AFRL-RI-RS-TR-2019-029

THE NEXT GENERATION OF PROBABILISTIC PROGRAMMING: MASSIVE DATA, DATA STREAMS, AND MODEL DIAGNOSTICS

PRINCETON UNIVERSITY

FEBRUARY 2019

FINAL TECHNICAL REPORT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

STINFO COPY

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE

■ **AIR FORCE MATERIEL COMMAND** ■ **UNITED STATES AIR FORCE** ■ **ROME, NY 13441**

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09. This report is available to the general public, including foreign nations. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RI-RS-TR-2019-029 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE CHIEF ENGINEER:

/ S /
STEVEN DRAGER
Work Unit Manager

/ S /
STEVEN JOHNS
Chief, Trusted Systems Branch
Computing & Communications Division
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
FEBRUARY 2019	FINAL TECHNICAL REPORT	JUL 2014 – AUG 2018

4. TITLE AND SUBTITLE

THE NEXT GENERATION OF PROBABILISTIC PROGRAMMING: MASSIVE DATA, DATA STREAMS, AND MODEL DIAGNOSTICS

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA8750-14-2-0009

5c. PROGRAM ELEMENT NUMBER
61101E

6. AUTHOR(S)

David M. Blei

5d. PROJECT NUMBER
PPML

5e. TASK NUMBER
3P

5f. WORK UNIT NUMBER
RI

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Trustees of Princeton University
The Office of Research and Project Administration
1 Nassau Hall
Princeton, NJ 08544-2001

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory/RITA
525 Brooks Road
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/RI

11. SPONSOR/MONITOR'S REPORT NUMBER
AFRL-RI-RS-TR-2019-029

12. DISTRIBUTION AVAILABILITY STATEMENT
Approved for Public Release; Distribution Unlimited. This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This effort made significant progress on inference for probabilistic programming. Probabilistic programming requires inference methods for approximating conditional distributions. Building on the framework of variational inference, this effort made this algorithm more efficient, more powerful, and more accurate. This effort developed new probabilistic models for economics, neuroscience, text analysis, population genetics, social network analysis, and recommendation systems. These methods were deployed in open-source software, on real-world programming systems and are currently in use by end-users of probabilistic programming. The work performed under this effort changed the landscape of approximate posterior inference, pushing forward the field of Bayesian machine learning and probabilistic programming.

15. SUBJECT TERMS
Probabilistic Programming, Machine Learning, Approximate Inference, Bayesian Nonparametric Models

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			**STEVEN DRAGER**
U	U	U	UU	39	19b. TELEPHONE NUMBER (*Include area code*) **N/A**

Table of Contents

List of Figures

1 Summary

The goal of probabilistic programming is to lubricate the probabilistic pipeline (Figure 1): to make it easier for real-world data analysts to design probability models, use them to analyze data, check the results, and revise the models.

The key algorithmic challenge is *posterior inference*, the algorithmic problem of squaring the model and the data to produce posterior estimates of the latent variables. More concretely, investigators need inference methods that are *general* and *scalable*. General inference are methods that apply to many models—a probabilistic programming system will outline a class of models that are expressible, and so we need inference methods that will work on that class. Scalable inference methods are ones that scale to the large data sizes that we now regularly encounter.

Through the years of our probabilistic programming and advanced machine learning (PPAML) project, we made significant progress on inference for probabilistic programming. We developed new scalable methods which enabled posterior inference with massive datasets. We developed new general methods, which greatly expanded the model class with which we can do automatic approximate inference. We deployed our methods on real-world probabilistic programming systems. Our algorithms are currently in use by the end-users of probabilistic programming. As the citations and implementations of our algorithms attest, the work that we did as part of this program changed the landscape of approximate posterior inference.

Our innovations in inference opened the door to new types of models. In parallel with innovating inference, we developed many new models. Some models were designed for specific applications, e.g., neuroscience, genetics, economics, language modeling, recommendation systems. Others are new classes of models, such as Bayesian nonparametric models that grow and change with the data or deep probabilistic models that learn layered representations of high dimensional data.

With new models and inference in hand, we also developed new ways to check and to strengthen data analysis with probabilistic machine learning. Note that the ability to revise a model is a luxury that comes with the general inference methods described above. Without general inference, changing the model requires too much labor to be feasible. In particular, we developed several methods for *robustness*, to build models that are robust to mismatches with the data. Further, we developed several new methods for checking models, to understand where and how the data violates the model in order to revise and improve it.

Put together, these accomplishments furthered the state of the art of every aspect of Figure 1.

2 Introduction

Analyzing, exploring, and predicting from data have become critical to science, industry, military, government, and society. Consider the following problems about data. (1) We have a social network of 250M people; we want to identify the communities in this network and summarize their demographic characteristics. (2) An unending stream of intelligence is monitored by a team of analysts. We want to intuitively organize the information in a navigator and deliver important information to the right people. (3) We have recorded the location and times of detonated explosives in a war-torn city. We want to predict where and when the next device will be detonated.

Experts solve these problems by following a data analysis pipeline. (a) Form assumptions about the data: How do different parts of the data relate to each other and what hidden structures might exist in the observations? (b) Analyze data (or multiple data sets) under the assumptions. (c) Use the analysis to form predictions, answer questions, make hypotheses, and explore the data.

Probabilistic modeling provides an elegant framework for executing this pipeline. It gives a formalism for describing assumptions about data, generic algorithms for analyzing data under those assumptions, and meaningful calculations for making predictions and exploring hidden structure. Building on this field, *probabilistic programming systems* (PPS) give expressive languages for specifying models and compilers to derive and implement the algorithms for using those models to analyze data. Such systems promise to let domain experts quickly develop and use sophisticated models without sophisticated machine learning expertise.

But probabilistic programming systems cannot yet fulfill their promise. (I) We need general-purpose algorithms that scale to **massive data** and we need to develop the theory and practice of applying probabilistic models to **data streams**. This is crucial for including probabilistic models in larger systems that continually collect, analyze, and act on data. (II) We need new methods to understand how well models work, methods for assessing **model fitness** and **model diagnostics**. As model building, fitting, and revising becomes a mainstream technological activity, assessing model fitness and diagnosing misfit must become equally mainstream.

My research lab spearheaded these developments. Our perspective is that building and using probabilistic models is part of an iterative process for solving data analysis problems. First, formulate a simple model based on the kinds of hidden structure that you believe exists in the data. Then, given a data set, use an inference algorithm to approximate the posterior—the conditional distribution of the hidden variables given the data—which points to the particular hidden pattens that your data exhibits. Finally, use the posterior to test the model against the data, identifying the

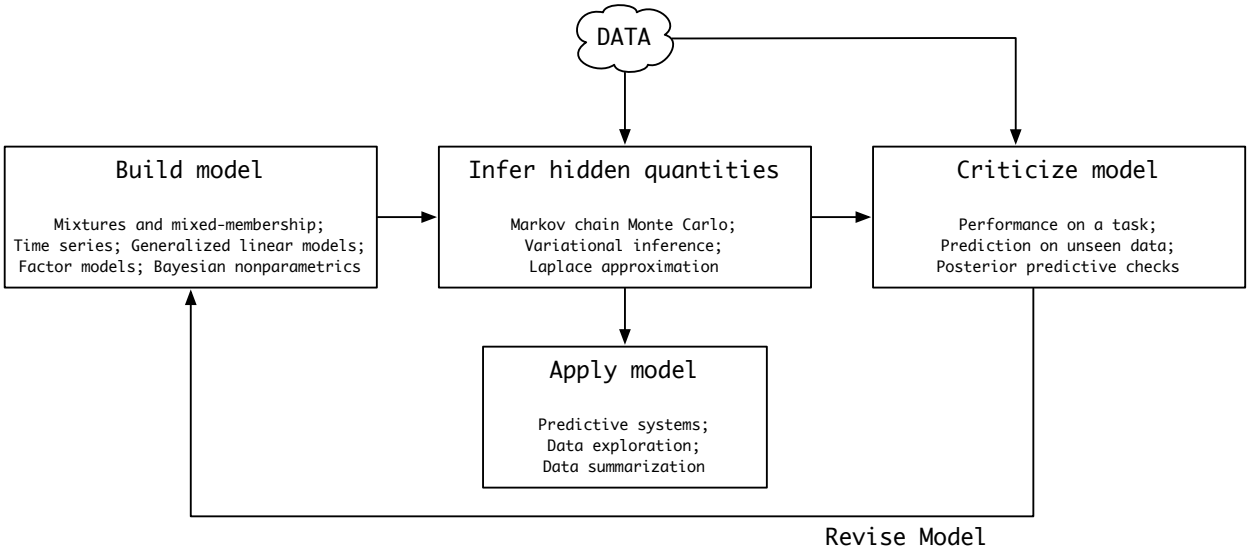


Figure 1: Box's loop.

important ways that it succeeds and fails. If satisfied, use the model to solve the problem; if not satisfied, revise the model according to the results of the criticism and repeat the cycle.

Building and computing with models is part of an iterative process for solving data analysis problems. Figure 1 illustrates this process. Probabilistic programming is the tool we need to be able to use and execute this pipeline. A user encodes her assumptions in a probabilistic program; she uses powerful inference algorithms to analyze a data stream, forming posterior and predictive distributions; she uses tools to evaluate and revise the model. She iterates through this process several times. Finally, she uses the revised model to explore data, form predictions, and ties its calculations into important applications.

