators with unusual noise bounds and a composite nonsmooth objective.

There are interesting recent developments in VI that attempt to provide convergence guarantees in the finite-sample regime while bypassing nonconvex optimization. Two results along these lines are variational annealed importance sampling [228, 191, 56] and MixFlows [221]; both derive their guarantees from the use of (nearly) invariant Markov kernels.

4.4 Variational Auto Encoders (VAEs)

Variational inference, as considered above, has been most widely used as a framework for approximating the posterior distribution in a standard Bayesian model, that is, with a fixed prior distribution and a specified likelihood. There are, however, some other important and interesting settings beyond the standard Bayesian paradigm where VI is used, with the variational auto encoder (VAE) [91, 168] emerging as one of the most important modern examples.

VAEs are one of the primary likelihood-based training methods for deep generative models, and have achieved tremendous success in modeling high-dimensional data such as images and videos [65, 205]. A statistical theory of deep generative models is now emerging [29, 189]. One typically assumes that a D-dimensional observation X is generated as $X = G(Z) + \varepsilon$, where Z is a d-dimensional latent variable drawn from known distribution $P_{\mathbf{Z}}$ and ϵ is a noise vector independent of Z. Here, $G : \mathbb{R}^d \to \mathbb{R}^D$ is an unknown function parameterized by a deep neural network called the *deep generator*. In a VAE, the distribution $P(X|Z)$, parameterized by a deep neural network, is called the *decoder* and $P(Z|X)$ is approximated by a variational family $q_\lambda(Z)$ where λ is parameterized or amortized by another deep neural network, called the *encoder*. [92] provide an introduction to VAEs. Theoretical analysis of VAEs from the lens of M-estimation [188], robustness analysis [26] and connection to harmonic analysis [25] have been conducted.

4.5 Open Questions and Future Directions

Uncertainty quantification of the VB posterior. It is well known that variational posteriors tend to underestimate uncertainty of the true posterior, so a central open question is how one can construct computationally efficient VB posteriors producing (a) credible balls with valid frequentist coverage and/or (b) posterior covariance matching that of the true posterior.

There is limited work on theory justifying statistical inference using the variational posterior, including credible intervals and hypothesis testing. A natural direction in this regard is theorems on limiting forms of variational posteriors as the sample size $\to \infty$, along the lines of Bernstein–von Mises (BvM) theorems guaranteeing that the exact posterior distribution converges to a Gaussian distribution under certain regularity conditions. An initial promising result along these lines is [216], but there is need for new research for broad classes of models and corresponding variational families.

Some successful attempts in achieving improved uncertainty quantification have been recorded for specific models, such as Gaussian models or Gaussian process regression, either through proper turning of the mean-field VB method [61] or adopting nonmean-field VB families [197, 85, 137].

Theoretical guarantees of gradient-based algorithms. Existing theoretical guarantees for VB only apply to the global solution of the variational optimization problem. In practice, this optimization problem tends to be highly nonconvex and algorithms are only guaranteed to converge to local optima. For certain variational families and model classes, these local optima can be dramatically different, so that there is a large sensitivity to the starting point of the algorithm. It is of critical importance to obtain guarantees on the algorithms being used and not just on inaccessible global optima. For example, can one obtain general theoretical guarantees for gradient-based black-box variational inference with or without warm-start conditions? Kim et al. [87] and [45] are notable recent advancements in this direction.

There is a parallel and growing literature on nonconvex optimization in other contexts, including providing reassurance that local optima can be sufficiently close in some cases [118, 52, 105, 100, 138]. However, to the best of our knowledge, there is no such work on theoretical aspects of local optima produced by variational Bayes.

VB based on generative models. Richer variational families can be constructed using deep generative models such as normalizing flows [168, 103]. Due to their impressive flexibility, the resulting variational posterior can approximate a very wide class of target posteriors accurately. Despite its practical usefulness and strong empirical performance, there is no theoretical support for such approaches, for example, providing upper bounds on the variational approximation gap or concentration properties. Choosing the neural network architecture and algorithmic tuning parameters involved in training to maximize computational efficiency and accuracy of posterior approximation is an additional important related area that may benefit from better theoretical understanding.

Online variational inference. Given a prior distribution on an unknown parameter, the posterior distribution can be understood as an updated belief after observing the data. The updated posterior distribution can be used as a prior distribution when new data arrive. The process can be repeated many times for analyzing streaming data [121, 60, 88, 80]. At each step, the VB posterior can be used as a new prior for computational convenience [104, 106, 135]. Sequential VB posteriors have shown strong computational promise, but have yet to receive significant theoretical attention.

5. BAYESIAN CORESETS

At its core, the problem of working with large-scale data efficiently is how to exploit *redundancy*. To draw principled conclusions about a data set based on a small fraction of examples, one must rule out the presence of unique additional information in the (vast) remainder of unexamined data. One approach incorporates redundancy directly into its formulation: *Bayesian coresets* [78]. The key idea is to represent the large-scale data by a small, weighted subset that can be passed to any standard inference algorithm, providing posterior inference at a reduced cost. Coresets have a long history in computational geometry and optimization; see, for example, [1, 11, 49].

Coresets have a number of compelling advantages. First, and perhaps most importantly, coresets preserve important model structure. If the original posterior exhibits symmetry, weak identifiability, discrete variables, heavy tails, low-dimensional subspace structure, etc., the coreset posterior typically will exhibit that same structure, because it is constructed using the same prior and likelihood terms. This makes coresets appealing for use in complex models where a Gaussian asymptotic assumption is inappropriate. Second, coresets are composable: coresets for two data sets can often be combined trivially to form a coreset for the union of data sets [51]. This makes coresets naturally applicable to streaming and distributed contexts [24], Section 4.3. Third, coresets are inference algorithm-agnostic: once built, a coreset can be passed to most inference methods—in particular, exact MCMC methods with guarantees—with enhanced scalability. Finally, coresets tend to come with guarantees relating the size of the coreset to the quality of posterior approximation.

In this vignette, we will cover the basics of Bayesian coresets as well as recent advances in Sections 5.1 and 5.2, and discuss open problems and exciting directions for future work in Section 5.3.

5.1 Introduction to Bayesian Coresets

5.1.1 *Setup.* We are given a target probability density $\pi(\theta)$ for $\theta \in \Theta$ that is comprised of N potentials $(f_n(\theta))_{n=1}^N$ and a base density $\pi_0(\theta)$,

$$\pi(\theta) = \frac{1}{C} \exp\left(\sum_{n=1}^{N} f_n(\theta)\right) \pi_0(\theta), \tag{30}$$

where the normalization constant C is not known. This setup corresponds to a Bayesian statistical model with prior π_0 and i.i.d. data x_n conditioned on θ, where $f_n(\theta) = \log p(x_n|\theta)$. The goal is to compute or approximate expectations under π; in the Bayesian scenario, π is the posterior distribution.

A key challenge arises in the large N setting. Bayesian posterior computation algorithms tend to become intractable. For example, MCMC typically has computational complexity $\Theta(NT)$ to obtain T draws, since $\sum_n f_n(\theta)$ (and often its gradient) needs to be evaluated at each step. In order to reduce this $\Theta(NT)$ cost, *Bayesian coresets* [78] replace the target with a surrogate density

$$\pi_w(\theta) = \frac{1}{C(w)} \exp\left(\sum_{n=1}^{N} w_n f_n(\theta)\right) \pi_0(\theta), \tag{31}$$

where $w \in \mathbb{R}^N$, $w \geq 0$ are a set of weights, and $C(w)$ is the new normalizing constant. If w has at most $M \ll N$ nonzeros, the $\Theta(M)$ cost of evaluating $\sum_n w_n f_n(\theta)$ (and its gradient) is a significant improvement upon the original $\Theta(N)$ cost. The goal is then to develop an algorithm for coreset construction, that is, selecting the weights w that:

1. produces a small coreset with $M \ll N$, so that computation with π_w is efficient;

2. produces a high-quality coreset with $\pi_w \approx \pi$, so that draws from π_w are similar to those from π and
3. runs quickly, so that building the coreset is actually worth the effort for subsequent fast draws from π_w.

These three desiderata are in tension with one another. The smaller a coreset is, the more "compressed" the data set becomes, and hence the worse the approximation $\pi_w \approx \pi$ tends to be. Similarly, the more efficient the construction algorithm is, the less likely we are to find an optimal balance of coreset size and quality with guarantees.

5.1.2 *Approaches to coreset construction.* There are three high-level strategies that have been used in the literature to construct Bayesian coresets.

Subsampling. The baseline method is to uniformly randomly pick a subset $\mathcal{I} \subseteq \{1, \dots, N\}$ of $|\mathcal{I}| = M$ data points and give each a weight of N/M, that is,

$$(32)\qquad w_n = \frac{N}{M} \quad \text{if } n \in \mathcal{I}, \qquad w_n = 0 \quad \text{otherwise},$$

resulting in the unbiased potential function approximation

$$(33)\qquad \sum_{n=1}^{N} f_n(\theta) \approx N\left(\frac{1}{M}\sum_{m\in\mathcal{I}} f_m(\theta)\right).$$

This method is simple and fast, but typically generates poor posterior approximations. Constructing the subset by selecting data with nonuniform probabilities does not improve results significantly [78]. Empirical and theoretical results hint that in order to maintain a bounded approximation error, the subsampled coreset must grow in size proportional to N, making it a poor candidate for efficient large-scale inference. Coresets therefore generally require more careful optimization.

Sparse regression. One can formulate coreset construction as a sparse regression problem [24, 23, 229],

$$w^\star = \operatorname*{argmin}_{w\in\mathbb{R}^N_+} \left\| \sum_{n=1}^{N} f_n - \sum_{n=1}^{N} w_n f_n \right\|^2 \quad \text{s.t.} \quad \|w\|_0 \le M,$$

where $\|\cdot\|$ is some functional (semi)norm, and $\|w\|_0$ is the number of nonzero entries in w. This optimization problem can be solved using iterative greedy optimization strategies that provably, and empirically, provide a significant improvement in coreset quality over subsampling methods [24, 23, 229]. However, this approach requires the user to design—and tends to be quite sensitive to—the (semi)norm $\|\cdot\|$ and so is not easy to use for the general practitioner. The (semi)norm also typically cannot be evaluated exactly, resulting in the need for Monte Carlo approximations with error that can dominate any improvement from more careful optimization.

Variational inference. Current state-of-the-art research formulates the coreset construction problem as variational inference in the family of coresets [22],

$$(34)\qquad w^\star = \operatorname*{argmin}_{w\in\mathbb{R}^N_+} \mathrm{D}_{\mathrm{KL}}(\pi_w \| \pi) \text{s.t.} \|w\|_0 \le M.$$

Unlike the sparse regression formulation, this optimization problem does not require expert user input. However, it is not straightforward to evaluate the KL objective,

$$(35)\qquad \begin{aligned} &\log C - \log C(w) \\ &\quad + \sum_{n=1}^{N} (w_n - 1) \int \pi_w(\theta) f_n(\theta)\, \mathrm{d}\theta, \end{aligned}$$

even up to a constant in w. The difficulty arises because equation (35) involves both the unknown normalization constant $C(w)$ and an expectation under π_w, from which we cannot in general obtain exact draws. This is unlike a typical variational inference problem, where the normalization of the variational density is known and obtaining draws is straightforward. Current research on coreset construction is generally focused on addressing these issues; this is an active area of work, and a number of good solutions have been found [22, 114, 79, 129, 34, 113].

5.2 Notable Recent Advances

The literature on Bayesian coresets is still in its early stages, and the field is developing quickly. We highlight some key recent developments here.

Coreset data point selection. Optimization-based coreset construction methods have tended to take a "one-at-a-time" greedy selection strategy to building a coreset, thus requiring a slow, difficult to tune inner-outer loop [24, 22]. Recent work [34, 129, 79] demonstrates coresets can be built without sacrificing quality by first uniformly subsampling the data set to select coreset points, followed by batch optimization of the weights. This is both significantly simpler and faster than past one-at-a-time selection approaches, while providing theoretical guarantees: for models with a strongly log-concave or exponential family likelihood, after subsampling, the KL divergence of the *optimally-weighted* coreset posterior converges to 0 as $N \to \infty$ as long as the coreset size $M \gtrsim \log N$ [129]. This guarantee does not say anything about whether one can *find* the optimal weights, but just that selecting coreset data points by subsampling does not limit achievable quality.

Optimizing the KL divergence. Given a selection of coreset points, there remains the problem of optimizing the KL objective over the coreset weights w; this is challenging because one cannot obtain exact draws from π_w, or compute its normalization constant. It is possible to use

MCMC to draw from π_w, and to circumvent the normalization constant issue by noting that derivatives are available via moments of the potential functions under π_w, for example,

$$\begin{aligned}&\frac{\partial}{\partial w_n}\mathrm{D_{KL}}(\pi_w\|\pi)\\ &\quad= -\operatorname{Cov}_w\left[f_n(\theta), \sum_{i=1}^{N}(1-w_i)f_i(\theta)\right],\end{aligned} \tag{36}$$

where Cov_w denotes covariance under π_w [22, 129, 33]. First-order methods interleave a single step of MCMC with each weight optimization step [33], while second order methods use many steps of MCMC per weight optimization step [129].

