Build, Compute, Critique, Repeat: Data Analysis with Latent Variable Models

David M. Blei
Columbia University

Abstract

We survey how to use latent variable models to solve data analysis problems. A latent variable model is a probabilistic model of hidden and observed variables, where the hidden variables encode hidden patterns in our data. We uncover these patterns through the posterior, the conditional distribution of the hidden variables given the observations, which we use to explore, summarize, and form predictions about the data. Latent variable models have had a major impact on numerous applied fields—computational biology, natural language processing, social network analysis, computer vision, and many others. Here, we take the perspective—inspired by the work of George Box—that models are developed in an iterative process: we formulate a model, use it to analyze data, assess how the analysis succeeds and fails, revise the model, and repeat. With this view, we describe how new research in statistics and machine learning has transformed each of these essential activities. First, we discuss the types of patterns that latent variable models capture and how to formulate complex models by combining simpler ones. We describe probabilistic graphical models as a language for formulating latent variable models. Next we discuss algorithms for using models to analyze data, algorithms that approximate the conditional distribution of the hidden structure given the observations. We describe mean field variational inference, an approach that transforms this probabilistic computation into an optimization problem. Finally we discuss how we use our analyses to solve real-world problems: exploring the data, forming predictions about new data, and pointing us in the direction of improved models. We discuss predictive sample re-use and posterior predictive checks, two methods that step outside of the model's assumptions to scrutinize its explanation of the observed data.

Keywords: Latent variable models, Graphical models, variational inference, predictive sample reuse, posterior predictive checks

Note: This is a corrected version of an original article; please cite Blei (2014).

1 Introduction

This article is about the craft of building and using probability models to solve data-driven problems. We focus on *latent variable models*, models that assume that a complex observed data set exhibits simpler, but unobserved, patterns. Our goal is to uncover the patterns that are manifest in the data and use them to help solve the problem.

Here are example problems that latent variable models can help solve:

- You are a sociologist who has collected a social network and various attributes about each person. You use a latent variable model to discover the underlying communities in this population and to characterize the kinds of people that participate in each. The discovered communities help you understand the structure of the network and help predict "missing" edges, i.e., people that know each other but are not yet linked or people that would enjoy meeting each other.
- You are a historian who has collected a large electronic archive that spans hundreds of years. You would like to use this corpus to help form hypotheses about evolving themes and trends in language and philosophy. You use a latent variable model to discover the themes that are discussed in the documents, how those themes changed over time, and which publications seemed to have a large impact on shaping those themes. These inferences help guide your historical study of the archive, revealing new connections and patterns in the documents.
- You are a biologist with a large collection of genetic measurements from a population of living individuals around the globe. You use a latent variable model to discover the ancestral populations that mixed to form the observed data. From the structure you discovered with the model, you form hypotheses about human migration and evolution. You attempt to confirm the hypotheses with subsequent experiments and data collection.

These are just a few examples. Data analysis problems like these, and solutions using latent variable models, abound in many fields of science, government, and industry. In this paper, we will show how to use probabilistic models as a language for articulating assumptions about data, how to derive algorithms for computing under those assumptions, and how to solve data analysis problems through model-based probabilistic computations.

Box's loop. Our perspective is that building and using latent variable 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 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. Figure 1 illustrates this process.

We call this process "Box's loop". It is an adaptation—an attempt at revival, really—of the ideas of George Box and collaborators in their papers from the 1960s and 1970s (Box and Hunter, 1962, 1965; Box and Hill, 1967; Box, 1976, 1980). Box focused on the scientific method, understanding nature by iterative experimental design, data collection, model formulation, and model criticism. But his general approach just as easily applies to other applications of probabilistic modeling. It applies to engineering, where the goal is to use a model to build a system that performs a task, such as information retrieval or item recommendation. And it applies to exploratory data analysis, where the goal is to summarize, visualize, and hypothesize about observational data, i.e., data that we observe but that are not part of a designed experiment.

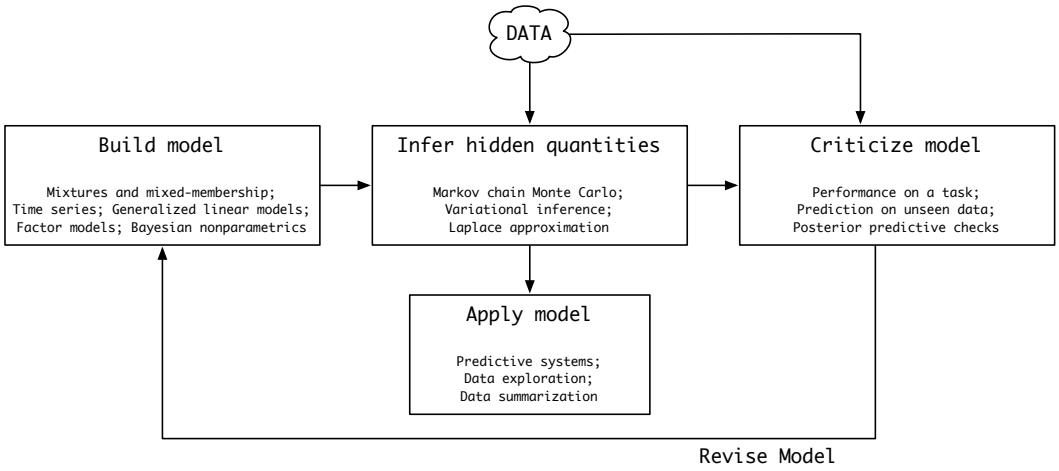

Figure 1: Building and computing with models is part of an iterative process for solving data analysis problems. This is Box's loop, an adaptation of the perspective of Box (1976).

Why revive this perspective now? The future of data analysis lies in close collaborations between domain experts and modelers. Box's loop cleanly separates the tasks of articulating domain assumptions into a probability model, conditioning on data and computing with that model, evaluating it in realistic settings, and using the evaluation to revise it. It is a powerful methodology for guiding collaborative efforts in solving data analysis problems.

As machine learning researchers and statisticians, our research goal is to make Box's loop easy to implement, and modern research has radically changed each component in the half-century since Box's inception. We have developed intuitive grammars for building models, scalable algorithms for computing with a wide variety of models, and general methods for understanding the performance of a model to guide its revision. This paper gives a curated view of the state-of-the-art research for implementing Box's loop.

In the first step of the loop, we build (or revise) a probability model. *Probabilistic graphical models* (Pearl, 1988; Dawid and Lauritzen, 1993; Jordan, 2004) is a field of research that connects graph theory to probability theory, and provides an elegant language for building models. With graphical models, we can clearly articulate what kinds of hidden structures are governing the data and construct complex models from simpler components—like clusters, sequences, hierarchies, and others—to tailor our models to the data at hand. This language gives us a palette with which to posit and revise our models.

The observed data enters the picture in the second step of Box's loop. Here we compute the posterior distribution, the conditional distribution of the hidden patterns given the obervations, to understand how the hidden structures we assumed are manifested in the data.[1] Most useful models are difficult to compute with, however, and researchers have developed powerful *approximate posterior inference* algorithms for approximating these conditionals. Techniques like Markov chain Monte Carlo (MCMC) (Metropolis et al., 1953; Hastings, 1970; Geman and Geman, 1984) and variational inference (Jordan et al., 1999; Wainwright and Jordan, 2008) make it possible for us to examine large

[1] In a way, we take a Bayesian perspective because we treat all hidden quantities as random variables and investigate them through their conditional distribution given observations. However, we prefer the more general langauge of latent variables, which can be either parameters to the whole data set or local hidden structure to individual data points (or something in between). Further, in performing model criticism we will step out of the Bayesian framework to ask whether the model we assumed has good properties in the sampling sense.

data sets with sophisticated statistical models. Moreover, these algorithms are modular—recurring components in a graphical model lead to recurring subroutines in their corresponding inference algorithms. This has led to more recent work in efficient generic algorithms, which can be easily applied to a wide class of models (Gelfand and Smith, 1990; Bishop et al., 2003).

Finally we close the loop, studying how our models succeed and fail to guide the process of revision. Here again is an opportunity for a revival. With new methods for quickly building and computing with sophisticated models, we can make better use of techniques like predictive sample reuse (Geisser, 1975) and posterior predictive checks (Box, 1980; Rubin, 1984; Gelman et al., 1996; Gelman and Shalizi, 2012). These are general techniques for assessing model fitness, contrasting the predictions that a model makes against the observed data. Understanding a model's performance in the ways the matter to the task at hand—an activity called *model criticism*—is essential to solving modern data analysis problems.

This paper. We will describe each component of Box's loop in turn. In Section 2 we describe probabilistic modeling as a language for expressing assumptions about data, and we give the graphical models notation as a convenient visual representation of structured probability distributions. In Section 3, we discuss several simple examples of latent variable models, and describe how they can be combined and expanded when developing new models for new problems. In Section 4, we describe mean field variational inference, a method of approximate posterior inference that can be easily applied to a wide class of models and handles large data sets. Finally, in Section 5, we describe how to criticize and assess the fitness of a model, using predictive likelihood and posterior predictive checks.

Again, this is a curated view. For other surveys of latent variable models, see Skrondal and Rabe-Hesketh (2007), Ghahramani (2012), and Bishop (2013). For more complete treatments of probabilistic modeling, see books like Bishop (2006) and Murphy (2013). Finally, for another perspective on iterative model building, see Krnjajić et al. (2008).

With the ideas presented here, our hope is that the reader can begin to iteratively build sophisticated methods to solve real-world problems. We emphasize, however, that data analysis with probability models is a craft. Here we will survey some its tools, but the reader will only master them as with any other craft—practice.

2 Latent Variable Models

When we build a latent variable model, we imagine what kinds of hidden quantities might be used to describe the type of data we are interested in and we encode that relationship in a joint probability distribution of hidden and observed random variables. Then, given an observed data set, we uncover the particular hidden quantities that describe it through the conditional distribution of the hidden variables given the observations, which is also called the posterior. Further, we use the posterior to form the predictive distribution, the distribution over future data that the observations and the model imply.

Consider a mixture model, one of the simplest latent variable models. A mixture model assumes that the data are clustered and that each data point is drawn from a distribution associated with its assigned cluster. The hidden variables of the model are the the cluster assignments and parameters to the per-cluster distributions. Given observed data, the mixture model posterior is a conditional distribution over clusterings and parameters. This conditional identifies a likely grouping of the data and the characteristics of each group.

More formally, a model consists of three types of variables. The first type is an observation, which represents a data point. We denote N data points as $x = x_{1:N}$. The second type is a hidden variable. These encode hidden quantities (like cluster memberships and cluster means) that are used to govern the distribution of the observations. We denote M hidden variables as $h = h_{1:M}$. The third type is a hyperparameter. These are fixed non-random quantities, which we denote by η.[2]

A model is a joint distribution of x and h, $p(h, x \mid \eta) = p(h \mid \eta)p(x \mid h)$, which formally describes how the hidden variables and observations interact in a probability distribution. After we observe the data, we are interested in the conditional distribution of the hidden variables, $p(h \mid x, \eta) \propto p(h, x \mid \eta)$. This also leads to the predictive distribution, $p(x_{\text{new}} \mid x) = \int p(x_{\text{new}} \mid h)p(h \mid x, \eta)dh$.