Our particular goals were to bring **scalable computation**, **streaming computation**, and **model diagnosis and fitness** into probabilistic programming systems. The previous state of the art lacks these capabilities, which are essential for solving modern data science problems. In parallel, we worked on particular applications—real-world applications motivate our research and ensure that our developments are ones that have practical impact and importance. Finally, we also focused on deploying our methods in usable systems. We developed a new probabilistic programming system called *Edward*, which is based on Google's TensorFlow library, and we deployed a variational inference algorithm in *Stan*, which is a very popular probabilistic programming system.

In the following report, we outline our main accomplishments:

- We developed new generic variational inference algorithms that scale to massive data. Our methods are based on **stochastic variational inference**, a scalable methodology for approx-

imate inference that we pioneered. Specifically, we developed new **generic variational inference** algorithms that easily integrate with probabilistic programming and that can be adapted to the stochastic setting. Our algorithms interleave **intelligent data collection** with **data computation**.

- We expanded the applicability of probabilistic models to **streaming data**, an innovation that is essential for modern applications. This required new ways of thinking about probabilistic models and new fundamental algorithms for interfacing models and streams. Streaming probabilistic models enable **life-long learning systems**, probabilistic models that continually observe, analyze, and act on data.

- We developed new scalable methods for calculating model fitness and model diagnostics. We revived and modernized **posterior predictive checks** and **predictive sample reuse**, two ideas that focus on the discrepancy between data sampled from the predictive distribution and true observations. In model diagnostics, we developed discrepancy functions that point the user to where her model fits and misfits. We developed composable discrepancies for **automating model diagnosis** in complex models.

- We developed new model classes, including **deep probabilistic models**, **Bayesian nonparametric models**, and **generalized embedding models**. We developed a suite of tools for making probabilistic models **robust** to misspecification.

- We worked on real applications in text analysis, recommendation systems, neuroscience, genetics, healthcare, network science, and economics. In each, we developed new probabilistic models, stress-tested our algorithms on them, and then improved the algorithms both for the model at hand and the wider model class. These applications pushed the state of the art of what is possible with probabilistic machine learning.

Our innovations made probabilistic programming widely accessible to data scientists in all domains. Our work allows them to build, revise, select, and incorporate sophisticated probabilistic models as a core component of their data analysis process.

Probabilistic programming systems can revolutionize modern data analysis, putting the powerful tools of model building, fitting, and revising into the hands of anyone seeking insights from their data. With the fundamental innovations that we developed, modern probabilistic programming systems will realize their potential.

3 Methods, Assumptions, and Procedures

Our work is in the framework of probabilistic machine learning. We review probabilistic machine learning and the central computational problems for a probabilistic programming system.

3.1 Probabilistic Machine Learning

Probabilistic machine learning uses probabilistic models to analyze data. In the process of probabilistic modeling, we first develop a joint distribution of hidden and observed variables that captures our assumptions about how the data arises and how it interacts with structures we cannot observe. We then analyze our data by computing the conditional distribution of the hidden variables given the observations. This distribution, called the posterior, lets us examine the hidden structure that was likely to lead to the observed data, to form a predictive distribution of new data, and to check our model for comparison to others and direction of misfit. We then revise the model and continue with the analysis. We call this cycle "Box's loop." (Blei, 2014).

The key technical problems to probabilistic modeling are how to compute the posterior, how to assess the quality of a model, and how to revise it based on our observations. As we describe in the next section, we developed fundamental new methods for posterior computation, model fitness, and model diagnostics. We deployed these methods to several applications.

Generically, let $x_{1:N}$ be N observations and divide the hidden variables into *global variables* β and *local variables* $z_{1:N}$. The global variables—like the mixture locations in a Gaussian mixture model—describe something about the whole data set. The local variables—like the component assignments—help govern the distribution of each data point, conditionally independent of the others. (Many models contain distinctive sets of variables like this, though not all. For other models, there may only be global hidden variables.) The posterior distribution is

$$p(z_{1:N}, \beta \mid x_{1:N}) = p(\beta) \prod_{n=1}^{N} p(z_n, x_n \mid \beta) / p(x_{1:N}). \tag{1}$$

The numerator is the joint distribution; the denominator is the marginal probability of the observations. Computing this posterior is the problem of *posterior inference*. For many models of interest, the denominator is not tractable to compute—it usually is construed as a complicated integral that marginalizes out the hidden variables—and we must resort to *approximate inference*.

The posterior is critical in the *predictive distribution* of new data given the observed data. In the

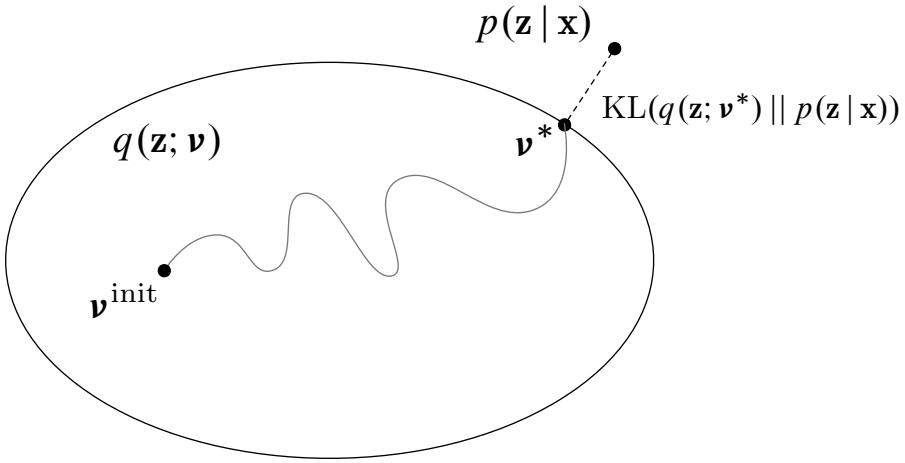


Figure 2: A schematic of variational inference.

predictive distribution, we marginalize out the hidden variables via the posterior,

$$p(x \mid x_{1:N}) = \int \left(\int p(z \mid \beta) p(x \mid z, \beta) dz \right) p(\beta \mid x_{1:N}) d\beta. \tag{2}$$

The inner integral is over the local hidden variables of the new data point; the outer integral is over the posterior of the global variables given the observed data set. This predictive distribution is used for both forming predictions and for implementing our proposed methods for assessing model fitness and developing model diagnostics.

3.2 Background: Variational Inference

In machine learning, there are two main methods for approximating the conditional—Markov chain Monte Carlo (MCMC) and variational inference. In MCMC, we form a Markov chain whose stationary distribution is the conditional, run the chain until it has "converged" (determining this convergence precisely is not usually possible), and then collect independent samples from which to approximate the posterior. MCMC is powerful, and has been widely studied, especially in Bayesian statistics. It is implemented in most existing probabilistic programming systems.

We built on *variational inference*, a deterministic alternative to MCMC that replaces sampling with optimization. Variational inference has been shown to be empirically faster than MCMC in several settings, though it is difficult to formally compare them. Mean-field variational inference provided the foundation for our research, though also extended beyond this assumption (see Section 4.3.) Building on this method, we dramatically sped up and expanded the scope of generic approximate inference algorithms, and thus made probabilistic programming much more scalable.

Here we review the basics of mean-field variational inference.

The idea is to posit a factorized distribution of the hidden variables that is indexed by free *variational parameters*, $q(z_{1:N}, \beta) = q(\beta \mid \lambda) \prod_{n=1}^{N} q(z_n \mid \phi_n)$. These parameters—the local variational parameters ϕ_n and global variational parameter λ—are fit to make $q(z_{1:N}, \beta)$ close in Kullback-Leibler (KL) divergence to the true posterior $p(\beta, z_{1:N} \mid x_{1:N})$. We then use the fitted q as a proxy for the posterior, e.g., in a predictive distribution of new data or to explore the hidden structure of the observations.

But the KL is not computable. Variational methods optimize the *evidence lower bound* (ELBO),

$$\mathcal{L}(\lambda, \phi_{1:N}) = \mathbb{E}_q[\log p(\beta, z_{1:N})] + \mathbb{H}(q), \tag{3}$$

where $\mathbb{H}(\cdot)$ is the entropy of the distribution q. This objective is equal to the negative KL plus a constant; thus maximizing it is equivalent to minimizing KL. Note that the variational "model" is not a model of data, but rather a flexible family of distributions over the latent variables. The connection to the data and to the posterior is via optimizing the ELBO with respect to that family.

Figure 2 illustrates the main idea behind variational inference (VI). There is a *variational family* of distributions of latent variables. It is indexed by *variational parameters*; each setting of the variational parameters is a distribution of latent variables. We want to approximate the *exact posterior*, which is outside the variational family. VI begins at an *initial setting* of the variational parameters; it then *optimizes* them to find the member of the family that is closest to the exact posterior. Closeness is measured by the *KL divergence*; the KL is the objective of the optimization. In our research on variational inference, we consider and develop each piece of this framework. Indeed, the accomplishments described below can all be seen as improving one aspect of this algorithmic idea.

Typical applications of variational inference optimize the ELBO using coordinate ascent, iteratively optimizing each variational parameter. These updates are in closed form for models where each complete conditional is in the exponential family. (A complete conditional is the distribution of a hidden variable given all the other variables in the model.) But these methods are not useful in probabilistic programming, where the user should be able to express models from a much wider class without regard for the specific form of the complete conditionals. Further, each application of variational inference has required painstaking derivation and mathematics. This goes against the philosophy of a PPS to make modern machine learning accessible to a wide audience of users.