Another promising approach is to use a surrogate variational family that is parametrized by the coreset weights w but enables tractable draws and exact normalization constant evaluation [34, 79, 113]. For example, Chen, Xu and Campbell [34] propose using a variational surrogate family q_w such that for all w, $q_w \approx \pi_w$, and then optimizing the surrogate objective function

$$w^\star = \operatorname*{argmin}_{w} \mathrm{D_{KL}}(q_w\|\pi). \tag{37}$$

Chen, Xu and Campbell [34] set q_w to be a normalizing flow based on sparse Hamiltonian dynamics targeting π_w. Concurrent work by Jankowiak and Phan [79] proposes a similar idea, but based on variational annealed importance sampling [176]. In either case, the optimization problem is then just a standard KL minimization over parameters w. Manousakas, Ritter and Karaletsos [113], in contrast, propose using a generic variational family q_λ with an auxiliary parameter λ to take draws, and adds an additional penalty to the optimization objective to tune q_λ to approximate π_w:

$$w^\star, \lambda^\star = \operatorname*{argmin}_{w,\lambda} \mathrm{D_{KL}}(\pi_w\|\pi) + \mathrm{D_{KL}}(q_\lambda\|\pi_w). \tag{38}$$

The unknown normalization constant on π_w cancels in the two KL divergence terms, and the $\mathrm{D_{KL}}(\pi_w\|\pi)$ term is estimated using self-normalized importance sampling based on draws from q_λ. Manousakas, Ritter and Karaletsos [113] use a diagonal-covariance Gaussian family for q_λ, and use an inner-outer loop optimization in which the inner loop optimizes λ to help ensure that q_λ remains close to π_w.

These two approaches are strongly connected. Consider the optimal auxiliary parameter

$$\lambda^\star(w) = \operatorname*{argmin}_{\lambda} \mathrm{D_{KL}}(q_\lambda\|\pi_w), \tag{39}$$

and assume that the family q_λ is flexible enough such that $q_{\lambda^\star(w)} = \pi_w$ for all w. Then the two approaches are equivalent if we define $q_w = q_{\lambda^\star(w)}$:

$$\mathrm{D_{KL}}(\pi_w\|\pi) + \mathrm{D_{KL}}(q_{\lambda^\star(w)}\|\pi_w) = \mathrm{D_{KL}}(q_w\|\pi). \tag{40}$$

The advantage of using a generic family q_λ is that it is much easier (and more flexible) than being forced to design a family q_w satisfying $q_w \approx \pi_w$. But self-normalized importance sampling is well known to struggle [31] even when the reverse KL divergence is small, and we still need to take draws from π_w once the coreset is built. The approach of directly designing q_w requires more up front effort, but the optimization is well behaved, and one can obtain i.i.d. draws directly from q_w afterward.

A comparison of current state-of-the-art algorithms—first- and second-order methods with draws from π_w using MCMC [33, 129], direct surrogate variational methods with $q_w \approx \pi_w$ [34] and parametrized surrogate variational methods using $q_\lambda \approx \pi_w$ [113]—has not yet been fully explored and is a direction for future research.

Optimization guarantees. Although variational inference in general is nonconvex, the coreset variational inference problem equation (34) facilitates guarantees. In particular, Naik, Rousseau and Campbell [129] obtain geometric convergence to a point near the optimal coreset via a quasi-Newton optimization scheme:

$$\|w_k - w_k^\star\| \le \eta^k\|w_0 - w_0^\star\| + \delta, \tag{41}$$

where w_k is the kth iterate, and $w_k^\star$ is its projection onto a subset of optimal coreset weights (the optimum may not be unique). The constants η and δ are related to how good of an approximation the *optimal* coreset is. If the optimal coreset is exact, then $0 < \eta < 1$ and $\delta = 0$. Note that this guarantee assumes exact quasi-Newton steps, which must be estimated via MCMC and data subsampling in practice; Chen and Campbell [33] recently improved this result to account for the use of both MCMC and subsampling.

5.3 Open Questions and Future Directions

Recent advances in coreset construction methods and theory have paved the way for a variety of new developments. In this section, we highlight important open problems and areas for investigation.

Complex model structure, data and symmetry. The coresets methodology and theory is now starting to coalesce for the basic model setup in equation (30) with a finite-dimensional parameter and conditionally i.i.d. data. Many popular models do not fit into this framework, such as certain network models [77], continuous time Markov chains [7], etc. Even some models that technically do fit in the framework of equation (30), such as certain hierarchical models [15], may be better summarized if more of their latent structure is exposed to the coreset construction algorithm. Some of these models involve $O(N^2)$ or larger computational cost, and would greatly benefit from a summarization approach.

Moving beyond the conditionally i.i.d. data setup, we advocate thinking about this problem as *model and data* *summarization*, broadly construed, as opposed to just the specific case of coresets. At an abstract level, Bayesian coresets are just one particular example of how one can construct a computationally inexpensive parametrized variational family π_w that provably contains (a distribution near) the true posterior π. In general, there is no reason this has to be associated with a sparse, weighted subset of data; we could, for example, summarize networks with subgraphs [144], summarize high-dimensional data with low-dimensional sketches [111], summarize expensive, complicated neural network structures with simpler ones [143], summarize expensive matrices with low-rank randomized approximations [218], etc. The key question is how to extend coresets, or summarization more broadly, to more sophisticated models beyond equation (30).

We believe that the key to answering this question is to understand the connections between Bayesian coresets, subsampling, probabilistic symmetries and sufficiency in statistical models; see, for example, [41, 96, 145]. Indeed, the fact that Bayesian coresets work at all is a reflection of the fact that one can use a small subset of data potentials as "approximately sufficient statistics," combined with the symmetry of their generating process. Assuming a fruitful connection is made, we expect that current Bayesian coreset construction methods—which are based on subsampling to select a "dictionary" of potentials, followed by optimization to tune the approximation—will serve as a good template in more general models.

Improved surrogates and optimization. Early Bayesian coresets literature [78, 23, 22, 24] suffered from the requirement of taking draws from π_w both during and after construction. Sampling *during* construction poses a particular challenge: if one intends to use MCMC to take draws from π_w, one needs to continually adapt the MCMC kernel to a changing target π_w as the weights w are refined. Recent developments discussed in Section 5.2 suggest that an easier way to approach the problem is to construct a tractable variational family q_w such that $q_w \approx \pi_w$ for all weights w—whether that is a normalizing flow [34], a variational annealed importance distribution [79], or an optimized parametric surrogate [113]—and then to tune the weights w so that $q_w \approx \pi$. The benefit of this approach is the ability to take exact i.i.d. draws and evaluate the density, which circumvents challenges of adaptive in-the-loop MCMC tuning. A major question is how to construct tractable, summarization-based variational families such that $q_w \approx \pi_w$ for all w.

For methods based on parametric surrogates [113] that set $q_w = q_{\lambda^\star(w)}$, where $\lambda^\star(w) = \operatorname{argmin}_\lambda \mathrm{D}_{\mathrm{KL}}(q_\lambda \| \pi_w)$, there are two major avenues for improvement. The first—and more likely achievable—goal is in the optimization of the parametric surrogate. In particular, the methodology currently involves slow inner-loop optimization of the surrogate, as well as potentially high-variance gradient estimates based on self-normalized importance sampling. Handling these two issues would be a major step forward for this approach. The second important area for future work—which may be far more challenging—is to provide theoretical guarantees on the quality of the coreset that is constructed using this method. The primary difficulty is that the surrogate optimization is as hard to analyze as other generic variational inference problems.

For methods based on direct surrogates [79, 34] where $q_w \approx \pi_w$ for all w, there are again two major areas for improvement. First, current methods involve Hamiltonian dynamics, and so are limited in scope to models with multidimensional real-valued variables; future work should extend these methods to models with a wider class of latent variables. The second area is once again to obtain rigorous theoretical guarantees on the quality of the surrogate. This is likely to be much easier than in the general parametric surrogate case above, as q_w is designed to approximate π_w directly, as opposed to just being a stationary point of a nonconvex optimization problem.

Privacy, pseudo-data and distributed learning. Distributed (or federated) learning, discussed in the next vignette, is a task in which data are stored in separate data centers and the goal is to perform global inference given all the data under the constraint that the data are not transmitted between centers. Both exact [38, 30] and approximate [178, 18] methods exist to perform Bayesian inference in this setting. A common additional constraint is that the data within each center are kept private, in some sense, from the other centers.

Coresets provide a potentially very simple solution to the distributed learning problem (both standard and privacy-preserving). In particular, coresets are often *composable*: if one builds subcoresets (independently and without communication) for subsets of a data set, one can combine these trivially to obtain a coreset for the full data set [49]. Coresets have also been extended to the privacy-aware setting, where one either trains pseudopoints with a differentially private scheme [114] or appropriately noises the coreset before sharing [50]. Subsequently, the data centers can share their privatized summaries freely with one another, or with a centralized repository that performs inference. There is some initial work on distributed Bayesian coresets constructed via sparse regression techniques [24], Section 4.3, but this work was done prior to the advent of modern construction methods. A natural next step would be to develop theory and methods for distributed (privacy-preserving) Bayesian coresets, leveraging recent advances in coreset construction.

Amortized and minimax coreset construction. Bayesian coresets are currently constructed in a model-specific manner by minimizing the KL objective in equation (34). In situations where multiple models are under consideration in exploratory analysis or sensitivity analysis, for example, one would need to retune the coreset weights for each model under consideration. Given that these retuning problems all involve the same data, they should be closely related but it is currently an open question how to construct multiple related coresets efficiently.

One potential direction of future work is to formulate a minimax optimization problem that is similar to equation (34), but where there is an outer maximization over a set of candidate models. A major question along these lines is whether it is actually possible to summarize a data set with a single coreset of $M \ll N$ data points such that the coreset provides a reasonable approximation for the worst-case model under consideration. Another possible way to tackle the problem is to amortize the cost of multiple coreset construction, in the spirit of *inference compilation* [97]. Rather than constructing individual coresets, we train a "coreset construction artifact:" a function that takes as input a candidate model and data subsample, and outputs a set of coreset weights. In other words, we *learn how to construct coresets*. The most likely candidate for such an artifact is a recurrent deep neural network, as is commonly-used in methods like inference compilation. A major question about this direction to consider is in which data analysis scenarios the cost of building such an artifact is worth the subsequent fast generation of coreset weights.

High-dimensional data and models. The coresets approach is designed with a focus on large-scale problems in the sense of the number of data points, N. But in practice, modern large-scale problems tend to also involve high-dimensional data and latent model parameters; the dimension may even grow with N. Empirical results have shown that coresets can be effective in problems with 10–100-dimensional data and parameters, while *pseudo*coresets [114, 113]—summaries via optimized synthetic pseudodata points, similar to inducing points for Gaussian processes [183]—have been used successfully on larger problems with 60,000-dimensional parameters and 800-dimensional data. But theoretical results on when we expect (pseudo)coresets to work well in high-dimensional settings are limited.

We begin with a negative (albeit pathological) example. When a large fraction of the potential functions $(f_n)_{n=1}^N$ encode unique information in the posterior, the coresets approach breaks down; it is not possible to maintain a good posterior approximation upon removing potentials. Manousakas et al. [114], Proposition 1, makes this intuition precise with a simple example. In a d-dimensional Gaussian location model with prior $\theta \sim \mathcal{N}(0, I)$, likelihood $\mathcal{N}(\theta, I)$, and data generated via $x_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, I)$, the *optimal* coreset of any size $M < d$ satisfies

$$\mathrm{D_{KL}}(\pi_{w^\star} \| \pi) \gtrsim d \quad \text{as } d \to \infty, \tag{42}$$

with high probability.[1] In some sense, this is unsurprising; the Gaussian location model with large d, despite its mathematical simplicity, is a worst-case scenario for data summarization, as one needs at least d potential functions f_n to span a d-dimensional space.

But in practice, high-dimensional data do not typically exhibit this worst-case behavior; they often instead exhibit some simpler, lower-dimensional structure. Developing (pseudo)coreset methods that take advantage of that structure is a key step needed to make summarization a worthwhile approach in large-scale modern problems. Furthermore, assuming that the coreset size should generally increase with dimension, additional work is needed to understand how the difficulty of the stochastic weight optimization scales. It is worth investigating whether the recently developed literature on data distillation in deep learning [211] contains any insights applicable to the Bayesian setting.

Improved automation and accessibility. Recent advances in research have, for the first time, made coresets a practical approach to efficient Bayesian computation. However, there is still much work to do to make their use possible by nonexperts. First and foremost, there is a need to develop a general, well-engineered code base that interfaces with common probabilistic programming libraries like Stan and Turing [27, 55]. In addition, there is a need for automated methods to (a) select coreset weight optimization tuning parameters, (b) select coreset size and (c) assess and summarize the quality of the coreset.

Other divergences. Currently, all variational coreset construction approaches optimize the reverse Kullback–Leibler divergence. A straightforward direction for future work would be to investigate the effect of using alternative divergences, for example, the Rényi divergence [102] or χ^2 divergence [42], in equation (34). These will all likely pose similar issues with the unknown normalization constant $Z(w)$, but divergences other than the reverse KL may provide coresets with distinct statistical properties.