Continuing with the example, a Gaussian mixture model has K mixture components $\mu_{1:K}$, each of which is the mean of a Gaussian distribution, and a set of mixture proportions θ, a non-negative K-vector that sums to one. Data arise by first choosing a component assignment z_n (an index from 1 to K) from the mixture proportions and then drawing the data point x_n from the corresponding Gaussian, $x_i \mid z_i \sim \mathcal{N}(\mu_{z_i}, 1)$. The hidden variables for this model are the mixture assignments for each data point $z_{1:N}$, the set of mixture component means $\mu_{1:K}$, and the mixture proportions θ. To fully specify the model, we place distributions, called priors, on the mixture components (e.g., a Gaussian) and the mixture proportions (e.g., a Dirichlet). The fixed hyperparameters η are the parameters to these distributions.

Suppose we observe a data set $x_{1:N}$ of real values. We analyze this data with a Gaussian mixture by estimating $p(\mu_{1:K}, \theta, z_{1:N} \mid x_{1:N})$, the posterior distribution of the mixture components, the mixture proportions, and the way the data are clustered. (See Section 4.) This posterior reveals hidden structure in our data—it clusters the data points into K groups and describes the location (i.e., the mean) of each group.[3] The predictive distribution, derived from the posterior, gives the distribution of the next data point. Figure 2 gives an example.

We will describe three ways of specifying a model: its generative probabilistic process, its joint distribution, and its directed graphical model.

The generative probabilistic process. The generative probabilistic process describes how data arises from the model.[4] We have already alluded to this above. The generative process for the Gaussian mixture model is:

1. Draw mixture proportions $\theta \sim \text{Dirichlet}(\alpha)$.
2. For each mixture component k, draw $\mu_k \sim \mathcal{N}(0, \sigma_0^2)$.
3. For each data point i:
   (a) Draw mixture assignment $z_i \mid \theta \sim \text{Discrete}(\theta)$.
   (b) Draw data point $x_i \mid z_i, \mu \sim \mathcal{N}(\mu_{z_i}, 1)$.

The posterior distribution can be seen as "reversing" this process: given data, what is the distribution of the hidden structure that likely generated it?

[2] In this paper we focus on models of hidden and observed random variables, and always assume that the hyperparameters are fixed. Estimating hyperparameters from data is the important, and difficult problem of *empirical Bayes* (Efron and Morris, 1973; Robbins, 1980; Morris, 1983; Efron, 2013)

[3] But note that even after computing the posterior, we do not yet know that a Gaussian mixture is a good model for our data. No matter which data we observe, we can obtain a posterior over the hidden variables, i.e., a clustering. (And, further, note the number of mixture components is fixed—the number we chose may not be appropriate.) After formulating a model and inferring the posterior, Section 5 discusses how to check whether it adequately explains the data.

[4] Recall here that the Dirichlet distribution is a distribution on the simplex, non-negative vectors that sum to one. Thus a draw from a Dirichlet can be used as a parameter to a multinomial downstream in the model. Further note that we call a single draw from a multinomial (or a multinomial with $N = 1$) a "discrete" distribution.

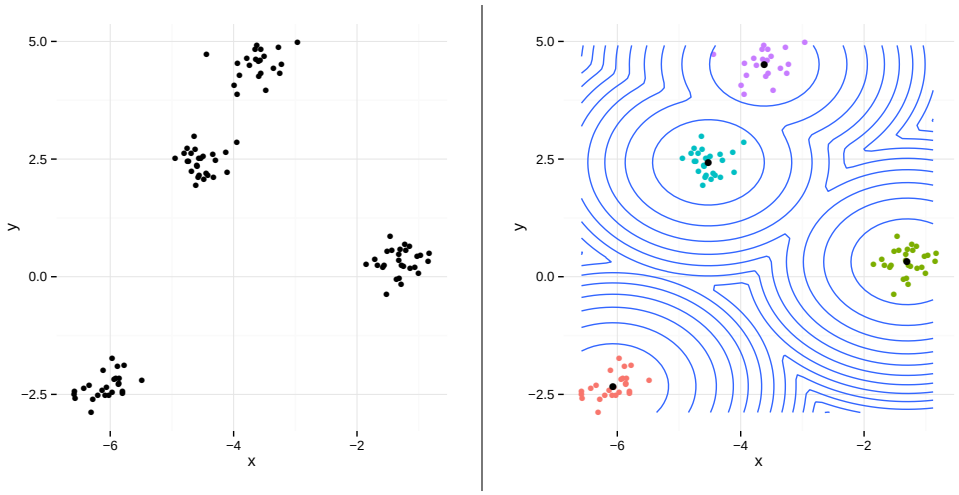

Figure 2: An example with a mixture of Gaussians. On the left is a data set of 100 points. On the right is the same data set, now visualized with the hidden structure that we derived from approximating the posterior for a mixture of four Gaussians. Each data point is colored with its post likely assigned cluster, the most likely cluster means are marked in black, and the contours give the posterior predictive distribution of the next data point.

This process makes the hyperparameters explicit; they are the variance of the prior on the mixture components σ_0^2 and the Dirichlet parameters α. The process also helps us identify *local* and *global* hidden variables. The global variables are the mixture proportions θ and the mixture components $\mu = \mu_{1:K}$. These are variables that describe hidden structure which is shared for the entire data set. The local variables are the mixture assignments; each assignment z_i only helps govern the distribution of the ith observation. This distinction will be important when we discuss algorithms for approximating the posterior.

The joint distribution. The traditional way of representing a model is with the factored joint distribution of its hidden and observed variables. This factorization comes directly from the generative process. For the Gaussian mixture, the joint is

$$p(\theta, \mu, z, x \mid \sigma_0^2, \alpha) = p(\theta \mid \alpha) \prod_{k=1}^{K} p(\mu_k \mid \sigma_0^2) \prod_{i=1}^{N} (p(z_i \mid \theta) p(x_i \mid z_i, \mu, z_i)) . \tag{1}$$

For each term, we substitute the appropriate density function. In this case, $p(\theta \mid \alpha)$ is the Dirichlet density, $p(\mu_k \mid \sigma_0)$ is the Gaussian density, $p(z_i \mid \theta) = \theta_{z_i}$ is a discrete distribution, and $p(x_i \mid z_i, \mu)$ is a Gaussian density centered at the z_ith mean. Notice the distinction between local and global variables: local variables have terms inside the product over N data points, whereas global variables have terms outside of this product.

The joint distribution lets us calculate the posterior. In the Gaussian mixture model, the posterior is

$$p(\theta, \mu, z \mid x, \sigma_0, \alpha) = \frac{p(\theta, \mu, z, x \mid \sigma_0, \alpha)}{p(x \mid \sigma_0, \alpha)}. \tag{2}$$

We use the posterior to examine the particular hidden structure that is manifest in the observed data. We also use the posterior (over the global variables) to form the posterior predictive distribution of

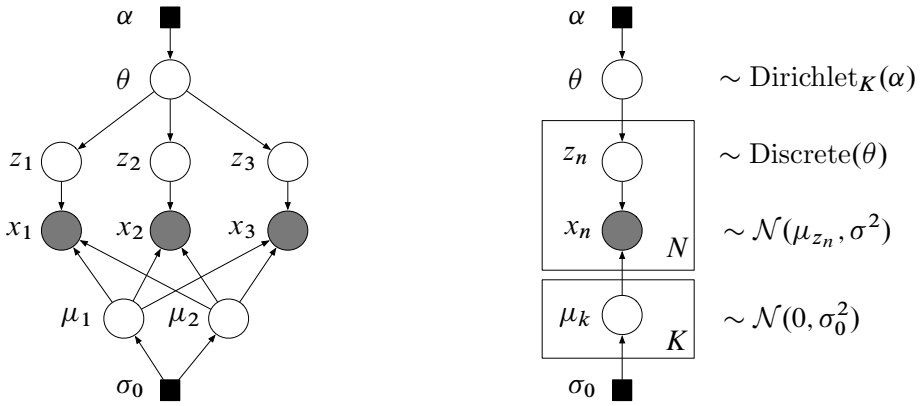

Figure 3: (Left) A graphical model for a mixture of two Gaussians. There are three data points. (Right) A graphical model for a mixture of K Gaussians with N data points.

future data. For the mixture, the predictive distribution is

$$p(x_{\text{new}} \mid x, \sigma_0, \alpha) = \int \left(\sum_{z_{\text{new}}} p(z_{\text{new}} \mid \theta) p(x_{\text{new}} \mid z_{\text{new}}, \mu) \right) p(\mu, \theta \mid x, \sigma_0, \alpha) d\mu d\theta. \tag{3}$$

In Section 5, we discuss how the predictive distribution is important for checking and criticizing latent variable models.

The denominator of Equation 2 is the marginal probability of the data, also called the "evidence", which is found by marginalizing out the hidden variables from the joint. For many interesting models the evidence is difficult to efficiently compute, and developing approximations to the posterior has thus been a focus of modern Bayesian statistics. In Section 4, we will describe variational inference, an technique for approximating the posterior that emerged from the statistical machine learning community.

The graphical model. Our final way of viewing a latent variable model is as a probabilistic graphical model, another representation that derives from the generative process. The generative process indicates a pattern of dependence among the random variables. In the mixture model's process, the mixture components μ_k and mixture proportions θ do not depend on any hidden variables; they are generated from distributions parameterized by fixed hyperparameters (steps 1 and 2). The mixture assignment z_i depends on the mixture proportions θ, which is the parameter to its distribution (step 3a). The observation x_i depends on the mixture components μ and mixture assignment z_i (step 3b). We can encode these dependencies with a graph in which nodes represent random variables and edges denote dependence between them. This is a graphical model.[5] It illustrates the structure of the factorized joint distribution and the flow of the generative process.

Figure 3 (left) illustrates a graphical model for three data points drawn from a mixture of two Gaussians. The figure illustrates some additional notation beyond nodes and edges: shaded nodes are observed variables; unshaded nodes are hidden variables; and fixed hyperparameters (like the Dirichlet parameters) are black square boxes. Furthermore, this is an "unpacked" model, where each data point is given its own substructure in the graph. We can summarize repeated components of a model with plates, rectangles that encase a substructure to denote replication. Figure 3 (right) is a more succinct graphical model for N data points modeled with a mixture of K Gaussians.

[5]The formal semantics of graphical models asserts that there is possible dependence between random variables connected by an edge. Here, as a practical matter, we can read the model as asserting dependence.

The field of graphical models is deep. It connects the topological structure of a graph with elegant algorithms for computing various quantities about the joint distributions that the graph describes. Formally, a graphical model represents the family of distributions that respects the independencies it implies. (These include the basic independencies described above, along with others that derive from graph theoretic calculations.) We will not discuss graphical models in depth. For good references see the books of Pearl (1988); Jordan (1999); Bishop (2006); Koller and Friedman (2009), and Murphy (2013).

We will use graphical models as a convenient visual language for expressing how hidden structure interacts with observations. In our applied research, we have found that graphical models are useful for domain experts (like scientists) to build and discuss models with statisticians and computer scientists.