4 Results and Discussion

In this section, we detail each of our accomplishments as part of the Defense Advanced Research Project Agency (DARPA) PPAML project.

4.1 Stochastic Variational Inference

We developed stochastic variational inference (SVI), a scalable algorithm for approximating posterior distributions (Hoffman et al., 2013). We developed this technique for a large class of probabilistic models and we demonstrated it with two probabilistic topic models, latent Dirichlet allocation and the hierarchical Dirichlet process topic model. When developing this procedure, we tested the method on several large collections of documents: 300K articles from Nature, 1.8M articles from The New York Times, and 3.8M articles from Wikipedia. Stochastic inference easily handled data sets of this size and outperformed traditional variational inference, which can only handle a smaller subset. Stochastic variational inference, which has become a widely used algorithm, lets us apply complex Bayesian models to massive data sets.

Throughout the project, we continued to build on SVI. Here are some of the notable results.

Adaptive algorithms. In several related results, we made SVI more adaptive. In Houlsby and Blei (2014), we presented an alternative perspective on SVI as approximate parallel coordinate ascent. SVI trades-off bias and variance to step close to the unknown true coordinate optimum given by batch variational Bayes (VB). We defined a model to automate this process. The model infers the location of the next VB optimum from a sequence of noisy realizations. As a consequence of this construction, we update the variational parameters using Bayes rule, rather than a hand-crafted optimization schedule. When our model is a Kalman filter this procedure can recover the original SVI algorithm and SVI with adaptive steps. We may also encode additional assumptions in the model, such as heavy-tailed noise. By doing so, our algorithm outperforms the original SVI schedule and a state-of-the-art adaptive SVI algorithm in two different domains.

Ranganath et al. (2013) studied adaptive learning rates for SVI. Operationally, stochastic inference iteratively subsamples from the data, analyzes the subsample, and updates parameters with a decreasing learning rate. However, the algorithm is sensitive to that rate, which usually requires hand-tuning to each application. We solved this problem by developing an adaptive learning rate for stochastic inference. Our method requires no tuning and is easily implemented with computations already made in the algorithm. We demonstrated our approach with latent Dirichlet allocation

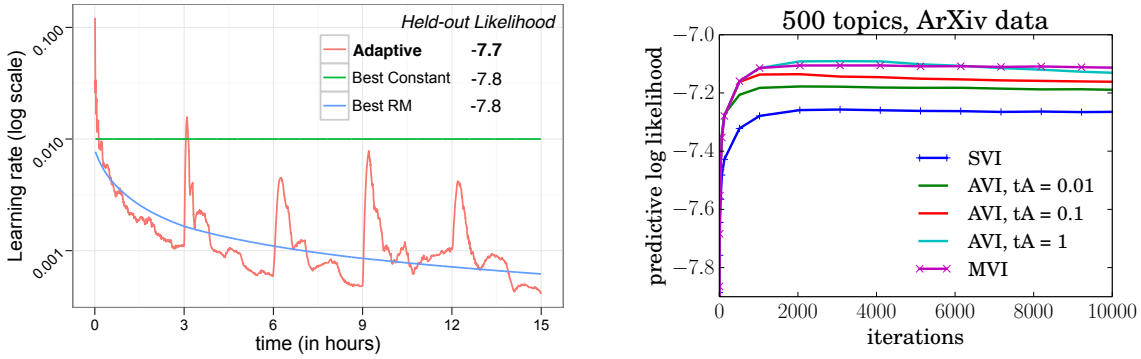


Figure 3: Adaptive approaches to stochastic variational inference.

applied to three large text corpora. Inference with the adaptive learning rate converges faster and to a better approximation than the best settings of hand-tuned rates.

Figure 3 (left) illustrates these results, the adaptive learning rate on a run of stochastic variational inference, compared to the best Robbins-Monro and best constant learning rate. Here the data arrives non-uniformly, changing its distribution every three hours. (The algorithms do not know this.) The adaptive learning rate spikes when the data distribution changes. This leads to better predictive performance, as indicated by the held-out likelihood in the top right.

Finally, Mandt et al. (2016b) developed variational tempering, an annealing approach to SVI. We first formulated a deterministic annealing approach for the generic class of conditionally conjugate exponential family models. This approach uses a decreasing temperature parameter which deterministically deforms the objective during the course of the optimization. A well-known drawback to this annealing approach is the choice of the cooling schedule. We therefore introduced variational tempering, a variational algorithm that introduces a temperature latent variable to the model. In contrast to related work in the Markov chain Monte Carlo literature, this algorithm results in adaptive annealing schedules. Lastly, we developed local variational tempering, which assigns a latent temperature to each data point; this allows for dynamic annealing that varies across data. Compared to the traditional VI, all proposed approaches find improved predictive likelihoods on held-out data.

Figure 3 (right) illustrates these results. We compare SVI, against variational tempering (also known as multicanonical variational inference, MVI) and annealed variational inference (AVI) for different temperature schedules. We plot the learning curves for latent Dirichlet allocation (LDA) on Arxiv text data. Adaptive methods perform better than SVI.

Smoothed gradients. Mandt and Blei (2014) developed smoothed gradients for SVI. As with

most traditional stochastic optimization methods, SVI takes precautions to use unbiased stochastic gradients whose expectations are equal to the true gradients. We developed the idea of following biased stochastic gradients in SVI. Our method replaces the natural gradient with a similarly constructed vector that uses a fixed-window moving average of some of its previous terms. We demonstrated many advantages of this technique. First, its computational cost is the same as for SVI and storage requirements only multiply by a constant factor. Second, it enjoys significant variance reduction over the unbiased estimates, smaller bias than averaged gradients, and leads to smaller mean-squared error against the full gradient. We tested this method on latent Dirichlet allocation with three large corpora.

Streaming SVI. In McInerney et al. (2015) we developed streaming SVI. Many modern data analysis problems involve inferences from streaming data. However, streaming data is not easily amenable to the standard probabilistic modeling approaches, which require conditioning on finite data. We developed "population variational Bayes," a new approach for using Bayesian modeling to analyze streams of data. It approximates a new type of distribution, the population posterior, which combines the notion of a population distribution of the data with Bayesian inference in a probabilistic model. We developed the population posterior for latent Dirichlet allocation and Dirichlet process mixtures. We studied our method with several large-scale data sets.

SVI for social networks. We adapted SVI to do inference in machine learning problems for community detection in massive networks (Gopalan and Blei, 2013). Detecting overlapping communities is essential to analyzing and exploring natural networks such as social networks, biological networks, and citation networks. However, most existing approaches do not scale to the size of networks that we regularly observe in the real world. We developed a scalable approach to community detection that discovers overlapping communities in massive real-world networks. We demonstrated how we can discover the hidden community structure of several real-world networks, including 3.7 million US patents, 575,000 physics articles from the arXiv preprint server, and 875,000 connected Web pages from the Internet. Furthermore, we demonstrated on large simulated networks that our algorithm accurately discovers the true community structure. This result opened the door to using sophisticated statistical models to analyze massive networks.

Figure 4 illustrates some of the scalable inferences that we can make with this method. The figure shows the top four link communities that include citations to "An alternative to compactification", an article that bridges several communities. We visualize the links between the articles and show some highly cited titles. Each community is labeled with its dominant subject area; nodes are sized by their "bridgeness," an inferred measure of their impact on multiple communities. This is taken from an analysis of the full 575,000 node network.

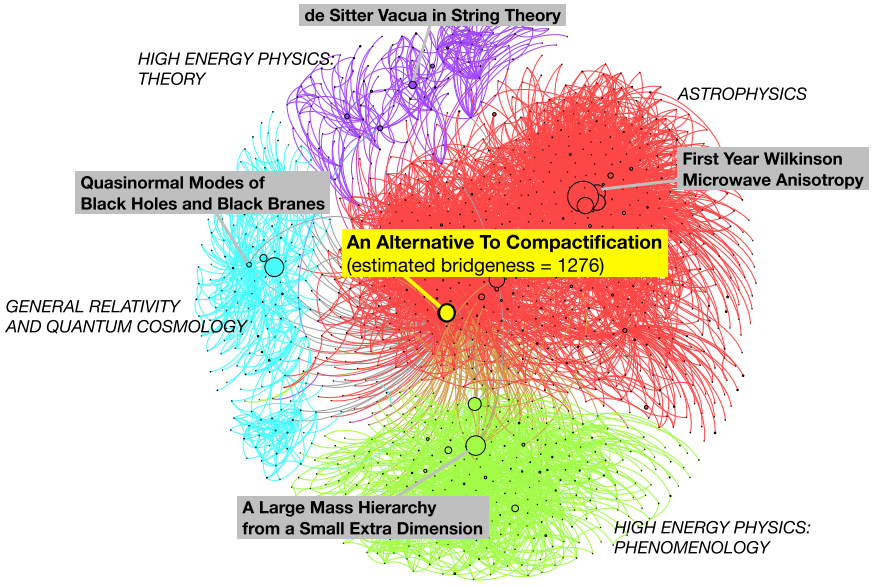


Figure 4: The discovered community structure in a subgraph of the arXiv citation network.