Other construction algorithms. Coresets are an area of active development outside of Bayesian statistics [11, 49], and recent techniques from that literature (e.g., halving [155]) may be helpful in the Bayesian context. However, a point of caution: a good coreset for optimization may be a disastrously bad coreset for Bayesian inference,

[1] The result by Manousakas et al. [114] is stated in terms of the inverse CDF of a χ^2 distribution with $d - M$ degrees-of-freedom. The $\Omega(d)$ lower bound follows directly by noting that $X \sim \chi^2(d - M)$ implies

$$\frac{X - (d - M)}{\sqrt{2(d - M)}} \xrightarrow{d} \mathcal{N}(0, 1) \quad d \to \infty. \tag{43}$$

and vice versa, as the appropriate coreset objective function in each case differs substantially. For example, while subsampling and importance weighting remains a popular and useful method in the optimization context, they are known to provide poor Bayesian posterior approximations.

6. DISTRIBUTED BAYESIAN INFERENCE

Distributed methods for Bayesian inference address the challenges posed by massive data using a divide-and-conquer technique. We reiterate the three main groups of distributed methods introduced in the motivation. The first class of methods is the simplest and has three steps: divide the data into disjoint subsets and store them across multiple machines, run a Monte Carlo algorithm in parallel on all the machines, and combine parameter draws from all the subsets on a central machine. The last step requires one round of communication, so these approaches belong to the class of *one-shot learning* methods [213, 123, 132, 185, 214, 178, 123, 134, 177, 66, 38, 222, 84, 220, 122, 67, 68, 119, 40]. They differ mainly in their combination schemes and are based on a key insight that the parameters drawn on the subsets provide a noisy approximation of the true posterior distribution.

The second class of methods relies on distributed extensions of stochastic gradient MCMC [4, 94, 32, 48], which are typically based on stochastic gradient Langevin dynamics (SGLD) [217, 108]. They also split the data into subsets but have several rounds of communication among the machines. In every iteration, they select a subset with a certain probability, draw the parameter using a SGLD-type update, and communicate the parameter draw to the central machine. The high variance of the stochastic gradients and high communication costs in distributed SGLD extensions have motivated the development of the third set of methods [20, 157]. They are stochastic extensions of global consensus methods for distributed optimization, such as Alternating Direction Method of Multipliers (ADMM) [17, 150]. They divide the data into subsets, store them on machines, and augment the posterior density with auxiliary variables, which are conditionally independent given the parameter and observed data. Under certain limiting assumptions, the distribution of the parameter given the observed data converges to the target. The conditional independence assumption is crucial for drawing the auxiliary variables in parallel, whereas the limiting condition ensures asymptotic accuracy. Every iteration consists of synchronous updates where the machines storing the data draw the auxiliary variables and send them to the central machine that uses them to draw the parameter [200, 167, 201, 157, 202].

Distributed Bayesian methods have three main advantages. First, most are algorithm-agnostic and are easily used with any Monte Carlo algorithm. Second, distributed methods come with asymptotic guarantees about their accuracy. Such results show that approximated and target posterior distributions are asymptotically equivalent under mild regularity assumptions. Finally, they are easily extended to handle application-specific constraints, such as clustering of samples in nonparametric models [136] and privacy-preserving federated learning [86].

We cover the basics of distributed Bayesian inference and recent advances in Section 6.1–Section 6.3, and discuss future research directions in Section 6.4.

6.1 One-Shot Learning[2]

We provide a brief overview of one-shot learning, for which a wide variety of methods are available in the literature. We start with the most common setup that assumes the observations are conditionally independent given the parameter, leading to a product form for the likelihood. Let $x^n = (x_1, \dots, x_n)$ denote the observed data. The model is specified using the distribution $\mathbb{P}_\theta$ with density $p(x|\theta)$ and p-dimensional parameter $\theta \in \Theta \subseteq \mathbb{R}^p$. Assume that x^n is randomly partitioned into K disjoint subsets. Let $x_{(j)} = \{x_{(j)1}, \dots, x_{(j)M}\}$ be the jth subset ($j = 1, \dots, K$), where we have assumed that all the subset sample sizes equal M and $n = KM$ for simplicity. The true and subset j likelihoods are $\ell_n(\theta) = \prod_{i=1}^n p(x_i|\theta)$ and $\ell_{jM}(\theta) = \prod_{i=1}^M p(x_{(j)i}|\theta)$. Let Π be a prior distribution on Θ with density $\pi(\theta)$. Then the posterior density of θ given x^n is $\pi_n(\theta|x^n) = \ell_n(\theta)\pi(\theta)/C_n$, where $C_n = \int_\Theta \ell_n(\theta)\pi(\theta)\,d\theta$ and C_n is finite.

6.1.1 *Consensus Monte Carlo.* Consensus Monte Carlo (CMC) and its generalizations leave the likelihood on the subsets unchanged and down-weight the prior. They exploit the observation that the full data posterior can be factored as a product of subset posteriors with tempered priors [178]:

$$\begin{aligned}\pi_n(\theta|x^n) &= C_n^{-1} \prod_{j=1}^K \{\pi(\theta)\}^{1/K} \ell_{jM}(\theta) \\ &\propto \prod_{j=1}^K \pi_M(\theta|x_{(j)}) \equiv \prod_{j=1}^K \pi_j(\theta).\end{aligned} \tag{44}$$

Here, $\pi_M(\theta|x_{(j)})$ (or $\pi_j(\theta)$) is the jth subset posterior density of θ computed using likelihood and prior $\ell_{jM}(\theta)$ and $\{\pi(\theta)\}^{1/K}$. Let $\theta_{(j)t}$ be the parameter draws obtained from $\pi_j(\theta)$ using a Monte Carlo algorithm ($j = 1, \dots, K$; $t = 1, \dots, T$) and $\hat{\pi}_j(\theta)$ be an estimate of $\pi_j(\theta)$ obtained using $\theta_{(j)t}$s. Then $\prod_{j=1}^K \hat{\pi}_j(\theta)$ is proportional to an estimate of $\pi_n(\theta|x^n)$. If $\pi_j(\theta)$s are Gaussian, then so is $\pi_n(\theta|x^n)$, implying that the weighted averages of $\theta_{(j)t}$s correspond to draws from $\pi_n(\theta|x^n)$ [178]. More accurate

[2] The literature refers to one-shot learning methods as *divide-and-conquer*, *embarrassingly parallel* or *single communication* methods

combination algorithms estimate $\pi_j(\theta)$ using kernel density estimation [132], Weierstrass transform [213], random partition trees [214], Gaussian process regression [134] and normalizing flows [119], where the last two approaches also use importance sampling to select promising $\theta_{(j)t}$s for better approximation accuracy.

6.1.2 *Subset posterior distributions.* The methods based on subset posterior distributions up-weight the subset likelihoods but leave the prior unchanged. They draw parameters from the subset posteriors using any Monte Carlo algorithm and obtain their empirical approximations. The distributed posterior distribution, which approximates the true posterior, combines the empirical approximations of subset posteriors via a geometric center or mixture.

Median and mean posterior distributions. These methods combine the subset posterior distributions using their geometric center, such as the median and mean posterior distributions. The main difference between them and CMC-type approaches is the definition of subset posterior densities. Specifically, the jth subset posterior density is

$$\pi_M(\theta|x_{(j)}) = C_M^{-1}\{\ell_{jM}(\theta)\}^K \pi(\theta) \equiv \tilde{\pi}_j(\theta), \tag{45}$$

where $C_M = \int_\Theta \{\ell_{jM}(\theta)\}^K \pi(\theta)\, d\theta$ is assumed to be finite for posterior propriety. The pseudo-likelihood $\{\ell_{jM}(\theta)\}^K$ in equation (45) is the likelihood of a pseudo sample resulting from replicating every sample in the jth subset K times [123]. This pseudo-likelihood ensures the posterior variance of the subset and true posterior densities are calibrated up to $o_P(n^{-1})$ terms [99, 123, 185]. Similar to the CMC-type methods, $\theta_{(j)t}$s are drawn in parallel from $\tilde{\pi}_j(\theta)$s using any Monte Carlo algorithm. Let $\tilde{\Pi}_j$ be the jth subset posterior distribution with density $\tilde{\pi}_j(\theta)$. Then its empirical approximation supported on the $\theta_{(j)t}$s is $\hat{\Pi}_j = T^{-1}\sum_{t=1}^T \delta_{\theta_{(j)t}}(\cdot)$, where $\delta_\theta(\cdot)$ is the delta measure supported on θ. The median and mean posterior distributions are approximated using empirical measures $\hat{\Pi}^*$ and $\hat{\overline{\Pi}}$ that are supported on $\theta_{(j)t}$s. The weights of $\theta_{(j)t}$s are estimated via optimization such that $\sum_{j=1}^K \mathsf{d}(\hat{\Pi}^*, \hat{\Pi}_j)$ and $\sum_{j=1}^K \mathsf{d}^2(\hat{\overline{\Pi}}, \hat{\Pi}_j)$ are minimized, respectively, where d is a metric on probability measures [123, 185]. In certain cases, the form of the mean posterior is tractable [99, 182]; for example, if θ is one-dimensional and d is the 2-Wasserstein distance, then the αth quantile of the mean posterior equals the average of αth quantiles of the K subset posteriors.

Mixture of recentered subset posteriors. The final combination algorithm uses a K-component mixture of recentered subset posterior densities in equation (45). Let $\overline{\theta}_{(j)}$ be the mean of $\pi_M(\theta|x_{(j)})$ and $\overline{\theta} = \sum_{j=1}^K \overline{\theta}_{(j)}/K$. Then the distributed posterior distribution with density

$$\tilde{\pi}(\theta|x^n) = \sum_{j=1}^K \frac{1}{K}\tilde{\pi}_j(\theta - \overline{\theta} + \overline{\theta}_j), \tag{46}$$

approximates $\pi_n(\theta|x^n)$, where $\tilde{\pi}_j$ is defined in equation (45) [222, 220]. To generate draws from $\tilde{\pi}(\theta|x^n)$ in equation (46), we obtain the empirical approximation of the distributed posterior $\tilde{\Pi}$ with density $\tilde{\pi}(\theta|x^n)$ as

$$\hat{\tilde{\Pi}} = \sum_{j=1}^K \sum_{l=1}^T \frac{1}{KT}\delta_{\hat{\theta}+\theta_{(j)l}-\hat{\theta}_j}(\cdot), \tag{47}$$

where $\hat{\theta}_j = \sum_{l=1}^T \theta_{(j)l}/T$ and $\hat{\theta} = \sum_{j=1}^K \hat{\theta}_j/K$. The K-mixture $\hat{\tilde{\Pi}}$ and geometric centers $\hat{\Pi}^*$, $\hat{\overline{\Pi}}$ are similar in that the atoms of the empirical measures are transformations of the subset posterior draws. The main difference between them lies in their approach to estimating the weights of the atoms. All the atoms of $\hat{\tilde{\Pi}}$ have equal weights (i.e., $(KT)^{-1}$), whereas the atom weights of $\hat{\Pi}^*$ and $\hat{\overline{\Pi}}$ are nonuniform and estimated via optimization algorithms.

Asymptotics. The large sample properties of the posterior estimated in one-shot learning, denoted as $\Pi_{\mathrm{D},n}$, are justified via a BvM theorem; however, these results are only known for the methods based on subset posterior distributions [99, 123]. A BvM for $\Pi_{\mathrm{D},n}$ shows that it is asymptotically normal under mild assumptions as K and n tend to infinity. The center of the limiting distribution is specific to the combination algorithm, but the asymptotic covariance matrix equals I_0^{-1}/n for all of them, where I_0 is the Fisher information matrix computed using $Y \sim \mathbb{P}_{\theta_0}$. This shows that the asymptotic covariance of the true and distributed posteriors are calibrated up to $o_P(n^{-1})$ terms. Under these assumptions and for a bounded Θ,

$$\|\Pi_{\mathrm{D},n}(\cdot|x^n) - \Pi_n(\cdot|x^n)\|_{\mathrm{TV}} \lesssim \|\tilde{\theta} - \hat{\theta}\|_2, \tag{48}$$

where $\|\cdot\|_{\mathrm{TV}}$ is the total variation distance, $\hat{\theta}$ is the maximum likelihood estimate (MLE) of θ computed using x^n, and $\tilde{\theta}$ is a center of the K subset MLEs of θ: $\hat{\theta}_1, \dots, \hat{\theta}_K$. They satisfy $\|\hat{\theta}_j - \theta_0\|_2 = O_P(M^{-1/2})$, so $\|\tilde{\theta} - \theta_0\|_2 = O_P(M^{-1/2})$ because $\tilde{\theta}$ is a center of the subset MLEs. Furthermore, $\|\hat{\theta} - \theta_0\|_2 = O_P(n^{-1/2})$ and combining it with the previous result and cancellation of leading order terms imply that $\|\tilde{\theta} - \hat{\theta}\|_2 = o_P(M^{-1/2})$, which does not scale in K [99]. This shows that the bias of $\Pi_{\mathrm{D},n}$ in approximating Π_n does not decrease as K increases and K does not generally impact the approximation accuracy of $\Pi_{\mathrm{D},n}$, unless $\tilde{\theta}$ is a root-n consistent estimator of θ_0.