3 Example Models

We have described the basic idea behind latent variable models and a simple example, the Gaussian mixture model. In this section we describe some of the commonly recurring components in latent variable models—mixed-membership, linear factors, matrix factors, and time-series—and point to some of their applications.

Many of the models we describe below were discovered (and sometimes rediscovered) in specific research communities. They were often bundled with a particular algorithm for carrying out posterior inference and sometimes developed without a probabilistic modeling perspective. Here we present these models probabilistically and treat them more generally than as standalone solutions. We separate the independence assumptions they make from their distributional assumptions, and we postpone the discussion of computation to the generic algorithms in Section 4.

The models below are useful in and of themselves—we selected a set of models that have useful applications and a history in the statistics and machine learning literature—but we hope to deemphasize their role in a "cookbook" of methods. Rather, we highlight their use as components in more complex models for more complex problems and data. This is the advantage of the probabilistic modeling framework.

3.1 Linear Factor Models

Linear factor models embed high-dimensional observed data in a low-dimensional space. These models have been a mainstay in Statistics for nearly a century—principal component analysis (Pearson, 1901; Hotelling, 1933), factor analysis (Thurstone, 1931, 1938; Thomson, 1939), and canonical correlation analysis (Hotelling, 1936) can all be interpreted this way, although the probabilistic perspective on them is more recent (Roweis, 1998; Tipping and Bishop, 1999; Collins et al., 2002). Factor models are important as a component in more complicated models (like the Kalman filter, see below) and have generally spawned many extensions (Bartholomew et al., 2011).

In a factor model, there is a set of hidden components (as in a mixture) and each data point is associated with a hidden vector of "weights", with one weight for each component. The data arise from a distribution whose parameters combines the global components with the per-data point weights. Conditioned on data, the posterior finds the global components that describe the data set and the local weights for each data point. The components capture general patterns in the data; the weights capture how each data point exhibits those patterns and serve as a low-dimensional embedding of the high-dimensional data.

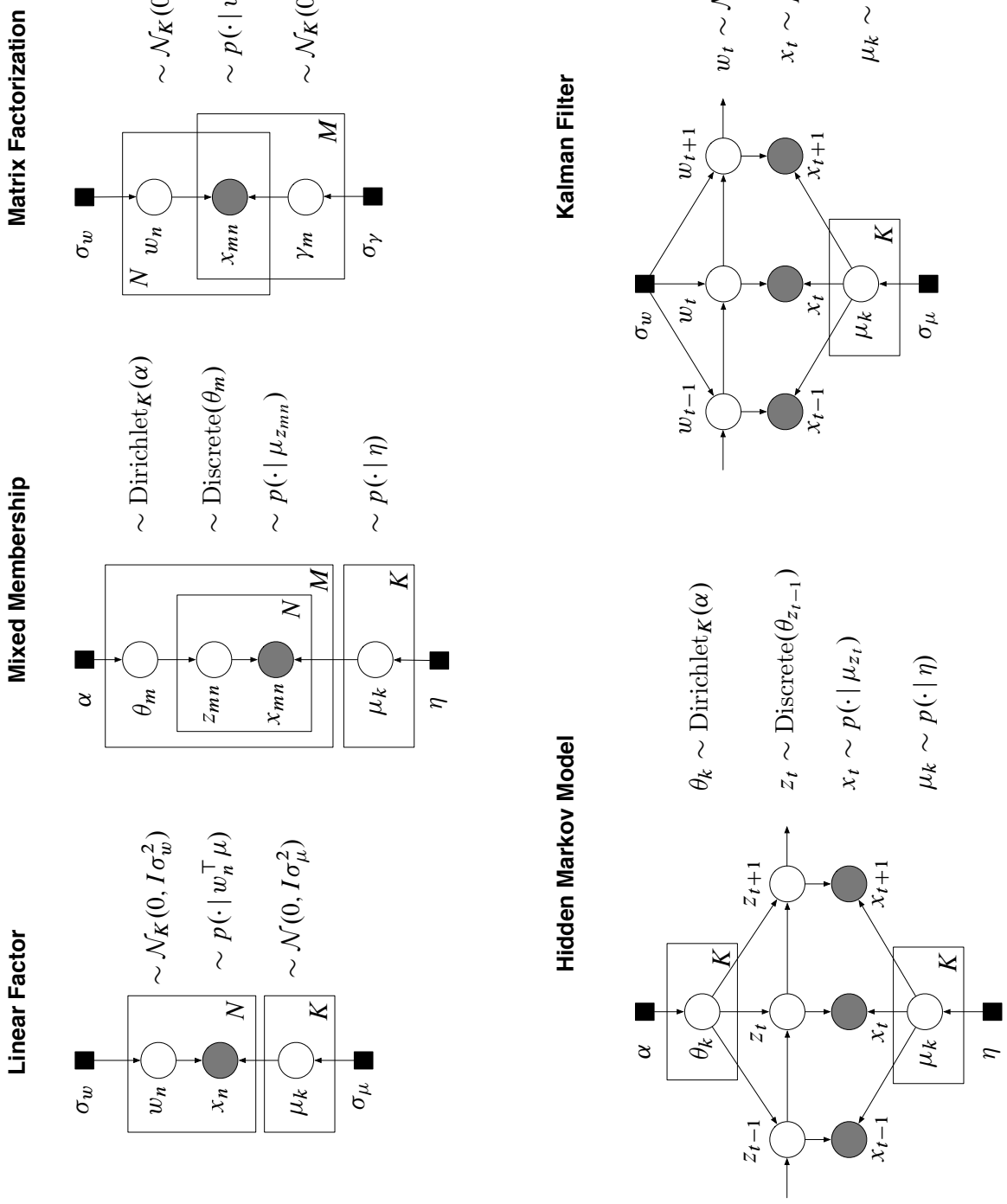

Figure 4: Graphical models for the model components described in Section 3.

Figure 4 (top left) illustrates the graphical model. Notice that the independence assumptions about the hidden and observed variables (which come from the structure of the graphical model) are similar to those from the mixture model (Figure 3). The linear factor model contains K components $\mu_{1:K}$, which are organized into a matrix μ. For each data point n, we draw a weight vector w_n and then draw the data point from a distribution parameterized by $w_n^\top \mu$. Traditionally, the data are drawn from a Gaussian (Tipping and Bishop, 1999), but extensions to linear factors have considered exponential families where $w_n^\top \mu$ is the natural parameter (Collins et al., 2002; Mohamed et al., 2008).

3.2 Mixed-Membership Models

Mixed-membership models are used for the unsupervised analyses of grouped data, multiple sets of observations that we assume are statistically related to each other. In a mixed-membership model, each group is drawn from a mixture, where the mixture proportions are unique to the group and the mixture components are shared across groups.

Consider the following examples: Text data are collections of documents, each containing a set of observed words; Genetic data are collections of people, each containing observed alleles at various locations along the genome; Survey data are completed surveys, each a collection of answers by a single respondent; Social networks are collections of people, each containing a set of connections to others. In these settings, mixed-membership models assume that there is a single set of coherent patterns underlying the data—themes in text (Blei et al., 2003), populations in genetic data (Pritchard et al., 2000), types of respondents in survey data (Erosheva et al., 2007), and communities in networks (Airoldi et al., 2008)—but that each group exhibits a different subset of those patterns and to different degrees.

Mixed-membership models posit a set of global mixture components. Each data group arises by first choosing a set of mixture proportions and then, for each observation in the group, choosing a mixture assignment from the per-group proportions and the data from its corresponding component. The groups share the same set of components but each exhibits them with different proportion. Thus the mixed-membership posterior uncovers the recurring patterns in the data, and the proportion with which they occur in each group. This is in contrast to simple mixture models, where each data point, i.e., group, is associated with a single mixture component. Mixed-membership models give better predictions than mixtures and more interesting exploratory structure.

Figure 4 (top center) illustrates the graphical model for a mixed-membership model. There are M groups of N data points; the observation x_{mn} is the nth observation in the mth group.[6] The hidden variable μ_k is an appropriate parameter to a distribution of x_{mn} (e.g., if the observation is discrete then μ_k is a discrete distribution) and $p(\cdot \mid \eta)$ is an appropriate prior with hyperparameter η. The other hidden variables are per-group mixture proportions θ_m, a point on the simplex (i.e., K-vectors that are positive and sum to one), and per-observation mixture assignments z_{mn}, which indexes one of the components. Given grouped data, the posterior finds the mixture components that describe the whole data set and mixture proportions for each group. Note that the mixture graphical model of Figure 3 is a component of this more complicated model.

For example, mixed-membership models of documents—where each document is a group of observed words—are called topic models (Blei et al., 2003; Erosheva et al., 2004; Steyvers and Griffiths, 2006; Blei, 2012). In a topic model, the data generating components are probability distributions over a vocabulary and each document is modeled as a mixture of these distributions. Given a collection, the posterior components put their probability mass on terms that are associated under a single theme—this connects to words that tend to co-occur—and thus are called "topics". The posterior proportions indicate how each document exhibits those topics, e.g., one document might be

[6] Each group need not have the same number of observations. But this makes the notation cleaner.

Figure 5: Topics found in a corpus of 1.8M articles from the New York Times. This figure is from Hoffman et al. (2013).

about "sports" and "health" while another is about "sports" and "business". (Again, we emphasize that the topics too are uncovered by the posterior.) Figure 5 shows the topics from a topic model fit to 1.8M articles from the New York Times. This figure was made by estimating the posterior (see Section 4) and then plotting the most frequent words from each topic.

3.3 Matrix Factorization Models

Many data sets are organized into a matrix, where each observation is indexed by a row and a column. Our goal may be to understand something about the rows and columns, or to predict the values of unobserved cells. For example, in the Netflix challenge problem, we observe a matrix where rows are users, columns are movies, and each cell x_{nm} (if observed) is how user n rated movie m. The goal is to predict which movies user n will like that she has not yet seen. Another example is political roll call data, how lawmakers vote on proposed bills. In the roll-call matrix, rows are lawmakers, columns are bills, and each cell is how lawmaker n voted on bill m. Here the main statistical problem is for exploration. Where do the lawmakers sit on the political spectrum? Which ones are most conservative? Which ones are most liberal?

A matrix factorization model uses hidden variables to embed both the rows and columns in a low-dimensional space. Each observed cell is modeled by a distribution whose parameters are a linear combination of the row embedding and column embedding, e.g., cells in a single row share the same row embedding but are governed by different column embeddings. Conditioned on an observed matrix, the posterior distribution gives low-dimensional representations of its rows and columns.

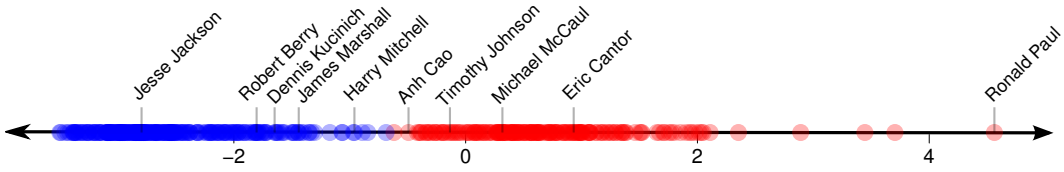

Figure 6: Ideal point models separate Republicans (red) from Democrats (blue) and try to capture the degree to which each lawmaker is liberal or conservative. (The figure is thanks to Sean Gerrish.)

