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

David M. Blei

Computer Science Department, Princeton University, Princeton, New Jersey 08540;
email: blei@cs.princeton.edu

Annu. Rev. Stat. Appl. 2014. 1:203–32

The *Annual Review of Statistics and Its Application* is online at statistics.annualreviews.org

This article's doi:
10.1146/annurev-statistics-022513-115657

Keywords

latent variable models, graphical models, variational inference, predictive sample reuse, posterior predictive checks

Abstract

We survey latent variable models for solving data-analysis problems. A latent variable model is a probabilistic model that encodes hidden patterns in the data. We uncover these patterns from their conditional distribution and use them to summarize data and form predictions. Latent variable models are important in many fields, including computational biology, natural language processing, and social network analysis. Our perspective is that models are developed iteratively: We build a model, use it to analyze data, assess how it succeeds and fails, revise it, and repeat. We describe how new research has transformed these essential activities. First, we describe probabilistic graphical models, a language for formulating latent variable models. Second, we describe mean field variational inference, a generic algorithm for approximating conditional distributions. Third, we describe how to use our analyses to solve problems: exploring the data, forming predictions, and pointing us in the direction of improved models.

1. INTRODUCTION

This review is about the craft of building and using probability models to solve data-driven problems. We focus on latent variable models, which 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 some problems that latent variable models can help solve:

1. 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, such as people who know each other but are not yet linked or people who would enjoy meeting each other.
2. 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.
3. 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.

Data-analysis problems like these, and solutions using latent variable models, abound in many fields of science, government, and industry. In this review, we 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.

1.1. Box’s Loop

Our perspective is that building and using latent variable models are part of an iterative process for solving data-analysis problems. First, formulate a simple model based on the types of hidden structures that you believe exist 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 patterns that your data exhibit. 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 beginning in the 1960s (Box & Hunter 1962, 1965; Box & 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, in which 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, in

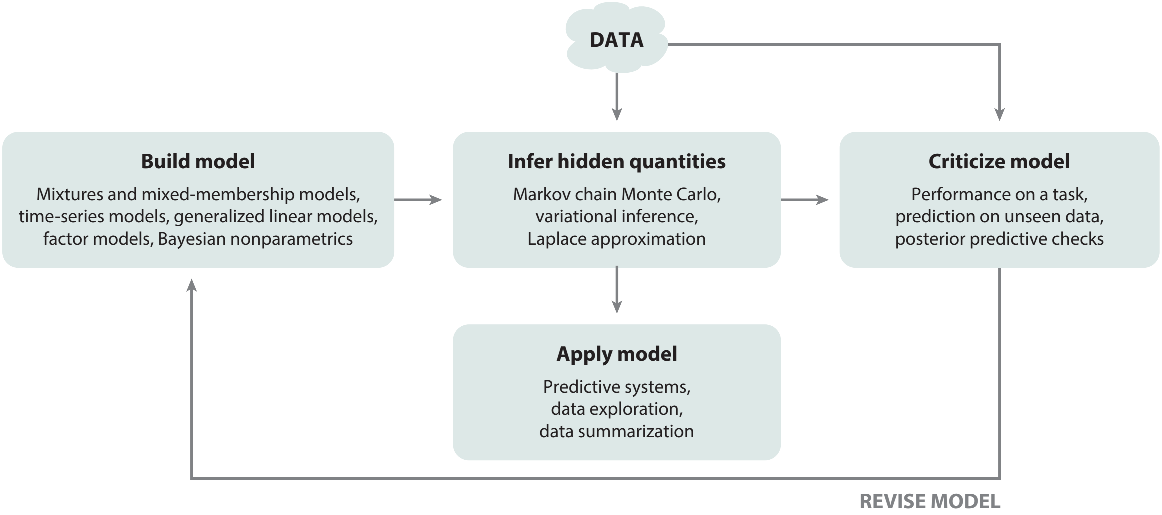

DATA
Build model
Mixtures and mixed-membership models, time-series models, generalized linear models, factor models, Bayesian nonparametrics
Infer hidden quantities
Markov chain Monte Carlo, variational inference, Laplace approximation
Criticize model
Performance on a task, prediction on unseen data, posterior predictive checks
Apply model
Predictive systems, data exploration, data summarization
REVISE MODEL

Figure 1

Box's loop. Building and computing with models are part of an iterative process for solving data-analysis problems. This is Box's loop, a modern interpretation of the perspective of Box (1976).

which the goal is to summarize, visualize, and hypothesize about observational data, namely data that we observe but that are not part of a designed experiment.