Notable recent advances. Methods based on subset posteriors have been generalized to dependent data. In time series data, smaller blocks of consecutive observations form the subsets to preserve the ordering of samples. A measure of dependence, such as the mixing coefficient, dictates the choice of K. The subset pseudo-likelihood in equation (45) is modified to condition on the immediately preceding time block to model the dependence and is raised to a power of K. For one-shot learning in hidden

Markov models with mixing coefficient ρ, the distributed posterior estimated using equation (47) with the modified pseudo-likelihood and $K = o(\rho^{-M})$ satisfies equation (48) [207]. These results have been generalized to a broader class of models, but guidance on the choice of K remains underexplored [146].

Posterior computations in Gaussian process (GP) regression fail to scale even for moderately large n [164, 12]. To address this challenge, various methods based on variational inference and mixtures-of-experts have been proposed in the past, but none exploit distributed computations [195, 163, 64, 39]. One-shot learning has addressed this challenge without strong theoretical guarantees [123, 185, 230]. The choice of K here depends on the smoothness of the regression function. Assuming a higher order of smoothness of regression functions guarantees accurate estimation on the subsets for larger values of K. Specifically, if the regression function is infinitely smooth, the predictor lies in [0, 1], and $K = O(n/\log^2 n)$, then the decay rates of estimation risks for the distributed and true posterior distributions depend only on n and are asymptotically equivalent. In more general problems where the regression function belongs to the Hölder class of functions on $[0, 1]^D$ with smoothness index α, the upper bound for K depends on n, D and α for guaranteeing optimal decay rate of the estimation risk [180, 67]. These results have been generalized to varying coefficients models [68].

Adapting to the unknown smoothness of the regression function (i.e., α) is a related and more difficult challenge in distributed GP regression. The smoothness index of the regression function is unknown in practice, so the goal is to construct distributed procedures that estimate the smoothness using automatic data-driven tuning. In a signal-in-white noise model and the previously mentioned models, the optimal guarantees of one-shot learning methods based on a GP prior depend on the unknown smoothness index, which is a major limitation in practice [186]. If the subsets communicate with a central machine under constraints, then adaptation in this model is possible under the frequentist setting [232, 187]. The main idea of these methods is to estimate different parts of the regression function using groups of data subsets, followed by a merging phase at the central machine that adjusts for the true regression function's unknown smoothness. Similar results are unavailable in distributed high-dimensional Bayesian estimation.

Limitations. The main limitation of one-shot learning methods is their reliance on the normality of the subset posterior distributions. Scaling of the parameter draws on the subsets helps in some cases but fails to generalize beyond the family of elliptical posterior distributions [182, 203]. De Souza et al. [40] identify three additional problems for one-shot learning. First, subset posteriors fail to capture the support of a multimodal posterior with high probability. Second, a subset posterior can be substantially biased and fail to be a reasonable approximation of the true posterior. Finally, subset posterior draws may fail to provide information about the tails of the true posterior, resulting in poor estimates of tail event probabilities and overconfident inference. A key observation of [40] is that communication among machines may be necessary for improving the approximation accuracy of subset posteriors; see [232, 187] for similar observations in distributed nonparametric regression.

6.2 Stochastic Gradient MCMC

Stochastic gradient MCMC (SGMCMC) methods are widely used for scalable Bayesian inference. The most popular variant is based on the Langevin diffusion process

$$\text{(49)} \qquad d\theta(t) = \frac{1}{2}\nabla \log \pi_n\big(\theta(t)|x^n\big)\,dt + dB_t,$$

where B_t is a p-dimensional Brownian motion. Under mild regularity assumptions, $\Pi_n(\cdot|x^n)$ is the stationary distribution of $\theta(t)$ [172, 156]. In practice, one typically uses a discrete-time Euler approximation of equation (49),

$$\text{(50)} \qquad \begin{aligned} &\theta_{t+1} = \theta_t + \frac{h}{2} g_n(\theta_t) + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, hI), \\ &g_n(\theta) = \frac{\partial}{\partial\theta}\log \pi(\theta) + \sum_{i=1}^{n} \frac{\partial}{\partial\theta}\log p(x_i|\theta), \end{aligned}$$

where θ_{t+1} is the proposed θ value at the $(t+1)$th iteration and the approximation accuracy increases as h decreases. To correct for the approximation error in equation (50), θ_{t+1} is accepted with the probability in equation (3) to guarantee that $\Pi_n(\cdot|x^n)$ is the stationary distribution of the $\{\theta_t\}$ chain.

SGMCMC algorithms based on equation (50) use $g_n(\theta)$, the gradient of $\log \pi_n(\theta|x^n)$, for generating θ proposals in an MH-type algorithm [170, 131]; however, computation of $g_n(\theta)$ requires cycling through all the samples, which is prohibitively slow for a large n. SGLD, based on stochastic gradient descent, bypasses this problem by sub-sampling a size m subset S_m of $\{1, \dots, n\}$ and proposing θ in the $(t+1)$th iteration using a noisy approximation of $g_n(\theta)$ as

$$\text{(51)} \qquad \begin{aligned} &\theta_{t+1} = \theta_t + \frac{h_t}{2} \hat{g}_n(\theta_t) + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, h_t I), \\ &\hat{g}_n(\theta) = \frac{\partial}{\partial\theta}\log \pi(\theta) + \frac{n}{m}\sum_{i\in S_m} \frac{\partial}{\partial\theta}\log p(x_i|\theta), \end{aligned}$$

where $\hat{g}_n(\theta)$ is a noisy estimate of $g_n(\theta)$ and the step size h_t decreases to 0 such that $\sum_{t=1}^{\infty} h_t = \infty$ and $\sum_{t=1}^{\infty} h_t^2 < \infty$ [217]. The factor (n/m) in equation (51) is similar to the up-weighting of the subset likelihood by a power a

K in equation (45). In practice, however, $h_t \propto 1/n$ for efficiency [20]. This produces a chain $\{\theta_t\}$ that does not have the target as the stationary distribution, but it mimics the true continuous-time Langevin dynamics closely, and hence has "approximately" the right target. SGMCMC has been generalized by replacing the Langevin dynamics with other processes, resulting in stochastic gradient Hamiltonian Monte Carlo (SGHMC) [108]; see [133] for a detailed overview of different types of SGMCMC algorithms and their theoretical properties.

Notable recent advances. The simplest SGMCMC extension replaces the uniform weights n/m in $\hat{g}_n(\theta)$ with nonuniform weights w_i for every i such that $\sum_{i \in S_m} w_i = n$ for any choice of S_m. The nonuniformly weighted gradient estimator can have a lower variance than $\hat{g}_n(\theta)$ in some applications [179]. Similar to this idea, subsampling-based MCMC algorithms leverage survey sampling techniques to construct unbiased estimates of the log-likelihood. The unbiased estimate replaces the log-likelihood in a sampling algorithm or yields an efficient approximation of the acceptance probability in equation (3); however, the application of these algorithms is currently limited to simple models such as logistic regression [13, 161].

Distributed extensions of SGLD are useful in applications where the data are stored on K machines, and moving the data to a central location for subsampling is infeasible. The Distributed SGLD (DSGLD) algorithm has been developed for such applications [4, 101]; however, DSGLD has two major limitations [48]. First, the variance of the gradient estimate is significantly larger than in SGLD. Second, the subsets require extensive communication to ensure convergence, resulting in computational bottlenecks. This problem has been addressed using better gradient surrogates with smaller variances [32, 48].

Limitations. The performance of SGMCMC algorithms is sensitive to the choice of step size, yet there is no standardized method for its selection. A common practical heuristic is to initially choose the largest step size that avoids divergence. By incrementally decreasing the step size from this starting point to successively smaller values, one can approach the step size that optimizes the algorithm's performance.

Relatively few SGMCMC algorithms exist for inference in constrained spaces and dependent data models. For constrained spaces, the popular choice is to project the Langevin dynamics into the constrained space, but this can result in slow convergence [20, 21]. The other alternative is to model the manifold structure of parameters, but this suffers from asymptotic bias on the boundary of the parameter space [153]. Similar to one-shot learning methods, the primary challenge in developing SGMCMC extensions for dependent data models lies in obtaining reliable gradient estimates using subsets of data. This has been achieved in hidden Markov and nonlinear state-space models via nonoverlapping sequential partitions of the data and under the assumption that the subsets are nearly independent [109, 5]; however, the accuracy is guaranteed only when subset sizes are sufficiently large, which offsets the computational gains from subsampling.

6.3 Asymptotically Exact Data Augmentation

Asymptotically exact data augmentation (AXDA) generalizes data augmentation (DA) using stochastic extensions of global consensus optimization algorithms such as ADMM [167, 200, 201]. AXDA has subset-specific auxiliary variables $z = (z_1, \dots, z_K) \in \prod_{k=1}^{K} \mathbb{R}^M$ and a tolerance parameter $\rho \in \mathbb{R}_+$, which are analogous to "missing data" in DA and the tolerance parameter in ADMM, respectively. An ADMM algorithm exploits the decomposability of the objective to efficiently find the optimum value through a sequence of conditional minimizations. The advantages of ADMM are that it is readily parallelized and has superior convergence properties; see [17] for details. AXDA exploits conditional independence of missing data given observed data and parameters. Using the notation in equation (44), z is chosen such that the augmented density satisfies

$$(52) \quad \pi_\rho(\theta, z_1, \dots, z_K | x^n) \propto \pi(\theta) \prod_{k=1}^{K} \ell_{k,\rho}(\theta, z_k),$$

where $\ell_{k,\rho}(\theta, z_k) = p_k(z_k, x_{(k)}) \kappa_\rho(z_k, \theta)$, κ_ρ is a kernel such that $\kappa_\rho(\cdot, \theta)$ converges weakly to $\delta_\theta(\cdot)$ as $\rho \to 0$ and $p_k(z_k, x_{(k)})$ is such that $\lim_{\rho \to 0} \int \ell_{k,\rho}(\theta, z_k)\, dz_k = \ell_{kM}(\theta) = \prod_{i=1}^{M} p(x_{(k)i} | \theta)$; that is, z_k and $(z_k, x_{(k)})$ are missing and complete data on subset k, and z preserves the observed data model as $\rho \to 0$, justifying that AXDA is asymptotically exact.

The advantage of the density in equation (52) is that the z_ks are conditionally independent given θ. In every iteration, z_ks are drawn in parallel on the machines storing $x_{(k)}$s. These draws are communicated to the central machine that uses them to draw θ and generates a Markov chain for θ, whose stationary density equals $\pi_n(\theta | x^n)$ under mild assumptions. AXDA has been used for Bayesian inference in generalized linear models and nonparametric regression [167, 201], but proper choices of $p_k(z_k, x_{(k)})$, ρ and κ_ρ limit the broader application of AXDA. Vono, Paulin and Doucet [202] and [157] develop AXDA using ADMM-type variable splitting and Monte Carlo algorithms based on Langevin dynamics. Like DSGLD, repeated communications among the machines diminish the computation gains from distributed sampling on the subsets.

6.4 Open Questions and Future Directions

This section highlights the additional limitations of distributed inference methodology, important open problems, and areas for future investigation.

High-dimensional and dependent data models. A variety of options exist for distributed Bayesian inference in independent data models, but they fail to generalize to high-dimensional models. The literature on distributed methods for inference in high-dimensional models is sparsely populated [84]. The development of distributed methods that exploit the low-dimensional structure in high-dimensional problems is desired.

Most distributed methods assume that the likelihood has a product form; see equation (44). This assumption fails for many time series and spatial models. There are one-shot learning methods for hidden Markov models [207], but they are inapplicable beyond the family of elliptical posterior distributions. No dependent data extensions are available for DSGLD and AXDA algorithms.

Bias and variance reduction. The bias between the true and distributed posterior in one-shot learning fails to decay as K increases. For parametric models, equation (48) shows that the distributed posterior distribution has a bias of the order $o_P(M^{-1/2})$, which is suboptimal compared to $o_P(n^{-1/2})$ order bias of the true posterior. This means that increasing K has no impact on the accuracy of the distributed posterior. One way to bypass this problem is by centering the distributed posterior at a root-n consistent estimator; see [207]. Addressing this problem is useful for Bayesian federated learning, where one-shot learning is increasingly used due to its simplicity [86]. Similarly, developing gradient surrogates with smaller variances is crucial for Bayesian federated learning using SGMCMC algorithms.

Asynchronous updates. Synchronous updates are crucial for convergence guarantees of DSGLD and Monte Carlo algorithms based on AXDA; however, they become expensive with increasing number of subsets, resulting in diminishing benefits of distributed computations. Asynchronous updates bypass such problems when the subset sizes are similar, but they imply that the $\{\theta_t\}$ chain is not Markov, which rules out conventional tools for proving convergence guarantees. Recently, asynchronous DA has been developed for high-dimensional variable selection and mixed effects models, demonstrating the benefits of multiple rounds of communication among the subsets under a fixed communication budget [231]. The extensions of this scheme to a broader class of models, including asynchronous DSGLD and AXDA, remain unknown.

Generalized likelihoods. Bayesian inference using generalized likelihoods has several advantages, including robustness and targeted inference; however, the current literature on distributed inference relies heavily on exploiting the structure of a traditional Bayesian hierarchical model. Preliminary results are available on the commonalities between AXDA and approximate Bayesian inference [201]. For broader applications, it is interesting to explore distributed extensions of the *cut* posterior in misspecified models [158] and distributed inference in Bayesian models based on generalized likelihoods.