These can be used to form predictions about unseen cells and can be used to explore hidden structure in the data.

For example, in the movie recommendation problem we use the model to cast each user and movie in a low-dimensional space. For an unseen cell (a particular user who has not watched a particular movie), our prediction of the rating depends on a linear combination of the user's embedding and movie's embedding. We can also use these inferred representations to find groups of users that have similar tastes and groups of movies that are enjoyed by the same kinds of users.

Figure 4 (top right) illustrates the graphical model. It is closely related to a linear factor model, except that the particular hidden variables that determine the distribution for each cell depend on its row and column. The overlapping plates show how the observations at the nth row share its embedding w_n, but use a different variables γ_m for each column. Similarly, the observations in the mth column share its embedding γ_m, but use different variables w_n for each row. Casting matrix factorization as a general probabilistic model is called probabilistic matrix factorization (Salakhutdinov and Mnih, 2008).

Matrix factorization is widely used. In quantitative political science, "ideal point models" are a one-dimensional matrix factorization of legislative roll-call data (Clinton et al., 2004). Figure 6 illustrates the one-dimensional embeddings of lawmakers in the 115th U.S. Congress. The model has captured the divide between Democrats and Republicans (denoted by color), and identified a finer-grained political spectrum from liberal to conservative. Similar models are used in educational testing scenarios, where rows are testers and columns are questions on a test (Baker, 1992). Finally, extensions of matrix factorization models were important in winning the Netflix challenge (Koren et al., 2009). These researchers essentially took into account Box's loop—they fit simple matrix models first and then, as a result of insightful model criticism, embellished the basic model to consider important elements like time and (latent) movie popularity.

3.4 Time-Series Models

Many observations are sequential: they are indexed by time or position and we want to take this structure into account when making inferences. For example, genetic data are indexed by location on the chromosome; data from radar observations is indexed by time; words in a sentence are indexed by their position. There are two canonical and closely related latent variable models of time series: the hidden Markov model and the Kalman filter. Each has had important scientific and engineering applications and each demonstrates how we can use simple latent variable models to construct more complex ones.

Hidden Markov models. In the hidden Markov model (HMM) each observation of the time-series is drawn from an unobserved mixture component, where that component is drawn conditional

on the previous observation's mixture component. The posterior is similar to a mixture model, but one where the model assumes a Markovian structure on the sequence of mixture assignments. HMMs have been successfully used in many applications, notably in speech recognition (Rabiner, 1989) and computational biology (Durbin et al., 1998). For example, in speech recognition, the hidden states represent words from a vocabulary, the Markov process is a model of natural language (it may be more complex than the simple first order model described above) and the data are the observed audio signal. Posterior inference of the hidden states gives us an estimate of what is being said in the audio signal.

Figure 4 illustrates the graphical model. Note how modeling a time series translates to a linked chain in the graph. The global hidden variables are the mixture components $\mu_{1:K}$ (as in a mixture model or mixed-membership model) and the transition probabilities $\theta_{1:K}$. The transition probabilities give K conditional distributions over the next component, given the value of the previous one.

The Kalman filter. While the HMM is a time-series adaptation of the mixture model, the *Kalman filter* is a time-series adaptation of a linear (Gaussian) factor model (Kalman, 1960). In particular, we draw the row embeddings from a state space model, from a Gaussian whose mean is the previous position's embedding. See West and Harrison (1997) for a general probabilistic perspective on continuous time-series.

Kalman filters are influential in radar tracking applications, where w_t represents the latent position of an object in space, the state-space model captures the assumed process by which the position moves—which may be more complex than the simple process laid out above—and the observations represent blips on a radar screen that are corrupted by noise (Bar-Shalom et al., 2004). Inferences from this model help track an objects true position and predict its next position.

Figure 4 illustrates the graphical model. Though the underlying distributions are different—in the Kalman filter the hidden chain of variables is a sequence of continuous variables, rather than discrete ones—the structure of the graphical model is nearly the same as for the hidden Markov model. This similar structure reveals close connections between algorithms developed independently for the Kalman filter and hidden Markov model. Finding such connections, and developing diverse applications with new models, is one of the advantages of the graphical models formalism.

3.5 The Craft of Latent Variable Modeling

We have described a selection of latent variable models, each building on simpler models. However, we emphasize that our goal is not to deliver a catalog of models for use in a variety of data analysis problems. In fact, we omitted some important components, like Bayesian nonparametric models (Ferguson, 1973; Antoniak, 1974; Teh and Jordan, 2008; Hjort et al., 2010; Gershman and Blei, 2012) that let the data determine the structure of the latent variables, random effects models (Gelman et al., 1995) that allow data to depend on hidden covarates, and hierarchical models (Gelman and Hill, 2007) that allow data to exhibit complex and overlapping groups. Rather, we want to demonstrate how to use probability modeling as a language of assumptions for tailoring latent variable models to each data analysis challenge.

There are several ways to adapt and develop new latent variable models. One way to adapt a model is to change the data generating distribution of an existing model. At the bottom of each probabilistic process is a step that generates an observation conditioned on the latent structure. For example, in the mixed-membership model this is a distribution conditioned on a component; in the factor model, it is a distribution conditioned on a linear combination of weights and factors. Depending on the type of observation at hand, changing this distribution can lead to new latent variable models. We might use ordinal distributions for ordered data, Gamma distributions and truncated Gaussians for

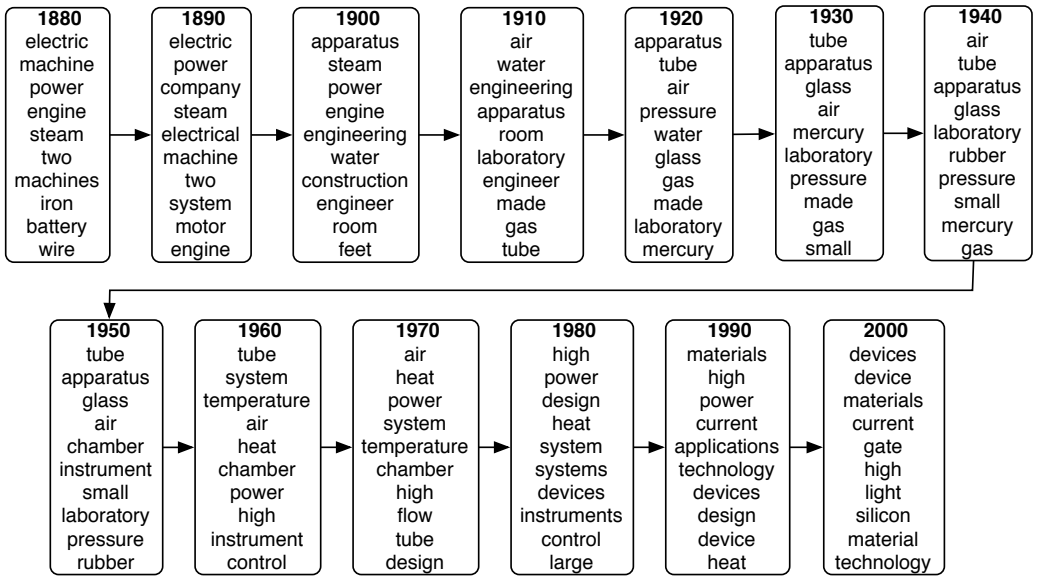

Figure 7: A dynamic topic found with a dynamic topic model applied to a large corpus from *Science* magazine. The model has captured the idea of "technology" and how it has changed throughout the course of the collection. This figure is from Blei (2012).

positive data, or discrete distributions for categorical data. We can even use conditional models, like generalized linear models (Nelder and Wedderburn, 1972; McCullagh and Nelder, 1989), that use observed (but not modeled) covariates to help describe the data distribution.

Another way to develop new models is to change the distributional assumptions on the latent variables. We might replace a Gaussian with a Gamma distribution to enforce positivity. This fundamentally changes models like Gaussian matrix factorization to a form of nonnegative matrix factorization, a technique that is important in computer vision problems (Lee and Seung, 1999). Or, we might replace simple distributions with more complex distributions that themselves have latent structure. For example, the idea behind "spike and slab" modeling (Ishwaran and Rao, 2005) is to generate a vector (such as the hidden weights in a factor model) in a two stage process: first generate a bank of binary variables that indicate which factors are relevant to a data point; then generate the weights of those factors from an appropriate distribution. This gives a sparse hidden vector in a way that a simple distribution cannot.

Finally, we can mix and match components from different models to create wholly new techniques. For example a *dynamic topic model* captures documents that are organized in time (Blei and Lafferty, 2006). At each time point, the documents are modeled with a mixed-membership model, but the components are connected sequentially from time-point to time-point. Figure 7 illustrates one of the topic sequences found with a dynamic topic model fit to *Science* magazine. Incorporating time gives a much richer latent structure than we find with the model of Figure 5. Dynamic topic models take elements from mixed-membership models and Kalman filters to solve a unique problem—it finds the themes that underlie the collection and uncovers how those themes change smoothly through time.

As another example, McAuliffe et al. (2004) and Siepel and Haussler (2004) combined tree-based models and hidden Markov models to simultaneously analyze genetic data from a collection of species organized in a phylogenetic tree. These models can predict the exact locations of protein-coding genes common to all the species analyzed.

We emphasize that we have only scratched the surface of the kinds of models that we can build. Models can be formed from myriad components—binary vectors, time-series, hierarchies, mixtures—which can be composed and connected in many ways. With practice and experience, data analysts can understand the kind of latent structure they wants to uncover and the generative assumptions that can capture it.

4 Posterior Inference with Mean Field Variational Methods

We have described the probabilistic modeling formalism, discussed several latent variable models, and shown how they can be combined and expanded to create new models. We now turn to the nuts and bolts of how to use models with observed data—how to uncover hidden structure and how to form predictions about new data. Both of these problems hinge on computing the posterior distribution, the conditional distribution of the hidden variables given the observations. Computing or approximating the posterior is the central algorithmic problem in probabilistic modeling. (It is the second step of Box's loop in Figure 1.) This is the problem of posterior inference.

For some basic models, we can compute the posterior exactly, but for most interesting models we must approximate it, and researchers in Bayesian statistics and machine learning have pioneered many methods for approximate inference. The most widely used methods include Laplace approximations and Monte Carlo Markov chain (MCMC) sampling. Laplace approximations (Tierney et al., 1989) represent the posterior as a Gaussian, derived from a Taylor approximation. Laplace approximations work well in some simple models, but are difficult to use in high-dimensional settings (MacKay, 2003). See Smola et al. (2003) and Rue et al. (2009) for recent innovations.

MCMC sampling methods (Robert and Casella, 2004) include fundamental algorithms like Metropolis-Hastings (Metropolis et al., 1953; Hastings, 1970) and Gibbs sampling (Geman and Geman, 1984; Gelfand and Smith, 1990). This class of methods form a Markov chain over the hidden variables whose stationary distribution is the posterior of interest. The algorithm simulates the chain, drawing consecutive samples from its transition distribution, to collect independent samples from its stationary distribution. It approximates the posterior with the empirical distribution over those samples. MCMC is a workhorse of modern Bayesian statistics.