SVI for population genetics. We also adapted the method to population genetics (Gopalan et al., 2016). A major goal of population genetics is to quantitatively understand variation of genetic polymorphisms among individuals. The aggregated number of genotyped humans is currently on the order of millions of individuals, and existing methods do not scale to data of this size. To solve this problem, we developed TeraStructure, an SVI algorithm to fit Bayesian models of genetic variation in structured human populations on tera-sample-sized data sets (10^{12} observed genotypes; for example, million individuals at million single nucleotide polymorphisms (SNPs)). We demonstrated that TeraStructure performs as well as existing methods on current globally sampled data, and we showed using simulations that TeraStructure continues to be accurate and is the only method that can scale to tera-sample sizes.

4.2 Black Box Variational Inference

Scaling up variational inference is important for real-world applications. But equally important, especially for probabilistic programming, is to develop *generic* variational inference, inference methods that can be easily adapted to large classes of models. In a long thread of research, my lab has been developing such methods. The vision is that, with generic inference methods, we can build the probabilistic programming systems that implement them on programs, i.e., a model class.

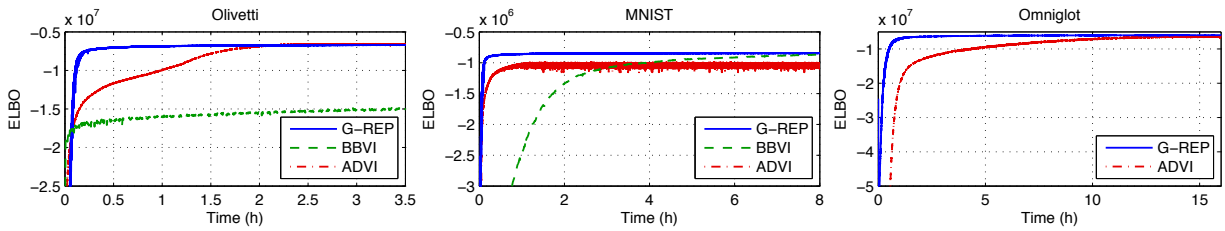
Our first accomplishment on this problem is Ranganath et al. (2014). It addresses the problem that deriving a variational inference algorithm generally required significant model-specific analysis and these efforts can hinder and deter us from quickly developing and exploring a variety of models for a problem at hand. We developed a black box variational inference (BBVI) algorithm, one that can be quickly applied to many models with little additional derivation. Our method is based on a stochastic optimization of the variational objective where the noisy gradient is computed from Monte Carlo samples from the variational distribution. (Contrast this to the use of stochastic optimization in Section 4.1, where it is data subsampling that provides the stochasticity.)

In BBVI, the variance of the noisy gradients is a problem. We developed a number of methods to reduce the variance of the gradient, always maintaining the criterion that we want to avoid difficult model-based derivations. In Ranganath et al. (2014), we evaluated our method against the corresponding black box sampling based methods. We found that it reaches better predictive likelihoods much faster than sampling methods. We demonstrated that BBVI lets us easily explore a wide space of models by quickly constructing and evaluating several models of longitudinal healthcare data.

Since the introduction of BBVI, this approach has been widely adapted and used. In our own work, we have innovated on this basic idea in several ways.

Overdispersed black box variational inference. In Ruiz et al. (2016b), we introduced overdispersed black-box variational inference, a method to reduce the variance of the Monte Carlo estimator of the gradient in black-box variational inference. Instead of taking samples from the variational distribution, we use importance sampling to take samples from an overdispersed distribution in the same exponential family as the variational approximation. Our approach is general since it can be readily applied to any exponential family distribution, which is the typical choice for the variational approximation. We ran experiments on two non-conjugate probabilistic models to show that our method effectively reduces the variance, and the overhead introduced by the computation of the proposal parameters and the importance weights is negligible. We found that our overdispersed importance sampling scheme provides lower variance than black-box variational inference, even when the latter uses twice the number of samples. This results in faster convergence of the black-box inference procedure.

Innovations on reparameterization gradients. Within BBVI, there are two ways to calculate approximate gradients—the score gradient and the reparameterization gradient. The reparameterization gradient is a widely used method to obtain Monte Carlo gradients to optimize the variational objective. However, this technique does not easily apply to commonly used distributions such as beta or gamma without further approximations, and most practical applications of the reparameterization



(a) ELBO (Olivetti dataset). **(b)** ELBO (MNIST dataset). **(c)** ELBO (Omniglot dataset).

Figure 5: Empirical study of the generalized reparameterization gradient.

gradient fit Gaussian distributions. Ruiz et al. (2016a) introduced the generalized reparameterization gradient, a method that extends the reparameterization gradient to a wider class of variational distributions. Generalized reparameterizations use invertible transformations of the latent variables which lead to transformed distributions that weakly depend on the variational parameters. This results in new Monte Carlo gradients that combine reparameterization gradients and score function gradients. We demonstrated our approach on variational inference for two complex probabilistic models. The generalized reparameterization is effective: even a single sample from the variational distribution is enough to obtain a low-variance gradient.

Figure 5 illustrates the advantages of this approach versus BBVI and autodifferentiation variational inference (ADVI, see below). It compares BBVI, ADVI, and the generalized reparameterization gradient in terms of the variational objective. The generalized reparameterization gradient outperforms BBVI because BBVI has not converged in the allowed time; it converges faster than ADVI.

The reparameterization trick is applicable when we can simulate a random variable by applying a (differentiable) deterministic function on an auxiliary random variable whose distribution is fixed. But for many distributions of interest (again, such as the gamma or Dirichlet), simulation of random variables relies on rejection sampling and the discontinuity introduced by the accept-reject step means that standard reparameterization tricks are not applicable. Naesseth et al. (2017) proposed a new method that lets us leverage reparameterization gradients even when variables are outputs of a rejection sampling algorithm. Like the work described above, this approach enables reparameterization on a larger class of variational distributions. In several studies of real and synthetic data, we showed that the variance of the estimator of the gradient is significantly lower than other state-of-the-art methods. This leads to faster convergence of stochastic optimization variational inference. (This paper won the "Best Student Paper Award" at AISTATS 2017.)

Proximity variational inference. Altosaar et al. (2018) developed proximity variational inference (PVI). It solves the problem that VI is sensitive to initialization and can be subject to poor local optima. Proximity variational inference is a method for optimizing the variational objective that

constrains subsequent iterates of the variational parameters to robustify the optimization path. Consequently, PVI is less sensitive to initialization and optimization quirks and finds better local optima. We demonstrated our method on four proximity statistics. We study PVI on a Bernoulli factor model and sigmoid belief network fit to real and synthetic data and compared to deterministic annealing (see above). We highlighted the flexibility of PVI by designing a proximity statistic for Bayesian deep learning models such as the variational autoencoder and showed that it gives better performance by reducing overpruning. PVI also yields improved predictions in a deep generative model of text. Empirically, we showed that PVI consistently finds better local optima and gives better predictive performance.

Augment and reduce. Ruiz et al. (2018) developed the augment-and-reduce method to scale up BBVI for high-dimensional categorical distributions. Categorical distributions are ubiquitous in machine learning, e.g., in classification, language models, and recommendation systems. However, when the number of possible outcomes is very large, using categorical distributions becomes computationally expensive, as the complexity scales linearly with the number of outcomes. We proposed augment and reduce (A&R), a method to alleviate the computational complexity. A&R uses two ideas: latent variable augmentation and stochastic variational inference. It maximizes a lower bound on the marginal likelihood of the data. Unlike existing methods which are specific to softmax, A&R is more general and is amenable to other categorical models, such as multinomial probit. On several large-scale classification problems, we showed that A&R provides a tighter bound on the marginal likelihood and has better predictive performance than existing approaches.

BBVI for implicit models. BBVI was an innovation in expanding VI to evaluable models. But in some fields—physics and ecology come immediately to mind—models cannot be evaluated from, only sampled. To this end Tran et al. (2017b) developed variational inference for implicit models. Implicit probabilistic models are a flexible class of models defined by a simulation process for data. They form the basis for theories which encompass our understanding of the physical world. Despite this fundamental nature, the use of implicit models remains limited due to challenges in specifying complex latent structure in them, and in performing inferences in such models with large data sets. We first introduced hierarchical implicit models (HIMs). HIMs combine the idea of implicit densities with hierarchical Bayesian modeling, thereby defining models via simulators of data with rich hidden structure. Next, we developed likelihood-free variational inference (LFVI), a scalable variational inference algorithm for HIMs. Key to LFVI is specifying a variational family that is also implicit. This matches the model's flexibility and allows for accurate approximation of the posterior. We demonstrated diverse applications: a large-scale physical simulator for predator-prey populations in ecology; a Bayesian generative adversarial network for discrete data; and a deep implicit model for text generation.

Nonconjugate variational inference. Finally, in a related theme to the goals of BBVI, Wang and Blei (2013) developed coordinate ascent variational inference for nonconjugate models. We developed two generic methods for nonconjugate models, Laplace variational inference and delta method variational inference. Our methods have several advantages: they allow for easily derived variational algorithms with a wide class of nonconjugate models; they extend and unify some of the existing algorithms that have been derived for specific models; and they work well on real-world datasets. We studied our methods on the correlated topic model, Bayesian logistic regression, and hierarchical Bayesian logistic regression. Though they do not satisfy the black box criteria, they work on the same class of models and can be faster (though with more of the investigator's effort).

4.3 Improving the Fidelity of Variational Inference

We have described our results around scaling variational inference and making it easily applicable to large classes of probabilistic models. Another important thread of our research activities revolved around making variational inference more accurate, i.e., increasing the fidelity of the approximation.