Applications. Distributed Bayesian inference has found applications in federated learning [86]. These methods are ideal for Bayesian analysis of multicenter longitudinal clinical studies because the data cannot be moved to a central location due to privacy concerns. Similar privacy concerns arise in industry applications in which moving user data around incurs a security risk. It is of interest to develop privacy-preserving extensions of distributed methods targeted to these applications.

Comparing to competitors. Different approaches to distributed Bayesian inference have developed more or less independently. While there are comprehensive overviews of specific methods, such as SGMCMC [133] and expectation propagation [198], thorough comparisons between these methods are lacking. Similarly, incorporating active learning into CMC applications has been shown to enhance CMC's accuracy [40], yet the advantages of CMC over other one-shot learning alternatives remain unclear. Furthermore, empirical and theoretical analyses comparing SGMCMC with one-shot learning techniques are absent from the literature. For instance, determining the conditions under which DSGLD outperforms one-shot learning in high-dimensional settings is particularly valuable.

Bayesian coresets employ subsampling to efficiently approximate posterior densities with a surrogate. The asymptotic results discussed in Section 6.1 suggest that the surrogate posterior may be biased owing to the reduced sample size, yet a comprehensive analysis of this issue has not been conducted. In contrast, the rate of convergence results for analogous frequentist methods that utilize subsampling are well established in a variety of settings [210, 209, 208, 226]; therefore, a promising avenue for research is to derive similar optimality results for the surrogate posterior obtained using Bayesian coresets.

Automated diagnostics and accessibility. Automated application and model diagnostics for distributed methods have received little attention. One-shot learning methods are often easily implemented using the parallel R package [190]; however, a similar general-purpose software for deploying distributed algorithms in practice remains to be developed. Addressing these challenges is crucial in facilitating the wide applicability of distributed methods.

7. DISCUSSION

Tools for Bayesian computation are evolving at a rapid pace, thanks largely to recent developments in machine learning. We highlighted this phenomenon with four vignettes. The first vignette discussed sampling with the aid of generative models, particularly normalizing flows. Normalizing flows define a new flexible family of proposal distributions with efficient, automatic proposal tuning as a key advantage over traditional methods that require manual parameter selection (e.g., selecting temperature schedules in parallel tempering). If sampling succeeds, then evaluating posterior expectations is trivial. Unfortunately, there are many posteriors for which standard sampling algorithms either fail to mix or are too computationally expensive. We discussed two approaches to reduce computational burden for large data: coresets and federated learning. Coresets take a variational approach to data compression, with recent methods leveraging deep neural networks to build flexible surrogate families; federated Bayesian learning methods instead distribute posterior computation over many computers. Federated learning methods may be faster if large-scale distributed resources are available. The remaining vignette covered variational inference, which replaces the posterior with a tractable approximation and is especially useful when sampling is infeasible or computational resources are limited.

Many more vignettes could be written on similar topics, such as accelerating sampling with diffusion-based generative models or accelerating approximate Bayesian computation using deep neural networks for data compression. We close with three themes, applicable to all vignettes, that we believe should receive future attention: accelerating inference using previous calculations, improving accessibility with new software and providing theoretical support for empirically promising algorithms.

The status-quo in Bayesian computation is to start from scratch in each posterior inference problem, such as recomputing coresets after changing the prior, or estimating a new variational approximation when applying an old model to new data. This is inefficient, as posterior inference in similar models must be somewhat informative about posterior inference in the current model. If the two models under consideration are directly comparable, such as posteriors under slightly different priors, then it may be easy to leverage previous calculations, for example, by using warm starts in optimization routines. Problems arise when the two models have different dimensions, such a hierarchical models with an extra layer of parameters. We are hopeful that methods for similar problems in machine learning—particularly transfer learning—will play a role in developing general solutions for Bayesians.

Another common theme was the need for improved automation and accessibility. Implementing methods involving neural networks or other machine learning techniques in a robust and reliable fashion is a nontrivial task, often requiring significant time and expert knowledge. Given the breakneck speed at which machine learning progresses, careful implementations can be outdated before they have a chance for widespread adoption. The focus should be on developing software, which is modular enough to withstand the next machine learning revolution, as well as user-friendly enough to be applied en-masse.

Finally, statisticians should be cautious with wholesale adoption of methods that achieve excellent practical performance at the expense of theoretical guarantees. Fast “approximations” to posterior distributions that can be arbitrarily far from the exact posterior may be useful for black box prediction but fall far short of what is needed for reliable and reproducible Bayesian inferences. This is particularly key in scientific and policy applications in which one needs to appropriately characterize uncertainty in learning from data, acknowledging complexities that arise in practice such as model uncertainty, data contamination etc. Guarantees are necessary to avoid highly misleading inferences and potentially catastrophic conclusions from the types of large and complex data sets that are being generated routinely in the sciences.

ACKNOWLEDGMENTS

T. Campbell, L. Lin and S. Srivastava contributed equally.

FUNDING

This research was partially supported by the Lifeplan project funded by the European Research Council under the European Union’s Horizon 2020 research and innovation programme (Grant Agreement No. 856506). Additional funding came from the United States Office of Naval Research (N000142112510) and National Institutes of Health (R01-ES028804, R01-ES035625).

REFERENCES

[1] AGARWAL, P. K., HAR-PELED, S. and VARADARAJAN, K. R. (2005). Geometric approximation via coresets. In *Combinatorial and Computational Geometry*. *Math. Sci. Res. Inst. Publ.* **52** 1–30. Cambridge Univ. Press, Cambridge. MR2178310 https://doi.org/10.4171/PRIMS/172

[2] AGRAWAL, R. (1995). The continuum-armed bandit problem. *SIAM J. Control Optim.* **33** 1926–1951. MR1358102 https://doi.org/10.1137/S0363012992237273

[3] AHN, S., CHEN, Y. and WELLING, M. (2013). Distributed and adaptive darting Monte Carlo through regenerations. In *Artificial Intelligence and Statistics* 108–116. PMLR.

[4] AHN, S., SHAHBABA, B. and WELLING, M. (2014). Distributed stochastic gradient MCMC. *Int. Conf. Mach. Learn.* **32** 1044–1052.

[5] AICHER, C., PUTCHA, S., NEMETH, C., FEARNHEAD, P. and FOX, E. B. (2023). Stochastic gradient MCMC for nonlinear state space models. *Bayesian Anal.* 1–23.

[6] ALQUIER, P. and RIDGWAY, J. (2020). Concentration of tempered posteriors and of their variational approximations. *Ann. Statist.* **48** 1475–1497. MR4124331 https://doi.org/10.1214/19-AOS1855

[7] Anderson, W. (2012). *Continuous-Time Markov Chains: An Applications Oriented Approach*. Springer, Berlin.

[8] Andricioaei, I., Straub, J. E. and Voter, A. F. (2001). Smart darting Monte Carlo. *J. Chem. Phys.* **114** 6994–7000.

[9] Andrieu, C. and Thoms, J. (2008). A tutorial on adaptive MCMC. *Stat. Comput.* **18** 343–373. MR2461882 https://doi.org/10.1007/s11222-008-9110-y

[10] Arbel, M., Matthews, A. and Doucet, A. (2021). Annealed flow transport Monte Carlo. In *Int. Conf. Mach. Learn.* **38** 318–330. PMLR.

[11] Bachem, O., Lucic, M. and Krause, A. (2017). Practical coreset constructions for machine learning. Available at arXiv:1703.06476.

[12] Banerjee, S., Carlin, B. P. and Gelfand, A. E. (2014). *Hierarchical Modeling and Analysis for Spatial Data*. CRC Press, Boca Raton, FL.

[13] Bardenet, R., Doucet, A. and Holmes, C. (2017). On Markov chain Monte Carlo methods for tall data. *J. Mach. Learn. Res.* **18** 47. MR3670492

[14] Betancourt, M. (2015). The fundamental incompatibility of Hamiltonian Monte Carlo and data subsampling. *Int. Conf. Mach. Learn.* **37** 533–540.

[15] Blei, D., Griffiths, T., Jordan, M. and Tenenbaum, J. (2003). Hierarchical topic models and the nested Chinese restaurant process. *Adv. Neural Inf. Process. Syst.* **16**.

[16] Blei, D. M., Kucukelbir, A. and McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *J. Amer. Statist. Assoc.* **112** 859–877. MR3671776 https://doi.org/10.1080/01621459.2017.1285773

[17] Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J. et al. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*.

[18] Broderick, T., Boyd, N., Wibisono, A., Wilson, A. and Jordan, M. (2013). Streaming variational Bayes. *Adv. Neural Inf. Process. Syst.* **26**.

[19] Brofos, J., Gabrié, M., Brubaker, M. A. and Lederman, R. R. (2022). Adaptation of the independent Metropolis-Hastings sampler with normalizing flow proposals. *Int. Conf. Artif. Intell. Stat.* **151** 5949–5986.

[20] Brosse, N., Durmus, A. and Moulines, E. (2018). The promises and pitfalls of stochastic gradient Langevin dynamics. *Adv. Neural Inf. Process. Syst.* **31**.

[21] Bubeck, S., Eldan, R. and Lehec, J. (2018). Sampling from a log-concave distribution with projected Langevin Monte Carlo. *Discrete Comput. Geom.* **59** 757–783. MR3802303 https://doi.org/10.1007/s00454-018-9992-1

[22] Campbell, T. and Beronov, B. (2019). Sparse variational inference: Bayesian coresets from scratch. *Adv. Neural Inf. Process. Syst.* **32**.

[23] Campbell, T. and Broderick, T. (2018). Bayesian coreset construction via greedy iterative geodesic ascent. *Int. Conf. Mach. Learn.* **80** 698–706.

[24] Campbell, T. and Broderick, T. (2019). Automated scalable Bayesian inference via Hilbert coresets. *J. Mach. Learn. Res.* **20** 15. MR3911422

[25] Camuto, A. and Willetts, M. (2022). Variational autoencoders: A harmonic perspective. In *Proceedings of the* 25*th International Conference on Artificial Intelligence and Statistics* **151** 4595–4611. PMLR.

[26] Camuto, A., Willetts, M., Roberts, S., Holmes, C. and Rainforth, T. (2021). Towards a theoretical understanding of the robustness of variational autoencoders. In *Proceedings of the* 24*th International Conference on Artificial Intelligence and Statistics* **24** 3565–3573. PMLR.

[27] Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P. et al. (2017). Stan: A probabilistic programming language. *J. Stat. Softw.* **76**.

[28] Casella, G. and Robert, C. P. (1996). Rao-Blackwellisation of sampling schemes. *Biometrika* **83** 81–94. MR1399157 https://doi.org/10.1093/biomet/83.1.81

[29] Chae, M., Kim, D., Kim, Y. and Lin, L. (2023). A likelihood approach to nonparametric estimation of a singular distribution using deep generative models. *J. Mach. Learn. Res.* **24** 77. MR4582499

[30] Chan, R., Pollock, M., Johansen, A. and Roberts, G. (2021). Divide-and-conquer Monte Carlo fusion. Available at arXiv:2110.07265.

[31] Chatterjee, S. and Diaconis, P. (2018). The sample size required in importance sampling. *Ann. Appl. Probab.* **28** 1099–1135. MR3784496 https://doi.org/10.1214/17-AAP1326

[32] Chen, C., Ding, N., Li, C., Zhang, Y. and Carin, L. (2016). Stochastic gradient MCMC with stale gradients. *Adv. Neural Inf. Process. Syst.* **29**.

[33] Chen, N. and Campbell, T. (2023). Coreset Markov chain Monte Carlo. Available at arXiv:2310.17063.

[34] Chen, N., Xu, Z. and Campbell, T. (2022). Bayesian inference via sparse Hamiltonian flows. *Adv. Neural Inf. Process. Syst.* **35**.

[35] Chen, R. T., Rubanova, Y., Bettencourt, J. and Duvenaud, D. K. (2018). Neural ordinary differential equations. *Adv. Neural Inf. Process. Syst.* **31**.

[36] Chen, X., Wen, H. and Li, Y. (2021). Differentiable particle filters through conditional normalizing flow. In 2021 *IEEE* 24*th International Conference on Information Fusion* **24** 1–6.

[37] Chérief-Abdellatif, B.-E. and Alquier, P. (2018). Consistency of variational Bayes inference for estimation and model selection in mixtures. *Electron. J. Stat.* **12** 2995–3035. MR3855643 https://doi.org/10.1214/18-EJS1475

[38] Dai, H., Pollock, M. and Roberts, G. (2019). Monte Carlo fusion. *J. Appl. Probab.* **56** 174–191. MR3981152 https://doi.org/10.1017/jpr.2019.12

[39] Deisenroth, M. and Ng, J. W. (2015). Distributed Gaussian processes. In *Int. Conf. Mach. Learn.* **37** 1481–1490. PMLR.

[40] De Souza, D. A., Mesquita, D., Kaski, S. and Acerbi, L. (2022). Parallel MCMC without embarrassing failures. *Int. Conf. Artif. Intell. Stat.* **151** 1786–1804.

[41] Diaconis, P. (1988). Sufficiency as statistical symmetry. *Proc. AMS Centen. Symp.* 15–26.

[42] Dieng, A., Tran, D., Ranganath, R., Paisley, J. and Blei, D. (2017). Variational inference via χ-upper bound minimization. *Adv. Neural Inf. Process. Syst.* **30**.