Approximate posterior inference is an active field and it is not our purpose to survey its many branches. Rather, we will discuss a particular strategy, mean field variational inference. Variational inference is a deterministic alternative to MCMC, where sampling is replaced by optimization (Jordan et al., 1999; Wainwright and Jordan, 2008). In practice, variational inference tends to be faster than sampling methods, especially with large and high-dimensional data sets, but it has been less vigorously studied in the statistics literature. Using stochastic optimization (Robbins and Monro, 1951), variational inference scales to massive data sets (Hoffman et al., 2013).

In this section, we first describe conditionally conjugate models, a large subclass of latent variable models. Then we present the simplest variational inference algorithm, mean field inference, for this subclass. This gives a generic algorithm for approximating the posterior distribution for many models.[7]

[7] This section is more mathematically dense and abstract than the rest of the paper. Readers may want to skip it if they are comfortable approximating a posterior with a different method (and not interested in reading about variational inference) or know someone who will implement posterior inference programs for their models and do not need to absorb the details.

4.1 Conditionally Conjugate Models

We describe *conditionally conjugate models*, the model subclass for which we will present mean-field variational inference. Let $x = x_{1:N}$ be observations, β be global latent variables, and $z = z_{1:N}$ be local latent variables, and η be fixed parameters. (We will explain the difference between local and global variables below.) We assume the joint distribution factorizes as

$$p(\beta, z, x \mid \eta) = p(\beta \mid \eta) \prod_{n=1}^{N} p(z_n \mid \beta) p(x_n \mid z_n, \beta). \tag{4}$$

Figure 8 (left) illustrates the corresponding graphical model.

The distinction between local and global variables is manifest in the data generating distribution. The distribution of the nth observation x_n depends only on the nth local variable z_n and the global variables β. For example, in the finite mixture model the local variables are the mixture assignments and the global variables are the mixture proportions and components. From looking at the graphical model, local variables are inside the data plate; global variables are outside the data plate.

The posterior inference problem is to compute the conditional distribution of the latent variables given the data. This is the joint distribution divided by the marginal,

$$p(\beta, z \mid x) = \frac{p(\beta, z, x)}{\int p(\beta, z, x) dz d\beta}. \tag{5}$$

As we have described above, the posterior is essential to using the model. It lets us examine the hidden structures that likely generated the data and, via the posterior predictive distribution, it is the gateway to prediction about new data. The difficulty is in the denominator $p(x)$, the marginal probability of the data. For example, to compute this quantity in the mixture model we must marginalize out every possible combination of assignments of the data points to mixture components. (There are exponentially many such combinations.) This is why, for many models, we must approximate the posterior.

We complete our definition of this class of models by specifying each *complete conditional*, the conditional distribution of a latent variable given the observations and other latent variables. We assume that each complete conditional is in the exponential family.[8] Thus the complete conditional for the global variable is

$$p(\beta \mid x, z) = h(\beta) \exp\{\eta_g(x, z)^\top t(\beta) - a(\eta_g(x, z))\}. \tag{7}$$

The complete conditional for the local variable is

$$p(z_n \mid x_n, \beta) = h(z_n) \exp\{\eta_\ell(x_n, \beta)^\top t(z_n) - a(\eta_g(x_n, \beta))\}. \tag{8}$$

We have overloaded notation for the base measure $h(\cdot)$, sufficient statistic $t(\cdot)$, and log normalizer $a(\cdot)$. For example, one complete conditional might be Gaussian; another might be discrete. Note that the natural parameters are functions of the conditioning variables. Also note that we have used the conditional independencies of the local variables—though they are implicitly conditioned on

[8] If the distribution of x is in an exponential family, then its density has the form

$$p(x \mid \eta) = h(x) \exp\left\{\eta^\top t(x) - a(\eta)\right\}. \tag{6}$$

The function $t(x)$ is the sufficient statistic; the function $h(x)$ is the base measure; the vector η is the natural parameter; the function $a(\eta)$ is the log normalizer, ensuring that the density integrates to one. The derivatives of $a(\eta)$ are the cumulants of the sufficient statistic. Many common distributions are in the exponential family: Gaussian, multinomial/categorical, Poisson, Gamma, Bernoulli, Dirichlet, Beta, and others. See Brown (1986).

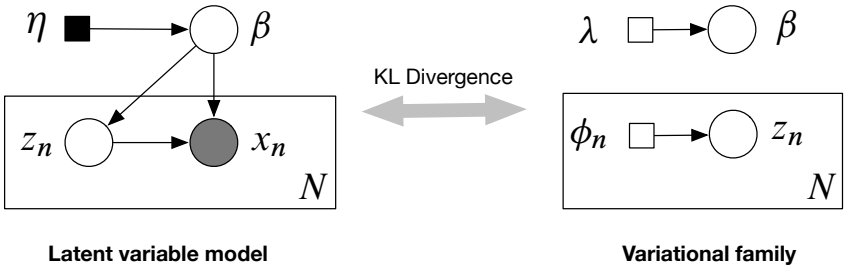

Figure 8: On the left is the general graph structure for a conditionally conjugate model with N observations. On the right is the mean field variational family. In variational inference, these distributions are related by optimizing the variational parameters (unshaded boxes) to make the variational distribution close in KL divergence to the model posterior.

all the observations and other latent variables, the local variables for the nth data context are only dependent on the nth data point and global variables.

Requiring complete conditionals in the exponential family is less stringent than requiring full conjugacy (and this leads to the monicker "conditional conjugacy"). Consider a fully conjugate model, where a latent variable is used as a parameter to the observations. The variables form a conjugate pair if the conditional distribution of the latent variable is in the same family as its prior, a property that depends on both the prior and the data generating distribution (Box and Tiao, 1973; Bernardo and Smith, 1994). For example, if β is drawn from a Dirichlet and $x_{1:N}$ are drawn from a multinomial with parameter β then the conditional distribution of β will remain in the Dirichlet family.

In complex latent variable models, however, the local variables prevent us from choosing priors to make the global variables conjugate to the observations, and this is why we resort to approximate inference. However, conditioned on both the local variables and observations we can often choose prior/likelihood pairs that leave the complete conditional in the same family as the prior. Continuing with mixtures, we can use any common prior/likelihood pair (e.g., Gamma-Poisson, Gaussian-Gaussian, Dirichlet-Multinomial) to build an appropriate conditionally conjugate mixture model.

This general model class encompasses many kinds of models. These include many forms of Bayesian mixture models, mixed-membership models, factor models, sequential models, hierarchical regression models, random effects models, Bayesian nonparametric models, and others. Generic inference algorithms for this class of models give us powerful tools for quickly building, checking, revising, and using sophisticated latent variable models.

4.2 Mean Field Variational Inference

We have defined a large class of models for which we would like to approximate the posterior distribution. We now present mean field variational inference as a simple algorithm for performing this approximation. Mean field inference is a fast and effective method for obtaining approximate posteriors.

Variational inference for probabilistic models was pioneered by machine learning researchers in the 1990s (Jordan et al., 1999; Neal and Hinton, 1999), building on previous work in statistical physics (Peterson and Anderson, 1987). The idea is to posit a family of distributions over the latent variables with free parameters (called "variational parameters") and then fit those parameters to find

the member of the family that is close to the posterior, where closeness is measured by Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951).

We will present coordinate ascent inference for conditionally conjugate models. Variants of this general algorithm have appeared several times in the machine learning research literature (Attias, 1999, 2000; Wiegerinck, 2000; Ghahramani and Beal, 2001; Xing et al., 2003). For good reviews of variational inference in general see Jordan et al. (1999) and Wainwright and Jordan (2008). Here we follow the treatment in Hoffman et al. (2013).

The variational objective function. We denote the variational family over the latent variables by $q(\beta, z \mid \nu)$, where ν are the free variational parameters that index the family. (We specify them in more detail below.) The goal is to find the optimal variational parameters by solving

$$\nu^* = \arg\min_{\nu} \text{KL}(q(\beta, z \mid \nu) || p(\beta, z \mid x). \tag{9}$$

It is through this objective that we tie the variational parameters ν to the observations x (see Figure 8). The inference problem has become an optimization problem.

Unfortunately, computing the KL divergence implicitly requires computing $p(x)$, the same quantity from Equation 5 that makes exact inference impossible. Variational inference optimizes a related objective function,

$$\mathcal{L}(\nu) = \text{E}\left[\log p(\beta, z, x \mid \eta)\right] - \text{E}\left[\log q(\beta, z \mid \nu)\right]. \tag{10}$$

This objective is equal to the negative KL divergence minus $\log p(x)$. Thus maximizing Equation 10 is equivalent to minimizing the divergence. Intuitively, the first term values variational distributions that put mass on latent variable configurations which make the data likely; the second term, which is the entropy of the variational distribution, values diffuse variational distributions.

The mean field variational family. Before optimizing the objective, we must specify the variational family in more detail. We will use the *mean field variational family*, the family where each latent variable is independent and governed by its own variational parameter. Let the variational parameters $\nu = \{\lambda, \phi_{1:N}\}$, where λ is a parameter to the global variable and $\phi_{1:N}$ are parameters to the local variables. The mean field family is

$$q(\beta, z \mid \nu) = q(\beta \mid \lambda) \prod_{n=1}^{N} q(z_n \mid \phi_n). \tag{11}$$

Note that each variable is independent, but they are not identically distributed. Though it cannot capture correlations between variables, this family is very flexible and can place mass on any complex configuration of them. We note that the data does not appear in Equation 11; data connects to the variational parameters only when we optimize the variational objective.[9]

To complete the specification, we set each variational factor to be in the same family as the corresponding complete conditional in the model (Equation 7 and Equation 8). If $p(\beta \mid x, z)$ is a Gaussian then λ are free Gaussian parameters; if $p(z_n \mid x_n, \beta)$ is discrete over K elements then ϕ_n is a free distribution over K elements. [10]

[9] Why do we choose a factored distribution? The reason is deep, having to do with why it is difficult to compute the normalizer of the posterior $p(x)$ and how that relates to graphical models. In brief, computing the marginal is difficult because in the posterior all the latent variables are potentially dependent on each other. In theory we could posit the fully dependent family as the variational family and, indeed, the exact posterior resides in that family. But that merely postpones the problem, as it results in a variational objective which is prohibitive to optimize. We simplify the family, acknowledging that we cannot represent all aspects of the posterior, to make the optimization problem tractable.

[10] Though we assume this structure, Bishop (2006) shows that the optimal variational distribution with factorization as in Equation 11 will necessarily be in this family.

Coordinate ascent variational inference. We now optimize the variational objective in Equation 10. We present the simplest algorithm, coordinate ascent variational inference. In coordinate inference, we iteratively optimize each variational parameter while holding all the other variational parameters fixed. We emphasize again that this algorithm applies to a large collection of models. We can use it to easily perform approximate posterior inference in many data analysis settings.