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 finally using the evaluation to revise the model's assumptions. 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 50 years since Box's original research. 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 review provides 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. The model itself is a well-defined mathematical object, but building a model is an art. For interesting perspectives, the reader is referred to Box & Draper (1987), Lehmann (1990), and Good [2009 (1983)]. In this review, we will draw on probabilistic graphical models (Pearl 1988, Dawid & Lauritzen 1993, Jordan 2004), 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 types of hidden structures are governing the data and construct complex models from simpler components—such as 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 enter 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 observations, 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 such as Markov chain Monte Carlo (MCMC) (Metropolis et al. 1953, Hastings 1970, Geman & Geman 1984) and variational inference (Peterson & Anderson 1987, Jordan et al. 1999, Wainwright & Jordan 2008) make it possible for us to examine large 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.

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 such as predictive sample reuse (PSR) (Geisser 1975) and posterior predictive checks (PPCs) (Box 1980, Rubin 1984, Meng 1994, Gelman et al. 1996). These techniques assess a model's fitness by contrasting the predictions that it makes against the observed data in the ways that matter to the task at hand. This activity, known as model criticism, is essential to solving modern data-analysis problems.

1.2. This Review

We 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 provide graphical model 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 that can handle large data sets. Finally, in Section 5, we describe how to criticize and assess the fitness of a model, using predictive likelihood and PPCs.

Again, this article represents a curated view. For other surveys of latent variable models, see Skrondal & Rabe-Hesketh (2007), Ghahramani (2012), and Bishop (2013). For more complete treatments of probabilistic modeling, see books such as 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 survey some of its tools, but the reader will master them only as with any other craft—with practice.

2. LATENT VARIABLE MODELS

When we build a latent variable model, we imagine what types of hidden quantities might be used to describe the 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 posterior, which is the conditional distribution of the hidden variables given the observations. Furthermore, we use the posterior to

[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 language 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). Furthermore, 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.

form the predictive distribution, the distribution over future data that the observations and the model imply.

For example, the mixture model is 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 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, which encodes hidden quantities (such as 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, which is a fixed nonrandom quantity that we denote by η. Note that we focus on models of hidden and observed random variables, and we always assume that the hyperparameters are fixed. Estimating hyperparameters from data is the important problem of empirical Bayes (Efron & Morris 1973, Robbins 1980, Morris 1983, Efron 2013).

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, $p(h \mid x, \eta) \propto p(h, x \mid \eta)$. The conditional distribution also leads to the predictive distribution, $p(x_{\text{new}} \mid x) = \int p(x_{\text{new}} \mid h) p(h \mid x, \eta) \mathrm{d}h$.

We continue with the example of a Gaussian mixture model. The variables of the model are K mixture components $\mu_{1:K}$, each of which is the mean of a Gaussian distribution, and a set of mixture proportions θ, a nonnegative 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 complete the model, we place distributions, known as 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 these 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. 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.[2] The predictive distribution, derived from the posterior, provides the distribution of the next data point. **Figure 2** shows an example. Below, we describe three ways of specifying a model: its generative probabilistic process, its joint distribution, and its directed graphical model.

2.1. The Generative Probabilistic Process

The generative probabilistic process describes how data would arise from the model. Though the model is rarely "true," the generative process helps make clear how the latent variables interact

[2] Note, however, 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, that is, a clustering. (Furthermore, note that the number of mixture components is fixed—the number we chose may not be appropriate.) After we have formulated the model and inferred the posterior, we discuss in Section 5 how to check whether it adequately explains the data.