[43] Dinh, L., Sohl-Dickstein, J. and Bengio, S. (2016). Density estimation using real NVP. ArXiv preprint. Available at arXiv:1605.08803.

[44] Dinh, V., Bilge, A., Zhang, C. and Matsen IV, F. A. (2017). Probabilistic path Hamiltonian Monte Carlo. *Int. Conf. Mach. Learn.* **70** 1009–1018.

[45] Domke, J., Garrigos, G. and Gower, R. (2019). Provable convergence guarantees for black-box variational inference. *Adv. Neural Inf. Process. Syst.* **32**.

[46] Dunson, D. B. and Johndrow, J. E. (2020). The Hastings algorithm at fifty. *Biometrika* **107** 1–23. MR4064137 https://doi.org/10.1093/biomet/asz066

[47] Durkan, C., Bekasov, A., Murray, I. and Papamakarios, G. (2019). Neural spline flows. *Adv. Neural Inf. Process. Syst.* **32**.

[48] EL MEKKAOUI, K., MESQUITA, D., BLOMSTEDT, P. and KASKI, S. (2021). Federated stochastic gradient Langevin dynamics. *Uncertainty Artif. Intell.* **161** 1703–1712.

[49] FELDMAN, D. (2020). Introduction to Core-sets: An updated survey. Available at arXiv:2011.09384.

[50] FELDMAN, D., FIAT, A., KAPLAN, H. and NISSIM, K. (2009). Private coresets. In *STOC'*09*—Proceedings of the* 2009 *ACM International Symposium on Theory of Computing* 361–370. ACM, New York. MR2780082

[51] FELDMAN, D. and LANGBERG, M. (2011). A unified framework for approximating and clustering data. In *STOC'*11*—Proceedings of the* 43*rd ACM Symposium on Theory of Computing* 569–578. ACM, New York. MR2932007 https://doi.org/10.1145/1993636.1993712

[52] FOSTER, D. J., SEKHARI, A. and SRIDHARAN, K. (2018). Uniform convergence of gradients for non-convex learning and optimization. *Adv. Neural Inf. Process. Syst.* **32**.

[53] GABRIÉ, M., ROTSKOFF, G. M. and VANDEN-EIJNDEN, E. (2022). Adaptive Monte Carlo augmented with normalizing flows. *Proc. Natl. Acad. Sci. USA* **119** e2109420119. MR4417577 https://doi.org/10.1073/pnas.2109420119

[54] GAO, C., ISAACSON, J. and KRAUSE, C. (2020). I-flow: High-dimensional integration and sampling with normalizing flows. *Mach. Learn.: Sci. Technol.* **1**.

[55] GE, H., XU, K. and GHAHRAMANI, Z. (2018). Turing: A language for flexible probabilistic inference. *Artif. Intell. Stat.* **84** 1682–1690.

[56] GEFFNER, T. and DOMKE, J. (2021). MCMC variational inference via uncorrected Hamiltonian annealing. *Adv. Neural Inf. Process. Syst.* **34**.

[57] GELMAN, A. and RUBIN, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statist. Sci.* **7** 457–472.

[58] GHOSAL, S., GHOSH, J. K. and VAN DER VAART, A. W. (2000). Convergence rates of posterior distributions. *Ann. Statist.* **28** 500–531. MR1790007 https://doi.org/10.1214/aos/1016218228

[59] GHOSAL, S., LEMBER, J. and VAN DER VAART, A. (2008). Nonparametric Bayesian model selection and averaging. *Electron. J. Stat.* **2** 63–89. MR2386086 https://doi.org/10.1214/07-EJS090

[60] GHOSH, S., DELLE FAVE, F. and YEDIDIA, J. (2016). Assumed density filtering methods for learning Bayesian neural networks. *Proc. AAAI Conf. Artif. Intell.* **30**.

[61] GIORDANO, R., BRODERICK, T. and JORDAN, M. I. (2018). Covariances, robustness, and variational Bayes. *J. Mach. Learn. Res.* **19** 51. MR3874159

[62] GONG, W., LI, Y. and HERNÁNDEZ-LOBATO, J. M. (2019). Meta-learning for stochastic gradient MCMC.

[63] GOODFELLOW, I., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A. and BENGIO, Y. (2020). Generative adversarial networks. *Commun. ACM* **63**.

[64] GRAMACY, R. B. and APLEY, D. W. (2015). Local Gaussian process approximation for large computer experiments. *J. Comput. Graph. Statist.* **24** 561–578. MR3357395 https://doi.org/10.1080/10618600.2014.914442

[65] GREGOR, K., DANIHELKA, I., GRAVES, A., REZENDE, D. and WIERSTRA, D. (2015). DRAW: A recurrent neural network for image generation. In *Proceedings of the* 32*nd Int. Conf. Mach. Learn.* **37** 1462–1471. PMLR.

[66] GUHANIYOGI, R. and BANERJEE, S. (2018). Meta-Kriging: Scalable Bayesian modeling and inference for massive spatial datasets. *Technometrics* **60** 430–444. MR3878099 https://doi.org/10.1080/00401706.2018.1437474

[67] GUHANIYOGI, R., LI, C., SAVITSKY, T. and SRIVASTAVA, S. (2023). Distributed Bayesian inference in massive spatial data. *Statist. Sci.* **38** 262–284. MR4597336 https://doi.org/10.1214/22-sts868

[68] GUHANIYOGI, R., LI, C., SAVITSKY, T. D. and SRIVASTAVA, S. (2022). Distributed Bayesian varying coefficient modeling using a Gaussian process prior. *J. Mach. Learn. Res.* **23** 84. MR4576669

[69] GUI, J., SUN, Z., WEN, Y., TAO, D. and YE, J. (2021). A review on generative adversarial networks: Algorithms, theory, and applications. *IEEE Trans. Knowl. Data Eng.* **35** 3313–3332.

[70] GULRAJANI, I., AHMED, F., ARJOVSKY, M., DUMOULIN, V. and COURVILLE, A. C. (2017). Improved training of Wasserstein GANs. *Adv. Neural Inf. Process. Syst.* **30**.

[71] GUO, F., WANG, X., FAN, K., BRODERICK, T. and DUNSON, D. (2016). Boosting variational inference **29**.

[72] HAARIO, H., SAKSMAN, E. and TAMMINEN, J. (2001). An adaptive Metropolis algorithm. *Bernoulli* **7** 223–242. MR1828504 https://doi.org/10.2307/3318737

[73] HALL, P., PHAM, T., WAND, M. P. and WANG, S. S. J. (2011). Asymptotic normality and valid inference for Gaussian variational approximation. *Ann. Statist.* **39** 2502–2532. MR2906876 https://doi.org/10.1214/11-AOS908

[74] HARSHVARDHAN, G. M., GOURISARIA, M. K., PANDEY, M. and RAUTARAY, S. S. (2020). A comprehensive survey and analysis of generative models in machine learning. *Comput. Sci. Rev.* **38** 100285. MR4131875 https://doi.org/10.1016/j.cosrev.2020.100285

[75] HOFFMAN, M. D., BLEI, D. M., WANG, C. and PAISLEY, J. (2013). Stochastic variational inference. *J. Mach. Learn. Res.* **14** 1303–1347. MR3081926

[76] HOFFMAN, M. D. and GELMAN, A. (2014). The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *J. Mach. Learn. Res.* **15** 1593–1623. MR3214779

[77] HOLLAND, P. W., LASKEY, K. B. and LEINHARDT, S. (1983). Stochastic blockmodels: First steps. *Soc. Netw.* **5** 109–137. MR0718088 https://doi.org/10.1016/0378-8733(83)90021-7

[78] HUGGINS, J., CAMPBELL, T. and BRODERICK, T. (2016). Coresets for scalable Bayesian logistic regression. *Adv. Neural Inf. Process. Syst.* **29**.

[79] JANKOWIAK, M. and PHAN, D. (2022). Surrogate likelihoods for variational annealed importance sampling. *Int. Conf. Mach. Learn.* **162** 9881–9901.

[80] JEONG, K., CHAE, M. and KIM, Y. (2023). Online learning for the Dirichlet process mixture model via weakly conjugate approximation. *Comput. Statist. Data Anal.* **179** 107626. MR4495773 https://doi.org/10.1016/j.csda.2022.107626

[81] JOHNDROW, J., PILLAI, N. and SMITH, A. (2020). No free lunch for approximate MCMC. Available at arXiv:2010.12514.

[82] JOLICOEUR-MARTINEAU, A. (2019). The relativistic discriminator: A key element missing from standard GAN.

[83] JORDAN, M., GHAHRAMANI, Z., JAAKKOLA, T. and SAUL, L. (1999). An introduction to variational methods for graphical models. *Mach. Learn.* **37** 183–233.

[84] JORDAN, M. I., LEE, J. D. and YANG, Y. (2019). Communication-efficient distributed statistical inference. *J. Amer. Statist. Assoc.* **114** 668–681. MR3963171 https://doi.org/10.1080/01621459.2018.1429274

[85] KATSEVICH, A. and RIGOLLET, P. (2023). On the approximation accuracy of Gaussian variational Inference.

[86] KIDD, B., WANG, K., XU, Y. and NI, Y. (2022). Federated learning for sparse Bayesian models with applications to electronic health records and genomics. *Pac. Symp. BioComputing* **28** 484–495.

[87] KIM, K., WU, K., OH, J., MA, Y. and GARDNER, J. (2023). On the convergence of black-box variational inference. *Adv. Neural Inf. Process. Syst.* **37**.

[88] KIM, Y., CHAE, M., JEONG, K., KANG, B. and CHUNG, H. (2016). An online Gibbs sampler algorithm for hierarchical Dirichlet processes prior. *Mach. Learn. Knowl. Discov. Databases* 509–523.

[89] KINGMA, D. P. and BA, J. (2017). Adam: A method for stochastic optimization.

[90] KINGMA, D. P., SALIMANS, T., JOZEFOWICZ, R., CHEN, X., SUTSKEVER, I. and WELLING, M. (2016). Improved variational inference with inverse autoregressive flow. *Adv. Neural Inf. Process. Syst.* **30**.

[91] KINGMA, D. P. and WELLING, M. (2014). Auto-encoding variational Bayes. In 2*nd International Conference on Learning Representations*, *Banff*, *AB*, *Canada*, *April* 14-16, 2014, *Conference Track Proceedings*.

[92] KINGMA, D. P. and WELLING, M. (2019). An introduction to variational autoencoders. *Found. Trends Mach. Learn.* **12** 307–392.

[93] KOBYZEV, I., PRINCE, S. J. and BRUBAKER, M. A. (2021). Normalizing flows: An introduction and review of current methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **43** 3964–3979.

[94] KORATTIKARA, A., CHEN, Y. and WELLING, M. (2014). Austerity in MCMC land: Cutting the Metropolis-Hastings budget. *Int. Conf. Mach. Learn.* **32** 181–189.

[95] LAN, S., STREETS, J. and SHAHBABA, B. (2014). Wormhole Hamiltonian Monte Carlo. *Proc. AAAI Conf. Artif. Intell.* 1953–1959.

[96] LAURITZEN, S. L. (1988). *Extremal Families and Systems of Sufficient Statistics*. *Lecture Notes in Statistics* **49**. Springer, New York. MR0971253 https://doi.org/10.1007/978-1-4612-1023-8

[97] LE, T. A., BAYDIN, A. G. and WOOD, F. (2017). Inference compilation and universal probabilistic programming. *Artif. Intell. Stat.* **54** 1338–1348.

[98] LEVY, D., HOFFMAN, M. D. and SOHL-DICKSTEIN, J. (2018). Generalizing Hamiltonian Monte Carlo with neural networks. *Int. Conf. Learn. Represent.*

[99] LI, C., SRIVASTAVA, S. and DUNSON, D. B. (2017). Simple, scalable and accurate posterior interval estimation. *Biometrika* **104** 665–680. MR3694589 https://doi.org/10.1093/biomet/asx033

[100] LI, J., LUO, X. and QIAO, M. (2020). On generalization error bounds of noisy gradient methods for non-convex learning.

[101] LI, W., AHN, S. and WELLING, M. (2016). Scalable MCMC for mixed membership stochastic blockmodels. In *Artificial Intelligence and Statistics* **51** 723–731. PMLR.

[102] LI, Y. and TURNER, R. (2016). Rényi divergence variational inference. *Adv. Neural Inf. Process. Syst.* **30**.

[103] LIANG, F., MAHONEY, M. and HODGKINSON, L. (2022). Fat–tailed variational inference with anisotropic tail adaptive flows. *Int. Conf. Mach. Learn.* **162** 13257–13270.

[104] LIN, D. (2013). Online learning of nonparametric mixture models via sequential variational approximation. *Adv. Neural Inf. Process. Syst.* **26**.

[105] LOH, P.-L. and WAINWRIGHT, M. J. (2013). Regularized M-estimators with nonconvexity: Statistical and algorithmic theory for local optima. *Adv. Neural Inf. Process. Syst.* **26**.

[106] LOO, N., SWAROOP, S. and TURNER, R. E. (2021). Generalized variational continual learning. In *International Conference on Learning Representations*.

[107] LU, X., PERRONE, V., HASENCLEVER, L., TEH, Y. W. and VOLLMER, S. (2017). Relativistic Monte Carlo. *Artif. Intell. Stat.* **54** 1236–1245.