With conditionally conjugate models and the mean field family, each update is available in closed form. Recall that the global factor $q(\beta \mid \lambda)$ is in the same family as Equation 7. The global update is the expected parameter of the complete conditional,

$$\lambda^* = \mathrm{E}_q[\eta_g(z, x)]. \tag{12}$$

The expectation is with respect to the variational distribution. See Hoffman et al. (2013) for a derivation.

The reason this update works in a coordinate algorithm is that it is only a function of the data and local parameters $\phi_{1:N}$. To see this, note that $\eta_g(z, x)$ is a function of the data and local variables, and recall from Equation 11 that the latent variables are independent in the variational family. Consequently, the expectation of $\eta_g(z, x)$ only involves the local parameters, which are fixed in the coordinate update of the global parameters.

The update for the local variables is analogous,

$$\phi_n^* = \mathrm{E}_q[\eta_\ell(\beta, x_n)]. \tag{13}$$

Again, thanks to the mean field family, this expectation only depends on the global parameters λ, which are held fixed when updating the local parameter ϕ_n.

Putting these steps together, the coordinate ascent inference algorithm is as follows:

1. Initialize global parameters λ randomly.
2. Repeat until the objective converges:
   (a) For each data point, update local parameter ϕ_n from Equation 13.
   (b) Update the global parameter λ from Equation 12.

This algorithm provably goes uphill in the variational objective function, leading to a local optimum. In practice, one uses multiple random restarts to find a good local optimum.

We return briefly to the Gaussian mixture model. The latent variables are the mixture assignments $z_{1:N}$ and mixture components $\mu_{1:K}$. The mean field variational family is

$$q(\mu, z) = \prod_{k=1}^{K} q(\mu_k \mid \lambda_k) \prod_{n=1}^{N} q(z_n \mid \phi_n). \tag{14}$$

The global variational parameters are Gaussian variational means λ_k, which describe the distribution over each mixture component; the local variational parameters are discrete distributions over K elements ϕ_n, which describe the distribution over each mixture assignment. The complete conditional for each mixture component is a Gaussian and the complete conditional for each mixture assignment is a discrete distribution. In step 2a, we estimate the approximate posterior distribution of the mixture assignment for each data point; in step 2b, we re-estimate the locations of the mixture components. This procedure coverges to an approximate posterior for the full mixture model.

Building on mean field inference. We have described the simplest variational inference algorithm. Developing and understanding these methods is an active area of machine learning and statistics research. *Structured variational inference* (Saul and Jordan, 1996) relaxes the mean field assumption; the variational distribution allows some dependencies among the variables. *Nonconjugate variational inference* (Wang and Blei, 2013; Knowles and Minka, 2011) relaxes the assumption that the complete conditionals are in the exponential family. Previous work on nonconjugate models required specialized approximations. These works seek to generalize those approaches. Finally, in *stochastic variational inference* (Hoffman et al., 2013) we iteratively subsample from the data, optimize the subsample's local parameters, and then adjust the global variational parameters; this leads to much faster convergence of the variational objective and accommodates massive data sets. In general, variational methods are a powerful tool for approximate posterior inference in complex models.

5 Model Criticism

With the tools from Sections 2, 3 and 4, the data-rich reader can compose complex models and approximate their posteriors with mean-field variational inference. These constitute the first two components of Box's loop in Figure 1. In this section, we discuss the final component, model criticism.

Typically, we perform two of types of tasks with a model—exploration and prediction. In exploration, we use our inferences about the hidden variables—usually, through approximate posterior expectations—to summarize the data, visualize the data, or facet the data, i.e., dividing it into groups and structures dictated by the inferences. Examples of exploratory tasks include using topic models to navigate large collections of documents or using clustering models on microarray data to suggest groups of related genes.

In prediction, we forecast future data using the posterior predictive distribution,

$$p(x_{\text{new}} \mid x) = \int p(\beta \mid x) \left(\int p(z_{\text{new}} \mid \beta) p(x_{\text{new}} \mid z_{\text{new}}, \beta) dz_{\text{new}} \right) d\beta. \tag{15}$$

(Equation 3 gave the predictive distribution for a mixture model.) Usually the posterior $p(\beta \mid x)$ is not available and we substitute an approximation $q(\beta)$, which we find by an approximate inference algorithm like MCMC or variational inference. Examples of predictive tasks include predicting which items a user will purchase based on a matrix factorization model or predicting future stock prices based on a sequential model of market history.

Both exploratory and predictive tasks require that we assess the fitness of the model, understanding how good it is in general and where it falls short in particular. The ultimate measure of model fitness is to use it for the task at hand—e.g., deploy a recommendation system on the web, trade on predictions about the stock market, form clusters of genes that biologists find useful—but it is also essential to evaluate methods as part of the iterative model-building process. In this section, we will review two useful general techniques: predictive likelihood with sample-reuse (Geisser, 1975) and posterior predictive checks (Box, 1980; Rubin, 1984; Meng, 1994; Gelman et al., 1996). Conceptually, both methods involve confronting the model's posterior predictive distribution with the observed data: a misfitted model's predictive distribution will be far away from the observations.

The practice of model criticism is fundamentally different from the practice of model selection (Claeskens and Hjort, 2008), the problem of choosing among a set of alternative models. First, we can criticize a model either with or without an alternative in mind. If our budget for time and energy only allowed for developing a single model, it is still useful to know how and where it succeeds and fails. Second, in real-world settings, the posterior predictive distribution of Equation 15 is a function of both the

model and chosen inference algorithm for approximating $p(\beta \mid x)$. We should test this bundle directly to better assess how the proposed solution—model and inference algorithm—will fare when deployed to its assigned task.

Finally, we comment about the relationship between model criticism and orthodox Bayesian thinking. Orthodox Bayesian thinking requires that all the uncertainty about our data be encoded in the model: if we think that multiple models might be at play then we build a "super model" that connects them in a mixture and integrate out the model choice, or we approximate the marginal probability of the observations (Kass and Raftery, 1995; MacKay, 2003) to determine which model more likely generated the data.

Model criticism, on the other hand, views model formulation as part of the iterative process of Box's loop. Criticism techniques seek to show how a model falls short and point to where we should make it more complex. Here we step out of the formal Bayesian framework to ask how well the model (and inference) captures the underlying data distribution, a perspective that is beautifully laid out in Box (1980) and Rubin (1984). Moreover, if we think of our model as a "working theory" about the data, these methods connect to ideas of falsification in the philosophy of science (Popper, 1959; Gelman and Shalizi, 2012). Observations that do not match the predictions of a theory are evidence against it, and point us in the directions of improvement.[11]

5.1 Predictive Sample Re-use

One way to assess a model (and its corresponding inference algorithm) is to evaluate its generalization performance, the probability that it assigns to unseen data. Geisser (1975) proposed "predictive sample re-use" (PSR), which amounts to using cross-validation to estimate this probability.[12]

Let $x_{[n]}$ be the data set with the nth item removed. Suppose our model is $p(\beta, z, x)$ and we used variational inference to approximate the posterior $p(\beta, z \mid x_{[n]})$ with $q_{[n]}(\beta, z)$. The log predictive likelihood for the nth data point is

$$\begin{aligned} \ell_n &= \log p(x_n \mid x_{[n]}) && (16) \\ &= \log \int \left(\int p(x_n \mid z_n, \beta) q(z_n) dz_n \right) q_{[n]}(\beta) d\beta. && (17) \end{aligned}$$

This is the (approximate) posterior predictive probability of the nth data point using data that does not contain it. The full predictive likelihood is $\sum_{n=1}^{N} \ell_n$. It estimates the held-out log probability of new data with leave-one-out cross validation. Note that $q(z_n)$ is the local variational parameter for the nth data point; it is computed holding $q_{[n]}$ fixed (which did not account for x_n) and estimating the posterior local context for x_n.

This procedure is expensive because it requires fitting N approximate posteriors, one for each data point. In practice, we can use K-fold cross-validation to estimate the score. We divide our data into K groups; we iteratively hold each group out and approximate the posterior over the global variables β for each group; finally we compute ℓ_n for each data point using the approximate posterior from the group that does not contain it.

One advantage of PSR is that it does not just help evaluate the model, but also places all approximate inference algorithms on the same playing field. Consider the model and inference algorithm as a conduit between the data and a predictive distribution of new data; the log predictive probability of

[11] All this said, we note that model criticism is a philosophically controversial idea in the Bayesian literature as it is "incoherent" under certain theoretical frameworks. See the discussion of Box (1980) and the comments in Lauritzen (2007).

[12] Cross-validation was co-invented by Geisser, independently of Stone (1974).

Equation 17 evaluates the predictive distribution no matter how it was approximated. In contrast, model selection based on approximations to the marginal probability of the observations (Kass and Raftery, 1995; MacKay, 2003) may be subject to unknown biases. For example, it is difficult to compare approximations to the marginal based on variational inference to those based on MCMC. Predictive sample re-use easily lets us compare two different approximations of the same predictive distribution.

A further advantage is that PSR can be adapted to the prediction problems most relevant to the model at hand. For example, in grouped data we may want to consider each group to be partially observed and assess the predictive likelihood of the remaining observations. This is a good way to evaluate probabilistic topic models (Blei and Lafferty, 2007; Asuncion et al., 2009). Moreover, the individual scores in Equation 17 can be examined in the same way that we examine residuals—practitioners can look for patterns in poor predictions to identify where a model succeeds and fails.

5.2 Posterior Predictive Checks

Posterior predictive checks (PPCs) (Guttman, 1967; Box, 1980; Rubin, 1984; Meng, 1994; Gelman et al., 1996) (PPCs) help answer the important question: "Is my model good enough in the ways that matter?" In a PPC, we locate the observed data in its posterior predictive distribution. If we find our observed data is not typical, i.e., it has low probability under the posterior predictive distribution, then there may be an issue with the model.

PPCs cleanly separate what we care about modeling from what we easily can model. The model building process requires computational comprimises—a Gaussian distribution where it is not appropriate or an independence assumption that we know not to be true—but a good model captures what we care about even in light of those compromises. PPCs are a tool for diagnosing whether our simplified models, built for computational convenience, are good enough in light of the aspects of the data that are important to us.

Here is the intuition, paraphrased from Rubin (1984). Suppose we were to repeat the data collection process that gave our observed data. This new data comes from the same process as the observations, so we expect the two data sets to be similar. Now consider the proposed model. If it is suitable then its posterior predictive distribution will give a good approximation to the data collection distribution, that is, our observations will lead to a predictive mechanism that captures their distribution. Thus we consider what a data set would look like if it were drawn from the model's posterior predictive distribution. If it likely does not look like the observations—the data drawn from the distribution that we are hoping the model will capture—then this indicates there is a problem with the model.

More formally, we define a *discrepancy* $T(X)$ to be a function of the data that matters to us—it is a property of the data that we hope our model captures in its predictive distribution. Let x^{rep} be a random set of observations. The posterior predictive check is

$$\text{PPC} = P(T(X^{\text{rep}}) > T(x) \mid x). \tag{18}$$

The only random variable is this expression is x^{rep}, a data set drawn from from the posterior predictive distribution. This PPC estimates the probability that the replicated data is "far away" (in terms of T) from the observations.