Structured stochastic variational inference. Hoffman and Blei (2015) developed structured SVI. The first SVI algorithm (Section 4.1 relies on the use of fully factorized variational distributions. However, this "mean-field" independence approximation limits the fidelity of the posterior approximation, and introduces local optima. We showed how to relax the mean-field approximation to allow arbitrary dependencies between global parameters and local hidden variables, producing better parameter estimates by reducing bias, sensitivity to local optima, and sensitivity to hyperparameters.

Variational inference with copula augmentation. Tran et al. (2015) developed a copula approach to variational inference, which preserves dependency among the latent variables. Our method uses copulas to augment the families of distributions used in mean-field and structured approximations. Copulas model the dependency that is not captured by the original variational distribution, and thus the augmented variational family guarantees better approximations to the posterior. With stochastic optimization, inference on the augmented distribution is scalable. Furthermore, our strategy is generic: it can be applied to any inference procedure that currently uses the mean-field or structured approach. Copula variational inference has many advantages: it reduces bias; it is less sensitive to local optima; it is less sensitive to hyperparameters; and it helps characterize and interpret the dependency among the latent variables.

	Model	HVM	Mean-Field
Poisson	100	**3386**	3387
	100-30	**3396**	3896
	100-30-15	**3346**	3962
Bernoulli	100	**3060**	3084
	100-30	3394	**3339**
	100-30-15	**3420**	3575

	Model	HVM	Mean-Field
Poisson	100	**3327**	3392
	100-30	**2977**	3320
	100-30-15	**3007**	3332
Bernoulli	100	**3165**	3166
	100-30	**3135**	3195
	100-30-15	**3050**	3185

Figure 6: Empirical study of hierarchical variational models.

Variational Gaussian process. Building on this theme, Tran et al. (2016) developed the variational Gaussian process (VGP), a Bayesian nonparametric variational family, which adapts its shape to match complex posterior distributions. The VGP generates approximate posterior samples by generating latent inputs and warping them through random non-linear mappings; the distribution over random mappings is learned during inference, enabling the transformed outputs to adapt to varying complexity. We proved a universal approximation theorem for the VGP, demonstrating its representative power for learning any model. For inference we presented a variational objective inspired by auto-encoders and perform black box inference over a wide class of models. At that time, the VGP achieved new state-of-the-art results for unsupervised learning, inferring models such as the deep latent Gaussian model and the deep recurrent attentive writer (DRAW) model.

Hierarchical variational models. This line of work on improving the fidelity of variational methods culminated in our research in hierarchical variational models (HVMs) (Ranganath et al., 2016c). HVMs augment a variational approximation with a prior on its parameters, which allows it to capture complex structure for both discrete and continuous latent variables. The algorithm we developed is black box, can be used for any HVM, and has the same computational efficiency as the original approximation. We studied HVMs on a variety of deep discrete latent variable models. HVMs generalize other expressive variational distributions and maintain higher fidelity to the posterior.

Figure 6 shows results on the deep exponential families; HVMs are the best way to do inference in this model. On the left is *New York Times* held-out perplexity (lower is better). HVM outperform mean-field in five models. Mean-field (Ranganath et al., 2015) fails at multi-level Poissons; HVM make it possible to study multi-level Poissons. On the right is *Science*. HVM outperforms mean-field on all six models. HVM identify that multi-level Poisson models are best, while mean-field does not.

Variational sequential Monte Carlo. The success of variational approaches depends on (i)

formulating a flexible parametric family of distributions, and (ii) optimizing the parameters to find the member of this family that most closely approximates the exact posterior. In parallel work to the above, Naesseth et al. (2018) developed a new approximating family of distributions, the variational sequential Monte Carlo (VSMC) family, and showed how to optimize it in variational inference. VSMC melds variational inference (VI) and sequential Monte Carlo (SMC), providing practitioners with flexible, accurate, and powerful Bayesian inference. The VSMC family is a variational family that can approximate the posterior arbitrarily well, while still allowing for efficient optimization of its parameters. We demonstrated its utility on state space models, stochastic volatility models for financial data, and deep Markov models of brain neural circuits.

4.4 Theoretical Results in Variational Inference

Stochastic inference, black box variational inference, and high-fidelity variational inference were the main themes of our practical accomplishments in approximate posterior inference. But for VI to be trusted as a viable method, it requires both practical success and a theoretical understanding. As part of the PPAML project, we developed two results around a theoretical understanding of VI. One connects VI to the broader world of Bayesian statistics; the other relates the popular algorithm of stochastic gradient descent (with a constant step size) to a variational approximation of the posterior. We also developed new objective functions for VI, expanding the theory of what it means to perform inference with optimization.

Consistency of variational inference. Variational Bayes methods have emerged as a popular alternative to the classical Markov chain Monte Carlo (MCMC) methods. VB methods tend to be faster while achieving comparable predictive performance. However, there are few theoretical results around the statistical properties of VB. Wang and Blei (pear) established frequentist consistency and asymptotic normality of VB methods. Specifically, we connected VB methods to point estimates based on variational approximations, called frequentist variational approximations, and we use the connection to prove a variational Bernstein-von Mises theorem. The theorem leverages the theoretical characterizations of frequentist variational approximations to understand asymptotic properties of VB. In summary, we proved that (1) the VB posterior converged to the Kullback-Leibler (KL) minimizer of a normal distribution, centered at the truth and (2) the corresponding variational expectation of the parameter is consistent and asymptotically normal. As applications of the theorem, we derived asymptotic properties of VB posteriors in Bayesian mixture models, Bayesian generalized linear mixed models, and Bayesian stochastic block models. We illustrated these theoretical results with a simulation study.

Variational inference and stochastic gradient descent. Stochastic Gradient Descent with a constant learning rate (constant SGD) simulates a Markov chain with a stationary distribution. With this perspective, Mandt et al. (2016a, 2017) derived several new results. (1) We showed that constant SGD can be used as an approximate Bayesian posterior inference algorithm. Specifically, we showed how to adjust the tuning parameters of constant SGD to best match the stationary distribution to a posterior, minimizing the Kullback-Leibler divergence between these two distributions. (2) We demonstrated that constant SGD gives rise to a new variational EM algorithm that optimizes hyperparameters in complex probabilistic models. (3) We also showed how to tune SGD with momentum for approximate sampling. (4) We analyzed stochastic-gradient MCMC algorithms. For Stochastic-Gradient Langevin Dynamics and Stochastic-Gradient Fisher Scoring, we quantified the approximation errors due to finite learning rates. Finally (5), we used the stochastic process perspective to give a short proof of why Polyak averaging is optimal. Based on this idea, we proposed a scalable approximate MCMC algorithm, the Averaged Stochastic Gradient Sampler, which can be seen as a variational MCMC hybrid.

New objective functions for variational inference. In two related papers, we developed new objective functions for VI, in both cases seeking to alleviate some of the theoretical issues of the KL divergence. (That said, those theoretical issues do not always appear to be practical issues and the classical KL divergence is still the most efficient variational objective function.)

In variational inference, closeness is usually measured via the KL divergence $D(q||p)$ from the variational approximation q to the exact posterior p. While successful, this approach also has problems. Notably, it typically leads to underestimation of the posterior variance. In Dieng et al. (2017), we proposed χ-divergence variational inference (ChiVI), a black-box variational inference algorithm that minimizes $D_\chi(p||q)$, the χ-divergence from p to q. ChiVI minimizes an upper bound of the model evidence, which we term the χ upper bound (CUBO). Minimizing the CUBO leads to improved posterior uncertainty, and it can also be used with the classical VI lower bound (ELBO) to provide a sandwich estimate of the model evidence. We studied ChiVI on three models: probit regression, Gaussian process classification, and a Cox process model of basketball plays. When compared to expectation propagation and classical VI, ChiVI produced better error rates and more accurate estimates of posterior variance.

As we mentioned, variational inference is an umbrella term for algorithms which cast Bayesian inference as optimization. Ranganath et al. (2016a) reexamined variational inference from its roots as an optimization problem. We used operators, or functions of functions, to design new variational objectives. As one example, we designed a variational objective with a Langevin-Stein operator. We developed a black box algorithm, operator variational inference (OPVI), for optimizing any operator

objective. Importantly, operators enable us to make explicit the statistical and computational tradeoffs for variational inference. We can characterize different properties of variational objectives, such as objectives that admit data subsampling—allowing inference to scale to massive data—as well as objectives that admit variational programs—a rich class of posterior approximations that does not require a tractable density. We illustrated the benefits of OPVI on a mixture model and a generative model of images.

4.5 Variational Inference and Probabilistic Programming

Putting all these results into practice. We developed a new probabilistic programming language, Edward, and adapted variational inference to a popular probabilistic programming language, Stan. We used Edward and Stan extensively in the other research cited here, both to apply our innovations and to develop new methodology.

Variational inference in Stan. For Stan, Kucukelbir et al. (2015, 2017a) developed automatic differentiation variational inference (ADVI). Using this method, a scientist need only provide a probabilistic model and a dataset, nothing else. ADVI automatically derives an efficient variational inference algorithm, freeing the scientist to refine and explore many models. ADVI is a black-box method and supports a broad class of models—no conjugacy assumptions are required. We studied ADVI across ten modern probabilistic models and applied it to a dataset with millions of observations. We deployed ADVI as part of Stan, a probabilistic programming system.

Figure 7 shows an example analysis of a large dataset with VI in Stan. The model is a mixture of probabilistic principal component analysis (pPCA), a complex nonconjugate model. The analysis is possible because of ADVI.