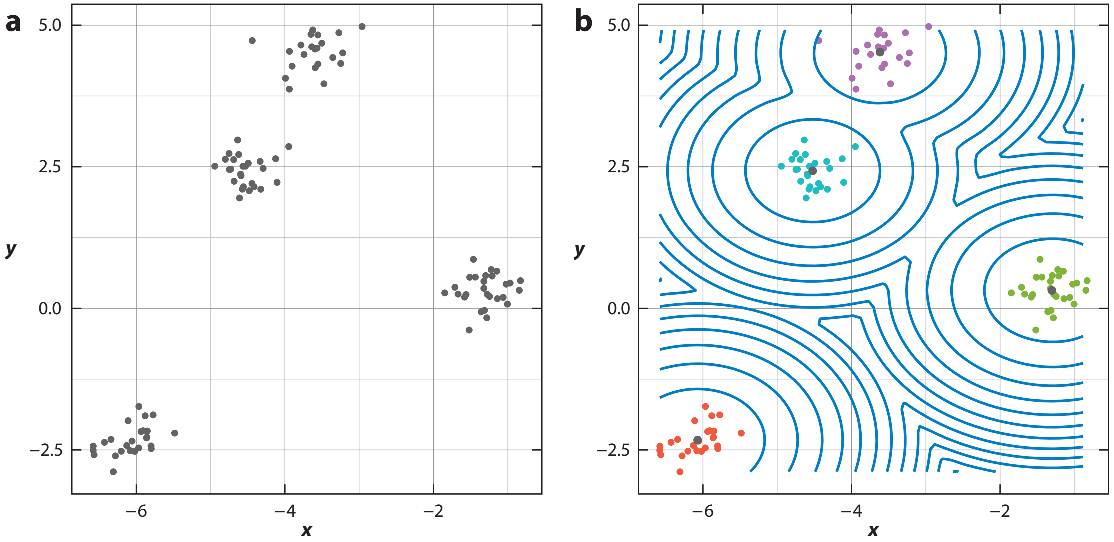


Figure 2

Example data and inference with a mixture of Gaussians. (*a*) A data set of 100 points. (*b*) 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 most likely assigned cluster, the most likely cluster means are marked in gray, and the contours give the posterior predictive distribution of the next data point.

to govern the distribution of the observations. The generative process for the Gaussian mixture model is the following:

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 n:
   a. Draw mixture assignment $z_n \mid \theta \sim \text{Discrete}(\theta)$.
   b. Draw data point $x_n \mid z_n, \mu \sim \mathcal{N}(\mu_{z_n}, 1)$.

The posterior distribution can be considered to reverse this process: Given data, what is the distribution of the hidden structure that probably generated them?

This process makes the hyperparameters explicit; they are the variance of the prior on the mixture components σ_0^2 and the Dirichlet[3] 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 variables describe hidden structure that is shared for the entire data set. The local variables are the mixture assignments; each assignment z_i helps govern only the distribution of the ith observation. This distinction becomes important in Section 4, when we discuss algorithms for approximating the posterior.

2.2. 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

[3] The Dirichlet distribution is a distribution on the simplex, nonnegative vectors that sum to one. Thus, a draw from a Dirichlet can be used as a parameter to a multinomial downstream in the model. Furthermore, note that we refer to a single draw from a multinomial (or a multinomial with $N = 1$) as a discrete distribution.

Gaussian mixture, the joint is

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

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

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

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 future data. For the mixture, the predictive distribution is

$$p(x_{\text{new}} \mid x, \sigma_0^2, \alpha) = \int \left(\sum_{z_{\text{new}}} p(z_{\text{new}} \mid \theta) p(x_{\text{new}} \mid z_{\text{new}}, \mu, \sigma_0^2) \right) p(\theta, \mu \mid x, \sigma_0^2, \alpha) \mathrm{d}\theta \mathrm{d}\mu. \tag{3.}$$

The inner sum marginalizes out the local hidden variables for the new data point, conditioned on the global hidden variables. The outer integral marginalizes out the global hidden variables, conditioned on the data. 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 known as 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 describe variational inference, a technique for approximating the posterior that emerged from the statistical machine learning community.