An important development in Meng (1994) is to define the discrepancy as a function of both the data and the latent variables $T(x, \beta)$. (For cleaner notation, we omit the local variable.) The PPC becomes

$$\text{PPC} = P(T(X^{\text{rep}}, \beta) > T(x, \beta) \mid x). \tag{19}$$

Here both the hidden variables β and replicated data x^{rep} are random. Their joint distribution is a product of the posterior and data generating distribution, $p(\beta, x^{\text{rep}} \mid x) = p(\beta \mid x)p(x^{\text{rep}} \mid \beta)$. Thus

the PPC can be decomposed as an expectation of an indicator

$$\text{PPC} = \int p(\beta \mid x) \int p(x^{\text{rep}} \mid \beta) \mathbf{1}[T(x^{\text{rep}}, \beta) > T(x, \beta)] dx^{\text{rep}} d\beta. \tag{20}$$

This reveals how to estimate it with a Monte Carlo approximation. For T replications:

1. Draw $\beta^{(t)}$ from the posterior $p(\beta \mid x)$ or approximate posterior.
2. Draw replicated data $x^{(t)}$ from the sampled $\beta^{(t)}$
3. Compute the discrepancies on both the sampled data $T(x^{(t)}, \beta^{(t)})$ and originally observed data $T(x, \beta^{(t)})$.

The "posterior predictive p-value" is the proportion of cases where $T(x^{(t)}, \beta^{(t)}) > T(x, \beta^{(t)})$. Gelman et al. (1996) emphasize the additional value of scatter plots (sampled values of β by the discrepancies) to further criticize and assess the model.

For example, consider the discrepancy to be the average log probability of the data,

$$T(x, \beta) = \frac{1}{N} \sum_{n=1}^{N} \log p(x_n \mid \beta). \tag{21}$$

We sample from the posterior, replicate a data set from the sample, and compute the average log probability of both the observed data and replicated data. In spirit, this leads to a check similar to classical goodness-of-fit tests of residuals (Cook and Weisberg, 1982): a model is a poor fit if the observed log probabilities are consistently smaller than those that are generated by the model's posterior.

A key advantage of the PPC is that it is adaptable to the particular needs of the practitioner. For example, the log score discrepancy can be the median of the log probabilities to account for outliers, a set of conditional probabilities if that is how the model will be used, or weighted towards the kinds of observations that are important to model well in the application at hand. We can further look at multiple discrepancies, even reusing computation to do so, to understand trade-offs in various models.

Examples. In applied data analysis, PPCs are not as widely used as we would hope. Part of our goal here is to rekindle interest in them as a general module in Box's loop. PPCs are best understood through example.

In astrophysics data, an emission line gives clues as to the chemical composition of stars. Evidence for emission lines is from significant overabundance of certain frequencies in spectral measurements. Such evidence is sparse, however, and whether it is "significant" is important to forming new astrophysics theories and designing subsequent experiments. Van Dyk and Kang (2004) used posterior predictive checks on complex spectral models—with a predictive likelihood as the discrepancy function—to determine the existence of emission lines.

Belin and Rubin (1995) and Gelman et al. (2005) provide two good examples of how PPCs can iteratively point to new models and help suggest when to stop expanding. Gelman et al. (2005) look at residual variance in a designed experiment. Their data set contains groups, and they use PPCs to decide to include group-level variance terms in their model. They further focus on how PPCs can be used with imputed data, local hidden variables (in the parlance of this paper) that are filled in by the model, to directly estimate discrepancies that are otherwise confounded by missingness.

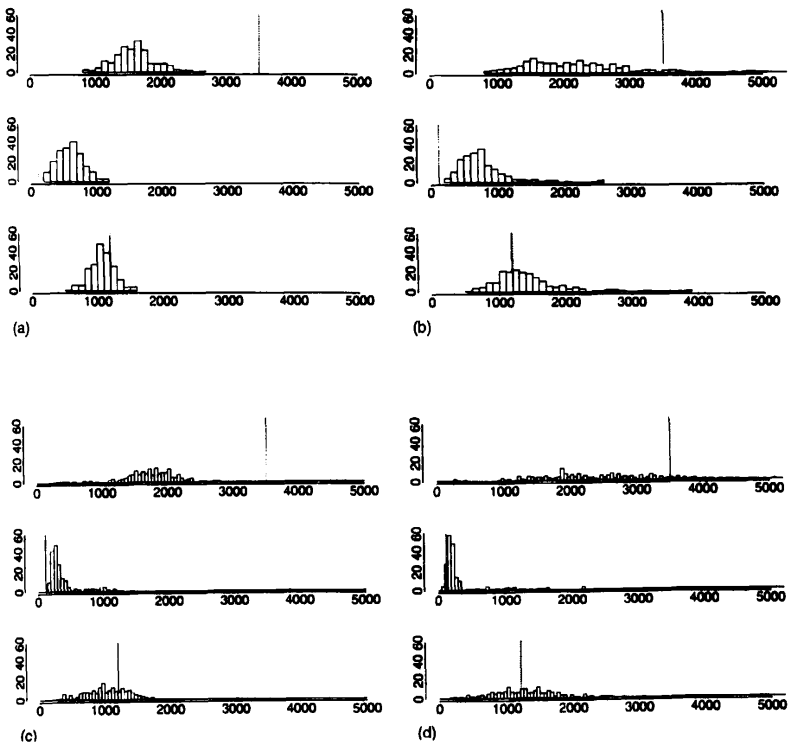

Figure 9: Figure 2 from Belin and Rubin (1995). There are three discrepancy measures and four models (a–d), in the order the authors built them. The predictive distributions from the later models better fit the observed data, where "fitness" is measured by the discrepancies.

Belin and Rubin (1995) use the PPC methodology to implement several iterations of Box's loop. They analyzed data from an experiment that collected response times by schizophrenics and non-schizophrenics on a psychological task. Their discrepancies were the largest observed variance for schizophrenics; the smallest observed variance for schizophrenics; and the average within-person variance across all subjects. Notice these would be difficult to build into the model, but they are easy to check in replicated data. In checking these multiple aspects of the replicated data, they build a sequence of four increasingly complex models—all on a single data set—before settling on one to assess on future data. Figure 9 is taken from their paper. It is clear how the sequence of models gives increasingly good predictive distributions of the observed data.

As a final example we summarize our work on PPCs for topic models (Mimno and Blei, 2011). We used a multivariate PPC on a probabilistic topic model to determine which components were meaningful and which violated the independence assumptions of the model. Our discrepancy was a per-topic conditional mutual information between latent variables and observations, a pair of variables that should be conditionally independent under the modeling assumptions. This work highlights an untraditional application of PPCs. When we use large latent variable models to explore data, we aim to find hidden structure that encodes interesting patterns. As all models are necessarily simplifications, we do not realistically expect any model to be "adequate". We used PPCs to filter out aspects of the model that were not capturing the data well so that we could better focus our explorations and visualizations on the meaningful patterns.

6 Discussion

We have reviewed the components of Box's loop (Figure 1), an iterative process for building and revising probabilistic models. We posit a model with a graphical model, use generic approximate inference methods to compute with it, and then step out of our assumptions to assess and criticize how well it fits our observed data. Implementing Box's loop with modern statistics and machine learning methods allows us to develop complex probabilistic models to solve real-world data analysis problems.

Currently, this loop is implemented at the level of a scientific community: researchers build a model, derive an inference method, and use the model to solve a problem. Other researchers (or the same researchers, on a different project) identify how the model falls short, build an improved model, and demonstrate the improvement. One goal for statisticians and computer scientists is to develop tools— methodological tools and technological tools—for implementing the loop more easily and more efficiently. These tools will let us develop and use *sequences* of models before settling on the appropriate one for the data at hand.

There are several challenges. One challenge is that we need approximate inference algorithms that are both scalable and generic, algorithms that can be used to compute with a large class of models and with massive data sets. One promising avenue of research in this vein is probabilistic programming. Researchers in probabilistic programming develop software that allows users to easily specify models and compute approximate posteriors (Gilks and Spiegelhalter, 1992; Bishop et al., 2003; Minka et al., 2010; McCallum et al., 2009; Stan Development Team, 2013). These existing systems are a good start, but probabilistic programming must be more flexible and more efficient for iterative model-building to become a standard practice.

A second challenge is to continue to develop the theory and methods of exploratory data analysis. Exploratory analysis has become increasingly important in recent years as scientists and other data consumers want to discover, understand, and exploit patterns in streams of observational data. (Such goals are outside of the traditional scientific goal of drawing conclusions from a designed experiment.) In this spirit, we should continue to develop methods like posterior predictive checks to help navigate large data sets with complex models. We should further develop the foundations of data exploration, following the lines of Tukey (1962), Good (1983), and Diaconis (1985), to develop principled methods for exploring observational data.

But the most important challenges are those we cannot yet identify. The limitations of our methods will only be revealed when we circle Box's loop with new data sets and new problems. In a sense, this is itself an instance of Box's loop. When we collaborate with scientists and engineers to solve new data analysis problems, we can identify how our methods fall short. These shortcomings, in turn, point us to where and how our methods can be improved.

Acknowledgments

This work is supported by NSF CAREER NSF IIS-0745520, NSF BIGDATA NSF IIS-1247664, NSF NEURO NSF IIS-1009542, ONR N00014-11-1-0651, and the Alfred P. Sloan foundation. The author is grateful to Steve Fienberg, Sean Gerrish, Jeremy Manning, David Mimno, Chris Wiggins, Chong Wang, and Tingran Wang for useful comments and discussions.

References

Airoldi, E., Blei, D., Fienberg, S., and Xing, E. (2008). Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 9:1981–2014.

Antoniak, C. (1974). Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174.

Asuncion, A., Welling, M., Smyth, P., and Teh, Y. (2009). On smoothing and inference for topic models. In *Uncertainty in Artificial Intelligence*.

Attias, H. (1999). Inferring parameters and structure of latent variable models by variational bayes. In *Uncertainty in Artificial Intelligence*, pages 21–30. Morgan Kaufmann Publishers Inc.

Attias, H. (2000). A variational Bayesian framework for graphical models. In *Advances in Neural Information Processing Systems 12*.

Baker, F. B. (1992). *Item Response Theory*. Marcel Dekker, New York.

Bar-Shalom, Y., Li, X., and Kirubarajan, T. (2004). *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons.

Bartholomew, D., Knott, M., and Moustaki, I. (2011). *Latent Variable Models and Factor Analysis: A unified approach*, volume 899. Wiley. com.

Belin, T. and Rubin, D. (1995). The analysis of repeated-measures data on schizophrenic reaction times using mixture models. *Statistics in Medicine*, 14(8):747–768.

Bernardo, J. and Smith, A. (1994). *Bayesian Theory*. John Wiley & Sons Ltd., Chichester.

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer New York.

Bishop, C. (2013). Model-based machine learning. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371(1984).

Bishop, C., Spiegelhalter, D., and Winn, J. (2003). VIBES: A variational inference engine for Bayesian networks. In *Neural Information Processing Systems*. Cambridge, MA.

Blei, D. (2012). Probabilistic topic models. *Communications of the ACM*, 55(4):77–84.

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. and Lafferty, J. (2006). Dynamic topic models. In *International Conference on Machine Learning*, pages 113–120, New York, NY, USA. ACM.