[108] MA, Y.-A., CHEN, T. and FOX, E. (2015). A complete recipe for stochastic gradient MCMC. *Adv. Neural Inf. Process. Syst.* **28**.

[109] MA, Y.-A., FOTI, N. J. and FOX, E. B. (2017). Stochastic gradient MCMC methods for hidden Markov models. In *International Conference on Machine Learning* 2265–2274. PMLR.

[110] MACLAURIN, D. and ADAMS, R. (2014). Firefly Monte Carlo: Exact MCMC with subsets of data. *Conf. Uncertain. Artif. Intell.* **30** 543–552.

[111] MAHONEY, M. (2011). Randomized algorithms for matrices and data. *Found. Trends Mach. Learn.* **3** 123–224.

[112] MANGOUBI, O., PILLAI, N. S. and SMITH, A. (2018). Does Hamiltonian Monte Carlo mix faster than a random walk on multimodal densities? ArXiv preprint. Available at arXiv:1808.03230.

[113] MANOUSAKAS, D., RITTER, H. and KARALETSOS, T. (2022). Black-box coreset variational inference. *Adv. Neural Inf. Process. Syst.* **36**.

[114] MANOUSAKAS, D., XU, Z., MASCOLO, C. and CAMPBELL, T. (2020). Bayesian pseudocoresets. *Adv. Neural Inf. Process. Syst.* **33**.

[115] MAO, X., LI, Q., XIE, H., LAU, R. Y., WANG, Z. and PAUL SMOLLEY, S. (2017). Least-squares generative adversarial networks. *Proceedings of the IEEE International Conference on Computer Vision*.

[116] MATHIEU, E. and NICKEL, M. (2020). Riemannian continuous normalizing flows. *Adv. Neural Inf. Process. Syst.* **33**.

[117] MATTHEWS, A., ARBEL, M., REZENDE, D. J. and DOUCET, A. (2022). Continual repeated annealed flow transport Monte Carlo. In *Int. Conf. Mach. Learn.* **162** 15196–15219. PMLR.

[118] MEI, S., BAI, Y. and MONTANARI, A. (2018). The landscape of empirical risk for nonconvex losses. *Ann. Statist.* **46** 2747–2774. MR3851754 https://doi.org/10.1214/17-AOS1637

[119] MESQUITA, D., BLOMSTEDT, P. and KASKI, S. (2020). Embarrassingly parallel MCMC using deep invertible transformations. *Uncertain. Artif. Intell.* **115** 1244–1252.

[120] MILLER, A. C., FOTI, N. J. and ADAMS, R. P. (2017). Variational boosting: Iteratively refining posterior approximations. *Proc.* 34*th Int. Conf. Mach. Learn.* **70** 2420–2429.

[121] MINKA, T. and LAFFERTY, J. (2002). Expectation-propagation for the generative aspect model. *Proc. Eighteen. Conf. Uncertain. Artif. Intell.* **18** 352–359.

[122] MINSKER, S. (2019). Distributed statistical estimation and rates of convergence in normal approximation. *Electron. J. Stat.* **13** 5213–5252. MR4043072 https://doi.org/10.1214/19-EJS1647

[123] MINSKER, S., SRIVASTAVA, S., LIN, L. and DUNSON, D. (2017). Scalable and robust Bayesian inference via the median posterior. *Int. Conf. Mach. Learn.* **32** 1656–1664.

[124] MIRZA, M. and OSINDERO, S. (2014). Conditional generative adversarial nets. ArXiv preprint. Available at arXiv:1411.1784.

[125] MIYATO, T., KATAOKA, T., KOYAMA, M. and YOSHIDA, Y. (2018). Spectral normalization for generative adversarial networks.

[126] MOHASEL AFSHAR, H. and DOMKE, J. (2015). Reflection, refraction, and Hamiltonian Monte Carlo. *Adv. Neural Inf. Process. Syst.* **28**.

[127] MÜLLER, T., MCWILLIAMS, B., ROUSSELLE, F., GROSS, M. and NOVÁK, J. (2019). Neural importance sampling. *ACM Trans. Graph.* **38** 1–19.

[128] NAGAPETYAN, T., DUNCAN, A., HASENCLEVER, L., VOLLMER, S., SZPRUCH, L. and ZYGALAKIS, K. (2017). The true cost of stochastic gradient Langevin dynamics. Available at arXiv:1706.02692.

[129] NAIK, C., ROUSSEAU, J. and CAMPBELL, T. (2022). Fast Bayesian coresets via subsampling and quasi-Newton refinement. *Adv. Neural Inf. Process. Syst.* **35**.

[130] NEAL, R. M. (2001). Annealed importance sampling. *Stat. Comput.* **11** 125–139. MR1837132 https://doi.org/10.1023/A:1008923215028

[131] NEAL, R. M. (2011). MCMC using Hamiltonian dynamics. In *Handbook of Markov Chain Monte Carlo. Chapman & Hall/CRC Handb. Mod. Stat. Methods* 113–162. CRC Press, Boca Raton, FL. MR2858447

[132] NEISWANGER, W., WANG, C. and XING, E. P. (2014). Asymptotically exact, embarrassingly parallel MCMC. *Proc. Thirtieth Conf. Uncertain. Artif. Intell.*

[133] NEMETH, C. and FEARNHEAD, P. (2021). Stochastic gradient Markov chain Monte Carlo. *J. Amer. Statist. Assoc.* **116** 433–450. MR4227705 https://doi.org/10.1080/01621459.2020.1847120

[134] NEMETH, C. and SHERLOCK, C. (2018). Merging MCMC subposteriors through Gaussian-process approximations. *Bayesian Anal.* **13** 507–530. MR3780433 https://doi.org/10.1214/17-BA1063

[135] NGUYEN, C. V., LI, Y., BUI, T. D. and TURNER, R. E. (2018). Variational continual learning. *Int. Conf. Learn. Represent.*.

[136] NI, Y., JI, Y. and MÜLLER, P. (2020). Consensus Monte Carlo for random subsets using shared anchors. *J. Comput. Graph. Statist.* **29** 703–714. MR4191237 https://doi.org/10.1080/10618600.2020.1737085

[137] NIEMAN, D., SZABO, B. and VAN ZANTEN, H. (2023). Uncertainty quantification for sparse spectral variational approximations in Gaussian process regression. *Electron. J. Stat.* **17** 2250–2288. MR4649981 https://doi.org/10.1214/23-ejs2155

[138] NIKOLAKAKIS, K. E., HADDADPOUR, F., KARBASI, A. and KALOGERIAS, D. S. (2022). Beyond Lipschitz: Sharp generalization and excess risk bounds for full-batch GD. ArXiv preprint. Available at arXiv:2204.12446.

[139] NING, B. (2021). Spike and slab Bayesian sparse principal component analysis. ArXiv preprint. Available at arXiv:2102.00305.

[140] NISHIMURA, A. and DUNSON, D. (2016). Geometrically tempered Hamiltonian Monte Carlo. ArXiv preprint. Available at arXiv:1604.00872.

[141] NISHIMURA, A., DUNSON, D. and LU, J. (2017). Discontinuous Hamiltonian Monte Carlo for sampling discrete parameters. ArXiv preprint. Available at arXiv:1705.08510.

[142] OHN, I. and LIN, L. (2021). Adaptive variational Bayes: Optimality, computation and applications. ArXiv preprint. Available at arXiv:2109.03204.

[143] ONEILL, J. (2020). An overview of neural network compression. Available at arXiv:2006.03669.

[144] ORBANZ, P. (2017). Subsampling large graphs and invariance in networks. Available at arXiv:1710.04217.

[145] ORBANZ, P. and ROY, D. (2015). Bayesian models of graphs, arrays, and other exchangeable structures. *IEEE Trans. Pattern Anal. Mach. Intell.* **37** 437–461.

[146] OU, R., SEN, D. and DUNSON, D. (2021). Scalable Bayesian inference for time series via divide-and-conquer. ArXiv preprint. Available at arXiv:2106.11043.

[147] PAISLEY, J., BLEI, D. M. and JORDAN, M. I. (2012). Variational Bayesian inference with stochastic search. In *Proceedings of the* 29*th International Coference on Int. Conf. Mach. Learn.* 1363–1370. Omnipress.

[148] PAKMAN, A. and PANINSKI, L. (2013). Auxiliary-variable exact Hamiltonian Monte Carlo samplers for binary distributions. *Adv. Neural Inf. Process. Syst.* **26**.

[149] PAPAMAKARIOS, G., NALISNICK, E., REZENDE, D. J., MOHAMED, S. and LAKSHMINARAYANAN, B. (2021). Normalizing flows for probabilistic modeling and inference. *J. Mach. Learn. Res.* **22** 57. MR4253750

[150] PARIKH, N., BOYD, S. et al. (2014). Proximal algorithms. *Found. Trends Optim.* **1**.

[151] PASARICA, C. and GELMAN, A. (2010). Adaptively scaling the Metropolis algorithm using expected squared jumped distance. *Statist. Sinica* **20** 343–364. MR2640698

[152] PATI, D., BHATTACHARYA, A. and YANG, Y. (2018). On statistical optimality of variational Bayes. *Proc.* 21*st Int. Conf. Artif. Intell. Stat.* **84** 1579–1588.

[153] PATTERSON, S. and TEH, Y. W. (2013). Stochastic gradient Riemannian Langevin dynamics on the probability simplex. *Adv. Neural Inf. Process. Syst.* **26**.

[154] PETZKA, H., FISCHER, A. and LUKOVNICOV, D. (2017). On the regularization of Wasserstein GANs. ArXiv preprint. Available at arXiv:1709.08894.

[155] PHILLIPS, J. M. and TAI, W. M. (2020). Near-optimal coresets of kernel density estimates. *Discrete Comput. Geom.* **63** 867–887. MR4110524 https://doi.org/10.1007/s00454-019-00134-6

[156] PILLAI, N. S., STUART, A. M. and THIÉRY, A. H. (2012). Optimal scaling and diffusion limits for the Langevin algorithm in high dimensions. *Ann. Appl. Probab.* **22** 2320–2356. MR3024970 https://doi.org/10.1214/11-AAP828

[157] PLASSIER, V., VONO, M., DURMUS, A. and MOULINES, E. (2021). DG-LMC: A turn-key and scalable synchronous distributed MCMC algorithm via Langevin Monte Carlo within Gibbs. *Int. Conf. Mach. Learn.* **139** 8577–8587.

[158] PLUMMER, M. (2015). Cuts in Bayesian graphical models. *Stat. Comput.* **25** 37–43. MR3304902 https://doi.org/10.1007/s11222-014-9503-z

[159] POMPE, E., HOLMES, C. and ŁATUSZYŃSKI, K. (2020). A framework for adaptive MCMC targeting multimodal distributions. *Ann. Statist.* **48** 2930–2952. MR4152629 https://doi.org/10.1214/19-AOS1916

[160] QUIROZ, M., KOHN, R., DANG, K.-D., VILLANI, M. and TRAN, M.-N. (2018). Subsampling MCMC—an introduction for the survey statistician. *Sankhya A* **80** S33–S69. MR3968357 https://doi.org/10.1007/s13171-018-0153-7

[161] QUIROZ, M., KOHN, R., VILLANI, M. and TRAN, M.-N. (2019). Speeding up MCMC by efficient data subsampling. *J. Amer. Statist. Assoc.* **114** 831–843. MR3963184 https://doi.org/10.1080/01621459.2018.1448827

[162] RANGANATH, R., GERRISH, S. and BLEI, D. (2014). Black box variational inference. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics* **33** 814–822. PMLR.

[163] RASMUSSEN, C. and GHAHRAMANI, Z. (2001). Infinite mixtures of Gaussian process experts. *Adv. Neural Inf. Process. Syst.* **14**.

[164] RASMUSSEN, C. E. and WILLIAMS, C. K. I. (2006). *Gaussian Processes for Machine Learning. Adaptive Computation and Machine Learning*. MIT Press, Cambridge, MA. MR2514435

[165] RAY, K. and SZABÓ, B. (2022). Variational Bayes for high-dimensional linear regression with sparse priors. *J. Amer. Statist. Assoc.* **117** 1270–1281. MR4480711 https://doi.org/10.1080/01621459.2020.1847121

[166] RAY, K., SZABÓ, B. and CLARA, G. (2020). Spike and slab variational Bayes for high dimensional logistic regression. *Proc.* 34*th Int. Conf. Neural Inf. Proc. Syst.* **34**.

[167] RENDELL, L. J., JOHANSEN, A. M., LEE, A. and WHITELEY, N. (2021). Global consensus Monte Carlo. *J. Comput. Graph. Statist.* **30** 249–259. MR4270501 https://doi.org/10.1080/10618600.2020.1811105

[168] REZENDE, D. and MOHAMED, S. (2015). Variational inference with normalizing flows. *Int. Conf. Mach. Learn.*.

[169] REZENDE, D. J., PAPAMAKARIOS, G., RACANIERE, S., ALBERGO, M., KANWAR, G., SHANAHAN, P. and CRANMER, K. (2020). Normalizing flows on tori and spheres. *Int. Conf. Mach. Learn.* **119** 8083–8092.