Edward. Tran et al. (2017a) proposed Edward, a Turing-complete probabilistic programming language. Edward defines two compositional representations—random variables and inference. By treating inference as a first class citizen, on a par with modeling, we showed that probabilistic programming can be as flexible and computationally efficient as traditional deep learning. For flexibility, Edward makes it easy to fit the same model using a variety of composable inference methods, ranging from point estimation to variational inference to MCMC. In addition, Edward can reuse the modeling representation as part of inference, facilitating the design of rich variational models and generative adversarial networks. For efficiency, Edward is integrated into TensorFlow, providing significant speedups over existing probabilistic systems. For example, we showed on a benchmark logistic regression task that Edward is faster than Stan and PyMC3. Further, Edward incurs no runtime overhead: it is as fast as handwritten TensorFlow.

Figure 7: An analysis of 1.7M taxi trajectories in Stan.

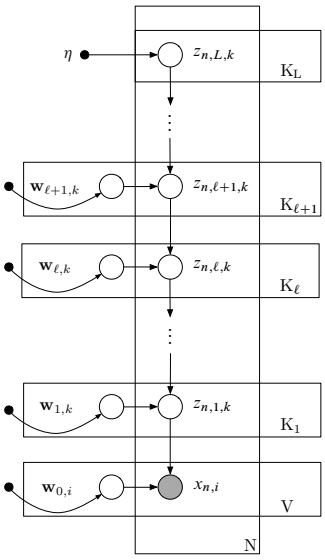
4.6 Designing New Models

Developing probabilistic programming more broadly also involves the other aspects of Box's loop, designing models, checking models, and applying models to real world data.

In designing models, we have developed three new classes of probabilistic models—correlated random measures, deep exponential families, and probabilistic embeddings. We also developed several methods to strengthen models, i.e., to make them more robust to data misfit. Finally, we developed new methods for checking models, in particular reviving posterior predictive checks for analyzing models of text and genetics.

Deep exponential families. Ranganath et al. (2015) describe deep exponential families (DEFs), a class of latent variable models that are inspired by the hidden structures used in deep neural networks. DEFs capture a hierarchy of dependencies between latent variables, and are easily generalized to many settings through exponential families. We performed inference using black box variational inference techniques, exploring many settings of the different parameters of a DEF. We evaluated various DEFs on text and combined multiple DEFs into a model for pairwise recommendation data. In an extensive study, we showed that going beyond one layer improves predictions for DEFs. We demonstrated that DEFs find interesting exploratory structure in large data sets, and give better predictive performance than state-of-the-art models.

Figure 8 shows the probabilistic graphical model for a DEF and results on using DEFs to analyze



Model	**W**	*NYT*	*Science*
LDA [6]		2717	1711
DocNADE [19]		2496	1725
Sparse Gamma 100	$\emptyset$	2525	1652
Sparse Gamma 100-30	Γ	2303	1539
Sparse Gamma 100-30-15	Γ	2251	1542
Sigmoid 100	$\emptyset$	2343	1633
Sigmoid 100-30	$\mathcal{N}$	2653	1665
Sigmoid 100-30-15	$\mathcal{N}$	2507	1653
Poisson 100	$\emptyset$	2590	1620
Poisson 100-30	$\mathcal{N}$	2423	1560
Poisson 100-30-15	$\mathcal{N}$	2416	1576
Poisson log-link 100-30	Γ	2288	1523
Poisson log-link 100-30-15	Γ	2366	1545

Figure 8: Deep exponential families.

text. The table shows how deep exponential families and black box variational inference let us analyze many different DEF structures. DEF models of text outperform existing methods. The table reports perplexity on a held out collection of 1K Science and NYT documents. Lower values are better. The DEF W column indicates the type of prior distribution over the DEF weights, for the gamma prior and $\mathcal{N}$ for normal. (Recall that one layer DEFs consist only of a layer of latent variables, thus we represent their prior with the empty set.)

Bayesian nonparametrics. In a thread of research, we continued to develop Bayesian nonparametric models. These are models that grow and change with the data, adapting their structure to the data at hand.

Ranganath and Perotte (2018) developed correlated random measures, random measures where the atom weights can exhibit a flexible pattern of dependence, and used them to develop powerful hierarchical Bayesian nonparametric models. Hierarchical Bayesian nonparametric models are usually built from completely random measures, a Poisson-process based construction in which the atom weights are independent. Completely random measures imply strong independence assumptions in the corresponding hierarchical model, and these assumptions are often misplaced in real-world settings. Correlated random measures address this limitation. They model correlation within the measure by using a Gaussian process in concert with the Poisson process. With correlated random measures, for example, we can develop a latent feature model for which we can infer both the properties of the latent features and their dependency pattern. We develop several other examples as well. We studied a correlated random measure model of pairwise count data. We derived an

efficient variational inference algorithm and show improved predictive performance on large data sets of documents, web clicks, and electronic health records.

Paisley et al. (2015) developed a nested hierarchical Dirichlet process (nHDP) for hierarchical topic modeling. The nHDP generalizes the nested Chinese restaurant process (nCRP) to allow each word to follow its own path to a topic node according to a per-document distribution over the paths on a shared tree. This alleviates the rigid, single-path formulation assumed by the nCRP, allowing documents to easily express complex thematic borrowings. We derive a stochastic variational inference algorithm for the model, which enabled efficient inference for massive collections of text documents. We demonstrated the algorithm on 1.8 million documents from The New York Times and 2.7 million documents from Wikipedia.

Polatkan et al. (2015) developed a new Bayesian nonparametric model for super-resolution. Our method uses a beta-Bernoulli process to learn a set of recurring visual patterns, called dictionary elements, from the data. Because it is nonparametric, the number of elements found is also determined from the data. We tested the results on both benchmark and natural images, comparing with several other models from the research literature. We performed large-scale human evaluation experiments to assess the visual quality of the results. In a first implementation, we use Gibbs sampling to approximate the posterior. However, this algorithm was not feasible for large-scale data. To circumvent this, we then developed a stochastic variational inference algorithm. This algorithm finds high quality dictionaries in a fraction of the time needed by the Gibbs sampler.

Latent feature models are widely used to decompose data into a small number of components. Bayesian nonparametric variants of these models, which use the Indian buffet process (IBP) as a prior over latent features, allow the number of features to be determined from the data. Gershman et al. (2015) presented a generalization of the IBP, the distance dependent Indian buffet process (dd-IBP), for modeling non-exchangeable data. It relies on distances defined between data points, biasing nearby data to share more features. The choice of distance measure allows for many kinds of dependencies, including temporal and spatial. Further, the original IBP is a special case of the dd-IBP. We developed the dd-IBP and theoretically characterized its feature-sharing properties. We studied its performance on real-world non-exchangeable data.

Exponential family embeddings. Word embeddings are a powerful approach for capturing semantic similarity among terms in a vocabulary. Rudolph et al. (2016) developed exponential family embeddings, a class of models that extends the idea of word embeddings to other types of high-dimensional data. As examples, we studied neural data with real-valued observations, count data from a market basket analysis, and ratings data from a movie recommendation system. The main idea is to model each observation conditioned on a set of other observations. This set is called

the context, and the way the context is defined is a modeling choice that depends on the problem. In language the context is the surrounding words; in neuroscience the context is close-by neurons; in market basket data the context is other items in the shopping cart. Each type of embedding model defines the context, the exponential family of conditional distributions, and how the latent embedding vectors are shared across data. On all three applications—neural activity of zebrafish, users' shopping behavior, and movie ratings—we found exponential family embedding models to be more effective than other types of dimension reduction. They better reconstruct held-out data and find interesting qualitative structure.

In many follow on papers, we expanded and extended exponential family embeddings. These included to time series (Rudolph and Blei, 2018), to latent contexts (Liu and Blei, 2017), and to hierarchies and groups (Rudolph et al., 2017).

4.7 Checking and Strengthening Models

In addition to designing new model classes, we developed several new methods for checking models and for strengthening them, i.e., to make them robust to deviations from the model assumptions.

We first discuss three methods for addressing robustness; we then discuss three methods for checking models.

Bayesian data reweighting. Probabilistic models analyze data by relying on a set of assumptions. Data that exhibit deviations from these assumptions can undermine inference and prediction quality. Robust models offer protection against mismatch between a model's assumptions and reality. Wang et al. (2017) proposed a way to systematically detect and mitigate mismatch of a large class of probabilistic models. The idea is to raise the likelihood of each observation to a weight and then to infer both the latent variables and the weights from data. Inferring the weights allows a model to identify observations that match its assumptions and down-weight others. This enables robust inference and improves predictive accuracy. We studied four different forms of mismatch with reality, ranging from missing latent groups to structure misspecification. A Poisson factorization analysis of the Movielens 1M dataset showed the benefits of this approach in a practical scenario.

A general approach to robust Bayesian models. As we discussed, robust Bayesian models are appealing alternatives to standard models, providing protection from data that contains outliers or other departures from the model assumptions. Historically, robust models were mostly developed on a case-by-case basis; examples include robust linear regression, robust mixture models, and bursty

topic models. Wang and Blei (2018) developed a general approach to robust Bayesian modeling. We showed how to turn an existing Bayesian model into a robust model, and then developed a generic computational strategy for it. We used our method to study robust variants of several models, including linear regression, Poisson regression, logistic regression, and probabilistic topic models. We discussed the connections between our methods and existing approaches, especially empirical Bayes and James-Stein estimation.