Blei, D. and Lafferty, J. (2007). A correlated topic model of Science. *Annals of Applied Statistics*, 1(1):17–35.

Blei, D., Ng, A., and Jordan, M. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Box, G. (1976). Science and statistics. *Journal of the American Statistical Association*, 71(356):791–799.

Box, G. (1980). Sampling and Bayes' inference in scientific modeling and robustness. *Journal of the Royal Statistical Society, Series A*, 143(4):383–430.

Box, G. and Hill, W. (1967). Discrimination among mechanistic models. *Technometrics*, 9(1):pp. 57–71.

Box, G. and Hunter, W. (1962). A useful method for model-building. *Technometrics*, 4(3):pp. 301–318.

Box, G. and Hunter, W. (1965). The experimental study of physical mechanisms. *Technometrics*, 7(1):pp. 23–42.

Box, G. and Tiao, G. (1973). *Bayesian inference in statistical analysis*. John Wiley & Sons.

Brown, L. (1986). *Fundamentals of Statistical Exponential Families*. Institute of Mathematical Statistics, Hayward, CA.

Claeskens, G. and Hjort, N. (2008). *Model Selection and Model Averaging*. Cambridge University Press.

Clinton, J., Jackman, S., and Rivers, D. (2004). The statistical analysis of roll call data. *American Political Science Review*, 98(2):355–370.

Collins, M., Dasgupta, S., and Schapire, R. (2002). A generalization of principal component analysis to the exponential family. In *Neural Information Processing Systems*.

Cook, R. and Weisberg, S. (1982). *Residuals and Influence in Regression*.

Dawid, A. and Lauritzen, S. (1993). Hyper Markov laws in the statistical analysis of decomposable graphical models. *The Annals of Statistics*, 21(3):1272–1317.

Diaconis, P. (1985). Theories of data analysis: From magical thinking through classical statistics. In *Exploring Data: Tables, Trends, and Shapes*, pages 1–36.

Durbin, R., Eddy, S., Krogh, A., and Mitchison, G. (1998). *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press.

Efron, B. (2013). Empirical bayes modeling, computation, and accuracy.

Efron, B. and Morris, C. (1973). Combining possibly related estimation problems. *Journal of the Royal Statistical Society, Series B*, 35(3):379–421.

Erosheva, E., Fienberg, S., and Joutard, C. (2007). Describing disability through individual-level mixture models for multivariate binary data. *Annals of Applied Statistics*.

Erosheva, E., Fienberg, S., and Lafferty, J. (2004). Mixed-membership models of scientific publications. *Proceedings of the National Academy of Science*, 97(22):11885–11892.

Ferguson, T. (1973). A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1:209–230.

Geisser, S. (1975). The predictive sample reuse method with applications. *Journal of the American Statistical Association*, 70:320–328.

Gelfand, A. and Smith, A. (1990). Sampling based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85:398–409.

Gelman, A., Carlin, J., Stern, H., and Rubin, D. (1995). *Bayesian Data Analysis*. Chapman & Hall, London.

Gelman, A. and Hill, J. (2007). *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press.

Gelman, A., Meng, X., and Stern, H. (1996). Posterior predictive assessment of model fitness via realized discrepancies. *Statistica Sinica*, 6:733–807.

Gelman, A. and Shalizi, C. (2012). Philosophy and the practice of Bayesian statistics. *British Journal of Mathematical and Statistical Psychology*.

Gelman, A., Van Mechelen, I., Verbeke, G., Heitjan, D., and Meulders, M. (2005). Multiple imputation for model checking: Completed-data plots with missing and latent data. *Biometrics*, 61(1):74–85.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741.

Gershman, S. and Blei, D. (2012). A tutorial on Bayesian nonparametric models. *Journal of Mathematical Psychology*, 56:1–12.

Ghahramani, Z. (2012). Bayesian nonparametrics and the probabilistic approach to modelling. *Philosophical Transactions of the Royal Society of London. Series A*.

Ghahramani, Z. and Beal, M. (2001). Propagation algorithms for variational Bayesian learning. In *NIPS 13*, pages 507–513.

Gilks, W. and Spiegelhalter, D. (1992). A language and program for complex Bayesian modelling. *The Statistician*, 3:169–177.

Good, I. (1983). The philosophy of exploratory data analysis. *Philosophy of Science*, pages 283–295.

Guttman, I. (1967). The use of the concept of a future observation in goodness-of-fit problems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 83–100.

Hastings, W. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57:97–109.

Hjort, N., Holmes, C., Muller, P., and Walker, S., editors (2010). *Bayesian Nonparametrics*. Cambridge University Press.

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

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417.

Hotelling, H. (1936). Relations between two sets of variates. *Biometrika*, 28(3/4):321–377.

Ishwaran, H. and Rao, J. S. (2005). Spike and slab variable selection: Frequentist and Bayesian strategies. *The Annals of Statistics*, 33(2):730–773.

Jordan, M., editor (1999). *Learning in Graphical Models*. MIT Press, Cambridge, MA.

Jordan, M. (2004). Graphical models. *Statistical Science*, 19(1):140–155.

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

Kalman, R. (1960). A new approach to linear filtering and prediction problems a new approach to linear filtering and prediction problems,". *Transaction of the AMSE: Journal of Basic Engineering*, 82:35–45.

Kass, R. and Raftery, A. (1995). Bayes factors. *Journal of the American Statistical Association*, 90(430):773–795.

Knowles, D. A. and Minka, T. (2011). Non-conjugate variational message passing for multinomial and binary regression. In *Neural Information Processing Systems*.

Koller, D. and Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.

Krnjajić, M., Kottas, A., and Draper, D. (2008). Parametric and nonparametric Bayesian model specification: A case study involving models for count data. *Computational Statistics and Data Analysis*, 52(4):2110–2128.

Kullback, S. and Leibler, R. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

Lauritzen, S. (2007). Discussion of some aspects of model selection for prediction: Article of chakrabarti and ghosh. In et al., J. M. B., editor, *Bayesian Statistics 8*, pages 84–90. Oxford University Press, Oxford.

Lee, D. and Seung, H. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791.

MacKay, D. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.

McAuliffe, J., Pachter, L., and Jordan, M. (2004). Multiple-sequence functional annotation and the generalized hidden Markov phylogeny. *Bioinformatics*, 20(12):1850–1860.

McCallum, A., Schultz, K., and Singh, S. (2009). Factorie: Probabilistic programming via imperatively defined factor graphs. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C. K. I., and Culotta, A., editors, *Advances in Neural Information Processing Systems 22*, pages 1249–1257.

McCullagh, P. and Nelder, J. A. (1989). *Generalized Linear Models*. London: Chapman and Hall.

Meng, X. (1994). Posterior predictive p-values. *The Annals of Statistics*, pages 1142–1160.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, M., and Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1092.

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

Minka, T., Winn, J., Guiver, J., and Knowles, D. (2010). Infer.NET 2.4. Microsoft Research Cambridge. http://research.microsoft.com/infernet.

Mohamed, S., Ghahramani, Z., and Heller, K. (2008). Bayesian exponential family PCA. In *Neural Information Processing Systems*, pages 1089–1096.

Morris, C. (1983). Parametric empirical Bayes inference: Theory and applications. *Journal of the American Statistical Association*, 78(381):47–65. With discussion.

Murphy, K. (2013). *Machine Learning: A Probabilistic Approach*. MIT Press.

Neal, R. and Hinton, G. (1999). A view of the EM algorithm that justifies incremental, sparse, and other variants. In *Learning in Graphical Models*, pages 355–368. MIT Press.

Nelder, J. A. and Wedderburn, R. W. M. (1972). Generalized linear models. *Journal of the Royal Statistical Society. Series A (General)*, 135:370–384.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Pearson, K. (1901). Principal components analysis. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 6(2):559.

Peterson, C. and Anderson, J. (1987). A mean field theory learning algorithm for neural networks. *Complex Systems*, 1(5):995–1019.

Popper, K. R. (1959). The logic of scientific discovery. *London: Hutchinson*, 1.

Pritchard, J., Stephens, M., and Donnelly, P. (2000). Inference of population structure using multilocus genotype data. *Genetics*, 155:945–959.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286.

Robbins, H. (1980). An empirical bayes estimation problem. *Proceedings of the National Academy of Sciences*, 77(12):6988.

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407.

Robert, C. and Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer-Verlag, New York, NY.

Roweis, S. (1998). EM algorithms for PCA and SPCA. In *Neural Information Processing Systems*, pages 626–632.

Rubin, D. (1984). Bayesianly justifiable and relevant frequency calculations for the applied statistician. *The Annals of Statistics*, 12(4):1151–1172.

Rue, H., Martino, S., and Chopin, N. (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society, Series B (Methodological)*, 71(2):319–392.

Salakhutdinov, R. and Mnih, A. (2008). Probabilistic matrix factorization. *Advances in Neural Information Processing Systems*, 20:1257–1264.

Saul, L. and Jordan, M. (1996). Exploiting tractable substructures in intractable networks. *Neural Information Processing Systems*, pages 486–492.

Siepel, A. and Haussler, D. (2004). Combining phylogenetic and hidden Markov models in biosequence analysis. *Journal of Computational Biology*, 11(2-3):413–428.

Skrondal, A. and Rabe-Hesketh, S. (2007). Latent variable modelling: A survey. *Scandinavian Journal of Statistics*, 34(4):712–745.

Smola, A., Vishwanathan, V., and Eskin, E. (2003). Laplace propagation. In *Advances in Neural Information Processing Systems*.

Stan Development Team (2013). Stan modeling language.

Steyvers, M. and Griffiths, T. (2006). Probabilistic topic models. In Landauer, T., McNamara, D., Dennis, S., and Kintsch, W., editors, *Latent Semantic Analysis: A Road to Meaning*. Laurence Erlbaum.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 111–147.

Teh, Y. and Jordan, M. (2008). Hierarchical Bayesian nonparametric models with applications.

Thomson, G. (1939). The factorial analysis of human ability. *British Journal of Educational Psychology*, 9(2):188–195.

Thurstone, L. (1931). Multiple factor analysis. *Psychological Review*, 38(5):406.

Thurstone, L. (1938). Primary mental abilities. *Psychometric monographs*.

Tierney, L., Kass, R., and Kadane, J. (1989). Fully exponential Laplace approximations to expectations and variances of nonpositive functions. *Journal of American Statistical Association*, 84(407).

Tipping, M. and Bishop, C. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622.

Tukey, J. (1962). The future of data analysis. *The Annals of Mathematical Statistics*, 33(1):1–67.

Van Dyk, D. and Kang, H. (2004). Highly structured models for spectral analysis in high-energy astrophysics. *Statistical Science*, pages 275–293.

Wainwright, M. and Jordan, M. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305.

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

West, M. and Harrison, J. (1997). *Bayesian Forecasting and Dynamic Models*. Springer.

Wiegerinck, W. (2000). Variational approximations between mean field theory and the junction tree algorithm. In *Uncertainty in Artificial Intelligence*, pages 626–633.

Xing, E., Jordan, M., and Russell, S. (2003). A generalized mean field algorithm for variational inference in exponential families. In *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence*.