[170] ROBERTS, G. O. and ROSENTHAL, J. S. (1998). Optimal scaling of discrete approximations to Langevin diffusions. *J. R. Stat. Soc. Ser. B. Stat. Methodol.* **60** 255–268. MR1625691 https://doi.org/10.1111/1467-9868.00123

[171] ROBERTS, G. O. and ROSENTHAL, J. S. (2009). Examples of adaptive MCMC. *J. Comput. Graph. Statist.* **18** 349–367. MR2749836 https://doi.org/10.1198/jcgs.2009.06134

[172] ROBERTS, G. O. and TWEEDIE, R. L. (1996). Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli* **2** 341–363. MR1440273 https://doi.org/10.2307/3318418

[173] ROSS, S. M. (2002). *Simulation*. Academic Press, Amsterdam.

[174] ROSSKY, P. J., DOLL, J. D. and FRIEDMAN, H. L. (1978). Brownian dynamics as smart Monte Carlo simulation. *J. Chem. Phys.* **69** 4628–4633.

[175] ROTH, K., LUCCHI, A., NOWOZIN, S. and HOFMANN, T. (2017). Stabilizing training of generative adversarial networks through regularization. *Adv. Neural Inf. Process. Syst.* **31**.

[176] SALIMANS, T., KINGMA, D. and WELLING, M. (2015). Markov chain Monte Carlo and variational inference: Bridging the gap. *Int. Conf. Mach. Learn.* **37** 1218–1226.

[177] SAVITSKY, T. D. and SRIVASTAVA, S. (2018). Scalable Bayes under informative sampling. *Scand. J. Stat.* **45** 534–556. MR3858945 https://doi.org/10.1111/sjos.12312

[178] SCOTT, S. L., BLOCKER, A. W., BONASSI, F. V., CHIPMAN, H. A., GEORGE, E. I. and MCCULLOCH, R. E. (2016). Bayes and big data: The consensus Monte Carlo algorithm. *Int. J. Manag. Sci. Eng. Manag.* **11** 78–88.

[179] SEN, D., SACHS, M., LU, J. and DUNSON, D. B. (2020). Efficient posterior sampling for high-dimensional imbalanced logistic regression. *Biometrika* **107** 1005–1012. MR4186502 https://doi.org/10.1093/biomet/asaa035

[180] SHANG, Z., HAO, B. and CHENG, G. (2019). Nonparametric Bayesian aggregation for massive data. *J. Mach. Learn. Res.* **20** 140. MR4030154

[181] SHUN, Z. and MCCULLAGH, P. (2018). Laplace approximation of high dimensional integrals. *J. Roy. Statist. Soc. Ser. B* **57** 749–760.

[182] SHYAMALKUMAR, N. D. and SRIVASTAVA, S. (2022). An algorithm for distributed Bayesian inference. *Stat* **11** e432. MR4449228 https://doi.org/10.1002/sta4.432

[183] SNELSON, E. and GHAHRAMANI, Z. (2005). Sparse Gaussian processes using pseudo-inputs. *Adv. Neural Inf. Process. Syst.* **18**.

[184] SONG, J., ZHAO, S. and ERMON, S. (2017). A-NICE-MC: Adversarial training for MCMC. *Adv. Neural Inf. Process. Syst.* **30**.

[185] SRIVASTAVA, S., CEVHER, V., DINH, Q. and DUNSON, D. (2015). WASP: Scalable Bayes via barycenters of subset posteriors. *Artif. Intell. Stat.* **38** 912–920.

[186] SZABÓ, B. and VAN ZANTEN, H. (2019). An asymptotic analysis of distributed nonparametric methods. *J. Mach. Learn. Res.* **20** 87. MR3960941

[187] SZABÓ, B. and VAN ZANTEN, H. (2022). Distributed function estimation: Adaptation using minimal communication. *Math. Stat. Learn.* **5** 159–199. MR4526299 https://doi.org/10.4171/msl/33

[188] TANG, R. and YANG, Y. (2021). On empirical Bayes variational autoencoder: An excess risk bound. In *Proceedings of Thirty Fourth Conference on Learning Theory* **134** 4068–4125. PMLR.

[189] TANG, R. and YANG, Y. (2023). Minimax rate of distribution estimation on unknown submanifolds under adversarial losses. *Ann. Statist.* **51** 1282–1308. MR4630949 https://doi.org/10.1214/23-aos2291

[190] R CORE TEAM (2021). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.

[191] THIN, A., KOTELEVSKII, N., DOUCET, A., DURMUS, A., MOULINES, E. and PANOV, M. (2021). Monte Carlo variational auto-encoders. *Int. Conf. Mach. Learn.* **139** 10247–10257.

[192] TOKDAR, S. T. and KASS, R. E. (2010). Importance sampling: A review. *Wiley Interdiscip. Rev.: Comput. Stat.* **2**.

[193] TRAN, D., RANGANATH, R. and BLEI, D. (2017). Hierarchical implicit models and likelihood-free variational inference. *Adv. Neural Inf. Process. Syst.* **31**.

[194] TRAN, D., VAFA, K., AGRAWAL, K., DINH, L. and POOLE, B. (2019). Discrete flows: Invertible generative models of discrete data. *Adv. Neural Inf. Process. Syst.* **32**.

[195] TRESP, V. (2000). A Bayesian committee machine. *Neural Comput.* **12** 2719–2741. https://doi.org/10.1162/089976600300014908

[196] UEHARA, M., SATO, I., SUZUKI, M., NAKAYAMA, K. and MATSUO, Y. (2016). Generative adversarial nets from a density ratio estimation perspective. ArXiv preprint. Available at arXiv:1610.02920.

[197] VAKILI, S., SCARLETT, J., SHAN SHIU, D. and BERNACCHIA, A. (2022). Improved convergence rates for sparse approximation methods in kernel-based learning. In *Int. Conf. Mach. Learn.* **162** 21960–21983. PMLR.

[198] VEHTARI, A., GELMAN, A., SIVULA, T., JYLÄNKI, P., TRAN, D., SAHAI, S., BLOMSTEDT, P., CUNNINGHAM, J. P., SCHIMINOVICH, D. et al. (2020). Expectation propagation as a way of life: A framework for Bayesian inference on partitioned data. *J. Mach. Learn. Res.* **21** 17. MR4071200

[199] VIHOLA, M. (2012). Robust adaptive Metropolis algorithm with coerced acceptance rate. *Stat. Comput.* **22** 997–1008. MR2950080 https://doi.org/10.1007/s11222-011-9269-5

[200] VONO, M., DOBIGEON, N. and CHAINAIS, P. (2019). Split-and-augmented Gibbs sampler—application to large-scale inference problems. *IEEE Trans. Signal Process.* **67** 1648–1661. MR3938771 https://doi.org/10.1109/TSP.2019.2894825

[201] VONO, M., DOBIGEON, N. and CHAINAIS, P. (2021). Asymptotically exact data augmentation: Models, properties, and algorithms. *J. Comput. Graph. Statist.* **30** 335–348. MR4270508 https://doi.org/10.1080/10618600.2020.1826954

[202] VONO, M., PAULIN, D. and DOUCET, A. (2022). Efficient MCMC sampling with dimension-free convergence rate using ADMM-type splitting. *J. Mach. Learn. Res.* **23** 25. MR4420750

[203] VYNER, C., NEMETH, C. and SHERLOCK, C. (2023). SwISS: A scalable Markov chain Monte Carlo divide-and-conquer strategy. *Stat* **12** e523. MR4538659 https://doi.org/10.1002/sta4.523

[204] WAINWRIGHT, M. and JORDAN, M. (2008). Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.* **1** 1–305.

[205] WALKER, J., DOERSCH, C., GUPTA, A. K. and HEBERT, M. (2016). An uncertain future: Forecasting from static images using variational autoencoders. In *European Conference on Computer Vision* **9911** 835–851.

[206] WAN, N., LI, D. and HOVAKIMYAN, N. (2020). f-divergence variational inference. In *Advances in Neural Information Processing Systems* **33**.

[207] WANG, C. and SRIVASTAVA, S. (2023). Divide-and-conquer Bayesian inference in hidden Markov models. *Electron. J. Stat.* **17** 895–947. MR4563529 https://doi.org/10.1214/23-ejs2118

[208] WANG, H. and MA, Y. (2021). Optimal subsampling for quantile regression in big data. *Biometrika* **108** 99–112. MR4226192 https://doi.org/10.1093/biomet/asaa043

[209] WANG, H., YANG, M. and STUFKEN, J. (2019). Information-based optimal subdata selection for big data linear regression. *J. Amer. Statist. Assoc.* **114** 393–405. MR3941263 https://doi.org/10.1080/01621459.2017.1408468

[210] WANG, H., ZHU, R. and MA, P. (2018). Optimal subsampling for large sample logistic regression. *J. Amer. Statist. Assoc.* **113** 829–844. MR3832230 https://doi.org/10.1080/01621459.2017.1292914

[211] WANG, T., ZHU, J.-Y., TORRALBA, A. and EFROS, A. (2018). Dataset distillation. Available at arXiv:1811.10959.

[212] WANG, W., SUN, Y. and HALGAMUGE, S. (2018). Improving MMD-GAN training with repulsive loss function. ArXiv preprint. Available at arXiv:1812.09916.

[213] WANG, X. and DUNSON, D. B. (2013). Parallelizing MCMC via Weierstrass sampler. ArXiv preprint. Available at arXiv:1312.4605.

[214] WANG, X., GUO, F., HELLER, K. A. and DUNSON, D. B. (2015). Parallelizing MCMC with random partition trees. *Adv. Neural Inf. Process. Syst.* **28**.

[215] WANG, Y., AUDIBERT, J.-Y. and MUNOS, R. (2008). Algorithms for infinitely many-armed bandits. *Adv. Neural Inf. Process. Syst.* **21**.

[216] WANG, Y. and BLEI, D. M. (2019). Frequentist consistency of variational Bayes. *J. Amer. Statist. Assoc.* **114** 1147–1161. MR4011769 https://doi.org/10.1080/01621459.2018.1473776

[217] WELLING, M. and TEH, Y. W. (2011). Bayesian learning via stochastic gradient Langevin dynamics. *Int. Conf. Mach. Learn.*.

[218] WILLIAMS, C. and SEEGER, M. (2001). Using the Nyström method to speed up kernel machines. *Adv. Neural Inf. Process. Syst.* **13**.

[219] WINN, J. and BISHOP, C. M. (2005). Variational message passing. *J. Mach. Learn. Res.* **6** 661–694. MR2249835

[220] WU, C. and ROBERT, C. P. (2017). Average of recentered parallel MCMC for big data. ArXiv preprint. Available at arXiv:1706.04780.

[221] XU, Z., CHEN, N. and CAMPBELL, T. (2023). MixFlows: Principled variational inference via mixed flows. In *Int. Conf. Mach. Learn.* **202** 38342–38376. PMLR.

[222] XUE, J. and LIANG, F. (2019). Double-parallel Monte Carlo for Bayesian analysis of big data. *Stat. Comput.* **29** 23–32. MR3905537 https://doi.org/10.1007/s11222-017-9791-1

[223] YANG, Y. and MARTIN, R. (2020). Variational approximations of empirical Bayes posteriors in high-dimensional linear models. ArXiv preprint. Available at arXiv:2007.15930.

[224] YANG, Y., PATI, D. and BHATTACHARYA, A. (2020). α-variational inference with statistical guarantees. *Ann. Statist.* **48** 886–905. MR4102680 https://doi.org/10.1214/19-AOS1827

[225] YIN, M. and ZHOU, M. (2018). Semi-implicit variational inference. In *Int. Conf. Mach. Learn.* **80** 5660–5669. PMLR.

[226] YU, J., WANG, H., AI, M. and ZHANG, H. (2022). Optimal distributed subsampling for maximum quasi-likelihood estimators with massive data. *J. Amer. Statist. Assoc.* **117** 265–276. MR4399084 https://doi.org/10.1080/01621459.2020.1773832

[227] ZHANG, F. and GAO, C. (2020). Convergence rates of variational posterior distributions. *Ann. Statist.* **48** 2180–2207. MR4134791 https://doi.org/10.1214/19-AOS1883

[228] ZHANG, G., HSU, K., LI, J., FINN, C. and GROSSE, R. (2021). Differentiable annealed importance sampling and the perils of gradient noise. *Adv. Neural Inf. Process. Syst.* **34**.

[229] ZHANG, J., KHANNA, R., KYRILLIDIS, A. and KOYEJO, O. (2021). Bayesian coresets: Revisiting the nonconvex optimization perspective. *Artif. Intell. Stat.* **130** 2782–2790.

[230] ZHANG, M. M. and WILLIAMSON, S. A. (2019). Embarrassingly parallel inference for Gaussian processes. *J. Mach. Learn. Res.* **20** 169. MR4048980

[231] ZHOU, J., KHARE, K. and SRIVASTAVA, S. (2023). Asynchronous and distributed data augmentation for massive data settings. *J. Comput. Graph. Statist.* **32** 895–907. MR4641467 https://doi.org/10.1080/10618600.2022.2130928

[232] ZHU, Y. and LAFFERTY, J. (2018). Distributed nonparametric regression under communication constraints. In *Int. Conf. Mach. Learn.* **35** 6009–6017. PMLR.

[233] ZIEGLER, Z. and RUSH, A. (2019). Latent normalizing flows for discrete sequences. *Int. Conf. Mach. Learn.* **97** 7673–7682.