Population empirical Bayes. Bayesian predictive inference employs a model to analyze a dataset and make predictions about new observations. When a model does not match the data, predictive accuracy suffers. Kucukelbir and Blei (2015) developed population empirical Bayes, a hierarchical framework that explicitly models the empirical population distribution as part of Bayesian analysis. We introduce a latent dataset as a hierarchical variable and set the empirical population as its prior. This leads to a new predictive density that mitigates model mismatch. We efficiently applied this method to complex models by proposing a stochastic variational inference algorithm, called bumping variational inference. We demonstrated improved predictive accuracy over classical Bayesian inference in three models: a linear regression model of health data, a Bayesian mixture model of natural images, and a latent Dirichlet allocation topic model of scientific documents.

Bayesian checking of mixed membership models. In two papers we developed new ways to check mixed-membership models, i.e., topic models and their cousins.

The first was to topic models. Real document collections do not fit the independence assumptions asserted by most statistical topic models, but how badly do they violate them? Mimno and Blei (2011) presented a Bayesian method for measuring how well a topic model fits a corpus. Our approach is based on posterior predictive checking, a method for diagnosing Bayesian models in user-defined ways. Our method can identify where a topic model fits the data, where it falls short, and in which directions it might be improved.

The second was to population genetics. Admixture models are a ubiquitous approach to capture latent population structure in genetic samples. But despite the widespread application of admixture models, little thought has been devoted to the quality of the model fit or the accuracy of the estimates of parameters of interest for a particular study. Mimno et al. (2015) developed methods for validating admixture models based on posterior predictive checks (PPCs), a Bayesian method for assessing the quality of fit of a statistical model to a specific dataset. We developed PPCs for five population-level statistics of interest: within-population genetic variation, background linkage disequilibrium, number of ancestral populations, between-population genetic variation, and the downstream use of admixture parameters to correct for population structure in association studies. Using PPCs, we evaluated the quality of the admixture model fit to four qualitatively different

population genetic datasets: the population reference sample (POPRES) European individuals, the HapMap phase 3 individuals, continental Indians, and African American individuals. We found that the same model fitted to different genomic studies resulted in highly study-specific results when evaluated using PPCs, illustrating the utility of PPCs for model-based analyses in large genomic studies.

The posterior dispersion index. Finally we developed a new way to diagnose misfit of individual datapoints. Probabilistic modeling is cyclical: we specify a model, infer its posterior, and evaluate its performance. Evaluation drives the cycle, as we revise our model based on how it performs. This requires a metric. Traditionally, predictive accuracy prevails. Yet, predictive accuracy does not tell the whole story. Kucukelbir et al. (2017b) proposed to evaluate a model through posterior dispersion. The idea is to analyze how each datapoint fares in relation to posterior uncertainty around the hidden structure. This highlights datapoints the model struggles to explain and provides complimentary insight to datapoints with low predictive accuracy. We presented a family of posterior dispersion indices (PDI) that captured this idea. We showed how a PDI identifies patterns of model mismatch in three real data examples: voting preferences, supermarket shopping, and population genetics.

4.8 Applications and Dissemination

We practice the art of probabilistic modeling by implementing Box's loop in real-world applications. Over the past six years, we have developed new models for diverse applications, stretching probabilistic modeling in new ways. In particular, we developed new models for the following applications:

- text analysis (Rabinovich and Blei, 2014; Chaney et al., 2016; Rudolph et al., 2016, 2017; Rudolph and Blei, 2018; Gerow et al., 2018)
- relational data and networks (Gopalan et al., 2013; Kim et al., 2013; Schein et al., 2015, 2016; Linderman and Blei, 2018)
- computational neuroscience (Gershman et al., 2014; Manning et al., 2014; Linderman et al., 2017; Manning et al., 2018)
- econometrics (Ruiz et al., 2017; Athey et al., 2018)
- healthcare records (Perotte et al., 2015; Ranganath et al., 2016b; Ranganath and Blei, 2018)
- population genetics (Mimno et al., 2015; Gopalan et al., 2016; Tran and Blei, 2018)

- recommendation systems (Gopalan et al., 2014a,b, 2015; Chaney et al., 2015; Charlin et al., 2015; Liang et al., 2016)

These myriad applications stretched the methodology and pushed it in directions that were useful to real-world scientists and investigators.

Finally, we also disseminate our ideas in review papers. In particular, we wrote three papers that explain probabilistic models (Blei, 2014), variational inference (Blei et al., 2017), and data science (Blei and Smyth, 2017) to new audiences. These papers further help disseminate the ideas and accomplishments from our work on the PPAML program.

5 Conclusions

We described our successes in pushing forward the state of the art of probabilistic machine learning. Our contributions have changed the landscape of inference, tools, and real-world applications.

To conclude, we will identify some of the remaining challenges and limitations of the field.

- Probabilistic programming has focused on classical problems in machine learning, i.e., fitting a model to data and then using the model for prediction or interpretation. Probabilistic modeling is also important in the field of *causality*, which seeks to understand true causal mechanisms from observational data. Using probabilistic programming to implement and work with causal inference algorithms could have a significant impact on the nascent field of applied causality.

- Black box variational inference provides general inference methods for a wide class of models. However, there are many variants and innovations on BBVI and there is yet little understanding of in which setting each one works well. Outlining the performance of these variants and making concrete recommendations for new modelers would be an important technical contribution. It would greatly facilitate practical applications of black box inference.

- Model checking is a key activity and particularly so in a world where we have robust and usable probabilistic programming. However, model checking is still a domain-specific activity. Developing generic methods for assessing model fitness would be a major step for using Box's loop to solve real-world problems. Generic metrics could give us an understanding of ways that model's succeed and fail, and point to aspects of the model that the investigator should change.

- Probabilistic programming has focused on aspects like expressivity of the programming language and scalability of the inference method. But the goal is to make probabilistic machine learning

usable. To this end, user interfaces for probabilistic programming are a key area where we need new innovation. What is the best way to articulate domain assumptions? How do we translate them into a probabilistic program? Working with UI researchers on making probabilistic programming usable is an important direction for future research.

In summary, through inference, checking, robustness, and applications, we made significant progress on making probabilistic programming a reality. However, our work is not done. In the coming years, our vision is that probabilistic machine learning will become even more robust and usable.

6 References

Altosaar, J., Ranganath, R., and Blei, D. (2018). Proximity variational inference. In *Artificial Intelligence and Statistics*.

Athey, S., Blei, D., Donnelly, R., Ruiz, F., and Schmidt, T. (2018). Estimating heterogeneous consumer preferences for restaurants and travel time using mobile location data estimating heterogeneous consumer preferences for restaurants and travel time using mobile location data. *AEA Papers and Proceedings*, 108:64–67.

Blei, D. (2014). Build, compute, critique, repeat: Data analysis with latent variable models. *Annual Review of Statistics and Its Application*, 1:203–232.

Blei, D., Kucukelbir, A., and McAuliffe, J. (2017). Variational inference: A review for statisticians. *Journal of American Statistical Association*, 112(518):859–877.

Blei, D. M. and Smyth, P. (2017). Science and data science. *Proceedings of the National Academy of Sciences*, 114(33):8689–8692.

Chaney, A., Blei, D., and Eliassi-Rad, T. (2015). A probabilistic model for using social networks in personalized item recommendation. In *ACM Conference on Recommendation Systems*.

Chaney, A., Wallach, H., Connelly, M., and Blei, D. (2016). Detecting and characterizing events. In *Empirical Methods in Natural Language Processing*.

Charlin, L., Ranganath, R., McInerney, J., and Blei, D. (2015). Dynamic Poisson factorization. In *ACM Conference on Recommendation Systems*.

Dieng, A., Tran, D., Ranganath, R., Paisley, J., and Blei, D. (2017). Variational inference via χ upper bound minimization. In *Neural Information Processing Systems*.

Gerow, A., Hu, Y., Boyd-Graber, J., Blei, D., and Evans, J. (2018). Measuring discursive influence across scholarship. *Proceedings of the National Academy of Sciences*, 115(13):3308–3313.

Gershman, S., Blei, D., Norman, K., and Sederberg, P. (2014). Decomposing spatiotemporal brain patterns into topographic latent sources. *NeuroImage*.

Gershman, S., Frazier, P., and Blei, D. (2015). Distance dependent infinite latent feature models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2).

Gopalan, P. and Blei, D. (2013). Efficient discovery of overlapping communities in massive networks. *Proceedings of the National Academy of Sciences*, 110(36):14534–14539.

Gopalan, P., Charlin, L., and Blei, D. (2014a). Content-based recommendations with Poisson factorization. In *Neural Information Processing Systems*.

Gopalan, P., Hao, W., Blei, D., and Storey, J. (2016). Scaling probabilistic models of genetic variation to millions of humans. *Nature Genetics*, 48:1587–1590.

Gopalan, P., Hofman, J., and Blei, D. (2015). Scalable recommendation with hierarchical Poisson factorization. In *Uncertainty in Artificial Intelligence*, pages 326–335.

Gopalan, P., Ruiz, F., Ranganath, R., and Blei, D. (2014b). Bayesian nonparametric Poisson factorization for recommendation systems. In *Artificial Intelligence and Statistics*.

Gopalan, P., Wang, C., and Blei, D. (2013). Modeling overlapping communities with node popularities. In *Neural Information Processing Systems*.

Hoffman, M. and Blei, D. (2015). Structured stochastic variational inference. In *Artificial Intelligence and Statistics*.

Hoffman, M., Blei, D., Wang, C., and Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347.

Houlsby, N. and Blei, D. (2014). A filtering approach to stochastic variational inference. In *Neural Information Processing Systems*.

Kim, D., Gopalan, P., Blei, D., and Sudderth, E. (2013). Efficient online inference for bayesian nonparametric relational models. In *Neural Information Processing Systems*.

Kucukelbir, A. and Blei, D. (2015). Population empirical Bayes. In *Uncertainty in Artificial Intelligence*.

Kucukelbir, A., Ranganath, R., Gelman, A., and Blei, D. (2015). Automatic variational inference in Stan. In *Neural Information Processing Systems*.

Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., and Blei, D. (2017a). Automatic differentiation variational inference. *Journal of Machine Learning Research*, 18(14):1–45.

Kucukelbir, A., Wang, Y., and Blei, D. (2017b). Evaluating Bayesian models with posterior dispersion indices. In *International Conference on Machine Learning*, pages 1925–1934.

Liang, D., Charlin, L., McInerney, J., and Blei, D. (2016). Modeling user exposure in recommendation. In *International Conference on World Wide Web*.

Linderman, S. and Blei, D. (2018). A discussion of "Nonparametric Bayes modeling of populations of networks.". *Journal of the American Statistical Association*.

Linderman, S., Johnson, M., Miller, A., Adams, R., Blei, D., and Paninski, L. (2017). Bayesian learning and inference in recurrent switching linear dynamical systems. In *Artificial Intelligence and Statistics*.

Liu, L. and Blei, D. (2017). Zero-inflated exponential family embeddings. In *International Conference on Machine Learning*.

Mandt, S. and Blei, D. (2014). Smoothed gradients for stochastic variational inference. In *Neural Information Processing Systems*.

Mandt, S., Hoffman, M., and Blei, D. (2016a). A variational analysis of stochastic gradient algorithms. In *International Conference on Machine Learning*.

Mandt, S., Hoffman, M., and Blei, D. (2017). Stochastic gradient descent as approximate Bayesian inference. *Journal of Machine Learning Research*, 18:1–35.

Mandt, S., McInerney, J., Abrol, F., Ranganath, R., and Blei, D. (2016b). Variational tempering. In *Artificial Intelligence and Statistics*.

Manning, J., Ranganath, R., Norman, K., and Blei, D. (2014). Topographic factor analysis: A Bayesian model for inferring brain networks from neural data. *PLos One*, 9(5).

Manning, J., Zhu, X., Willke, T., Ranganath, R., Stachenfeld, K., Hasson, U., Blei, D., and Norman, K. (2018). A probabilistic approach to discovering dynamic full-brain functional connectivity patterns. *NeuroImage*, 180:243–252.

McInerney, J., Ranganath, R., and Blei, D. (2015). The population posterior and Bayesian modeling on streams. In *Neural Information Processing Systems*.

Mimno, D. and Blei, D. (2011). Bayesian checking for topic models. In *Empirical Methods in Natural Language Processing*, pages 227–237.

Mimno, D., Blei, D., and Engelhardt, B. (2015). Posterior predictive checks to quantify lack-of-fit in admixture models of latent population structure. *Proceedings of the National Academy of Sciences*.

Naesseth, C., Linderman, S., Ranganath, R., and Blei, D. (2018). Variational sequential Monte Carlo. In *Artificial Intelligence and Statistics*.

Naesseth, C., Ruiz, F., Linderman, S., and Blei, D. (2017). Reparameterization gradients through acceptance-rejection sampling algorithms. In *Artificial Intelligence and Statistics*.

Paisley, J., Wang, C., Blei, D., and Jordan, M. (2015). Nested hierarchical Dirichlet processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2).

Perotte, A., Ranganath, R., Hirsch, J., Blei, D., and Elhadad, N. (2015). Risk prediction for chronic kidney disease progression using heterogeneous electronic health record data and time series analysis. *Journal of the American Medical Informatics Association*, 22(4).

Polatkan, G., Zhou, M., Carin, L., Blei, D., and Daubechies, I. (2015). A Bayesian nonparametric approach to image super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2).

Rabinovich, M. and Blei, D. (2014). The inverse regression topic model. In *International Conference on Machine Learning*.

Ranganath, R., Altosaar, J., Tran, D., and Blei, D. (2016a). Operator variational inference. In *Neural Information Processing Systems*.

Ranganath, R. and Blei, D. (2018). Correlated random measures. *Journal of the American Statistical Association*, 113(521):417–430.

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

Ranganath, R. and Perotte, A. (2018). Multiple causal inference with latent confounding. *arXiv preprint arXiv:1805.08273*.

Ranganath, R., Perotte, A., Elhadad, N., and Blei, D. (2016b). Deep survival analysis. *Machine Learning for Health Care*.

Ranganath, R., Tang, L., Charlin, L., and Blei, D. (2015). Deep exponential families. In *Artificial Intelligence and Statistics*.

Ranganath, R., Tran, D., and Blei, D. (2016c). Hierarchical variational models. In *International Conference on Machine Learning*.

Ranganath, R., Wang, C., Blei, D., and Xing, E. (2013). An adaptive learning rate for stochastic variational inference. In Dasgupta, S. and McAllester, D., editors, *International Conference on Machine Learning*, pages 298–306.

Rudolph, M. and Blei, D. (2018). Dynamic embeddings for language evolution. In *International World Wide Web Conference*.

Rudolph, M., Ruiz, F., Athey, S., and Blei, D. (2017). Structured embedding models for grouped data. In *Neural Information Processing Systems*.

Rudolph, M., Ruiz, F., Mandt, S., and Blei, D. (2016). Exponential family embeddings. In *Neural Information Processing Systems*.

Ruiz, F., Titsias, M., and Blei, D. (2016a). The generalized reparameterization gradient. In *Neural Information Processing Systems*.

Ruiz, F., Titsias, M., and Blei, D. (2016b). Overdispersed black-box variational inference. In *Uncertainty in Artificial Intelligence*.

Ruiz, F., Titsias, M., Dieng, A., and Blei, D. (2018). Augment and reduce: Stochastic inference for large categorical distributions. In *International Conference on Machine Learning*.

Ruiz, F. J., Athey, S., and Blei, D. M. (2017). Shopper: A probabilistic model of consumer choice with substitutes and complements. *arXiv:1711.03560*.

Schein, A., Paisley, J., Blei, D., and Wallach, H. (2015). Bayesian Poisson tensor factorization for inferring multilateral relations from sparse dyadic event counts. In *Knowledge Discovery and Data Mining*, pages 1045–1054.

Schein, A., Zhou, M., Blei, D., and Wallach, H. (2016). Bayesian Poisson Tucker decomposition for learning the structure of international relations. In *International Conference on Machine Learning*.

Tran, D., Blei, D., and Airoldi, E. (2015). Copula variational inference. In *Neural Information Processing Systems*.

Tran, D. and Blei, D. M. (2018). Implicit causal models for genome-wide association studies. In *International Conference on Learning Representations*.

Tran, D., Hoffman, M., Sauraus, R., Brevdo, E., Murphy, K., and Blei, D. (2017a). Deep probabilistic programming. In *International Conference on Learning Representations*.

Tran, D., Ranganath, R., and Blei, D. (2016). The variational Gaussian process. In *International Conference on Learning Representations*.

Tran, D., Ranganath, R., and Blei, D. (2017b). Hierararchical implicit models and likelihood-free variational inference. In *Neural Information Processing Systems*.

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

Wang, C. and Blei, D. (2018). A general method for robust Bayesian modeling. *Bayesian Analysis*, 13(4):1163–1191.

Wang, Y. and Blei, D. (to appear). Frequentist consistency of variational Bayes. *Journal of the American Statistical Association*.

Wang, Y., Kucukelbir, A., and Blei, D. (2017). Robust probabilistic modeling with Bayesian data reweighting. In *International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 3646–3655, International Convention Centre, Sydney, Australia. PMLR.

7 List of Symbols, Abbreviations, and Acronyms

A&R	augment and reduce
ADVI	automatic differentiation variational inference
AVI	annealed variational inference
BBVI	black box variational inference
ChiVI	χ-divergence variational inference
CUBO	χ-upper bound
DARPA	Defense Advanced Research Project Agency
dd-IBP	distance-dependent Indian buffet process
DEF	deep exponential family
DRAW	deep recurrent attentive writer
ELBO	evidence lower bound
EM	expectation maximization
GP	Gaussian process
HIM	hierarchical implicit model
HVM	hierarchical variational model
IBP	Indian buffet process
KL divergence	Kullback-Leibler divergence
LDA	latent Dirichlet allocation
LFVI	likelihood-free variational inference
MCMC	Markov chain Monte Carlo
MVI	multicanonical variational inference
nCRP	nested Chinese restaurant process
nHDP	nested hierarchical Dirichlet process
OPVI	operator variational inference
PDI	posterior dispersion index
POPRES	population reference sample
PPAML	probabilistic programming for advanced machine learning
PPC	posterior predictive check

(continued on the next page)

pPCA	probabilistic principal component analysis
PPS	probabilistic programming system
PVI	proximity variational inference
SGD	stochastic gradient descent
SMC	sequential Monte Carlo
SNP	single nucleotide polymorphism
SVI	stochastic variational inference
VB	variational Bayes
VGP	variational Gaussian process
VI	variational inference
VSMC	variational sequential Monte Carlo