2.3. 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. For example, 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 in the generative process described in Section 2.1). The mixture assignment z_i depends on the mixture proportions θ, which parameterizes 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.[4] The graphical model illustrates the structure of the factorized joint distribution and the flow of the generative process.

[4] Formally, the semantics of graphical models assert that there is a possible dependence between random variables connected by an edge. Here, as a practical matter, we read the graph as asserting dependence.

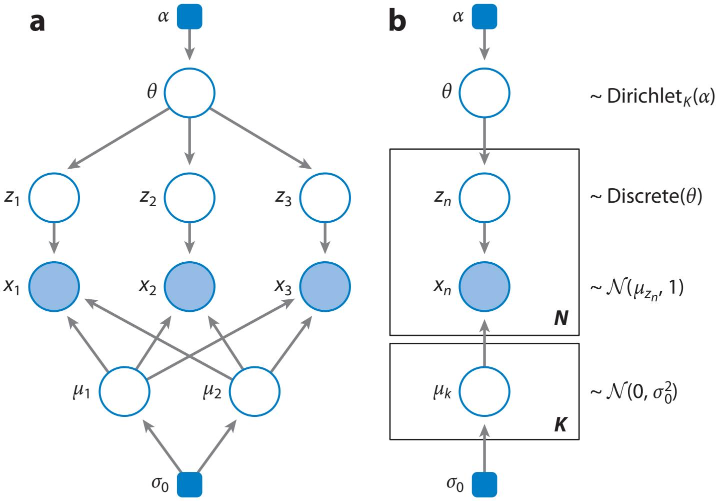

a
α
θ
z_1
z_2
z_3
x_1
x_2
x_3
μ_1
μ_2
σ_0
b
α
θ
$\sim \text{Dirichlet}_K(\alpha)$
z_n
$\sim \text{Discrete}(\theta)$
x_n
$\sim \mathcal{N}(\mu_{z_n}, 1)$
N
μ_k
$\sim \mathcal{N}(0, \sigma_0^2)$
K
σ_0

Figure 3

(*a*) A graphical model for a mixture of two Gaussians. There are three data points. The shaded nodes are observed variables, the unshaded nodes are hidden variables, and the blue square boxes are fixed hyperparameters (such as the Dirichlet parameters). (*b*) A graphical model for a mixture of *K* Gaussians with *N* data points.

Figure 3*a* illustrates a graphical model for three data points drawn from a mixture of two Gaussians. This is an unpacked model, in which 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*b*** is a more succinct graphical model for *N* data points modeled with a mixture of *K* Gaussians.

The field of graphical models provides a powerful approach to reasoning about probability distributions. 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 do not discuss graphical models in depth. Good references include Pearl (1988), Jordan (1999), Bishop (2006), Koller & Friedman (2009), and Murphy (2013).

We simply 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 (such as scientists) to build and discuss models with statisticians and computer scientists.

3. EXAMPLE MODELS

Above, we describe the basic idea behind latent variable models and provide a simple example, the Gaussian mixture model. In this section, we describe some of the commonly recurring components in latent variable models—mixed memberships, 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 were 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 until Section 4 our discussion of how to compute the posterior.

The models below are useful in and of themselves—we select 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 the field of 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 & Bishop 1999, Collins et al. 2002). Factor models are important as a component in more complicated models (such as the Kalman filter; see Section 3.4.2. below) and have generally spawned many extensions (Bartholomew et al. 2011).

In a factor model, there is a set of hidden components, 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 combine the global components with the per–data point weights. Conditioned on data, the posterior locates 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*a* 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**). Suppose the data are p-dimensional. The linear factor model contains K components $\mu_{1:K}$, each of which is a p-vector. We organize the components into a $K \times p$ matrix μ. For each data point n, we draw a K-dimensional 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 & Bishop 1999), but extensions to linear factors have considered exponential families in which $w_n^\top \mu$ is the natural parameter (Collins et al. 2002, Mohamed et al. 2008).

3.2. Mixed-Membership Models

Mixed-membership models are used for 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, in which 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

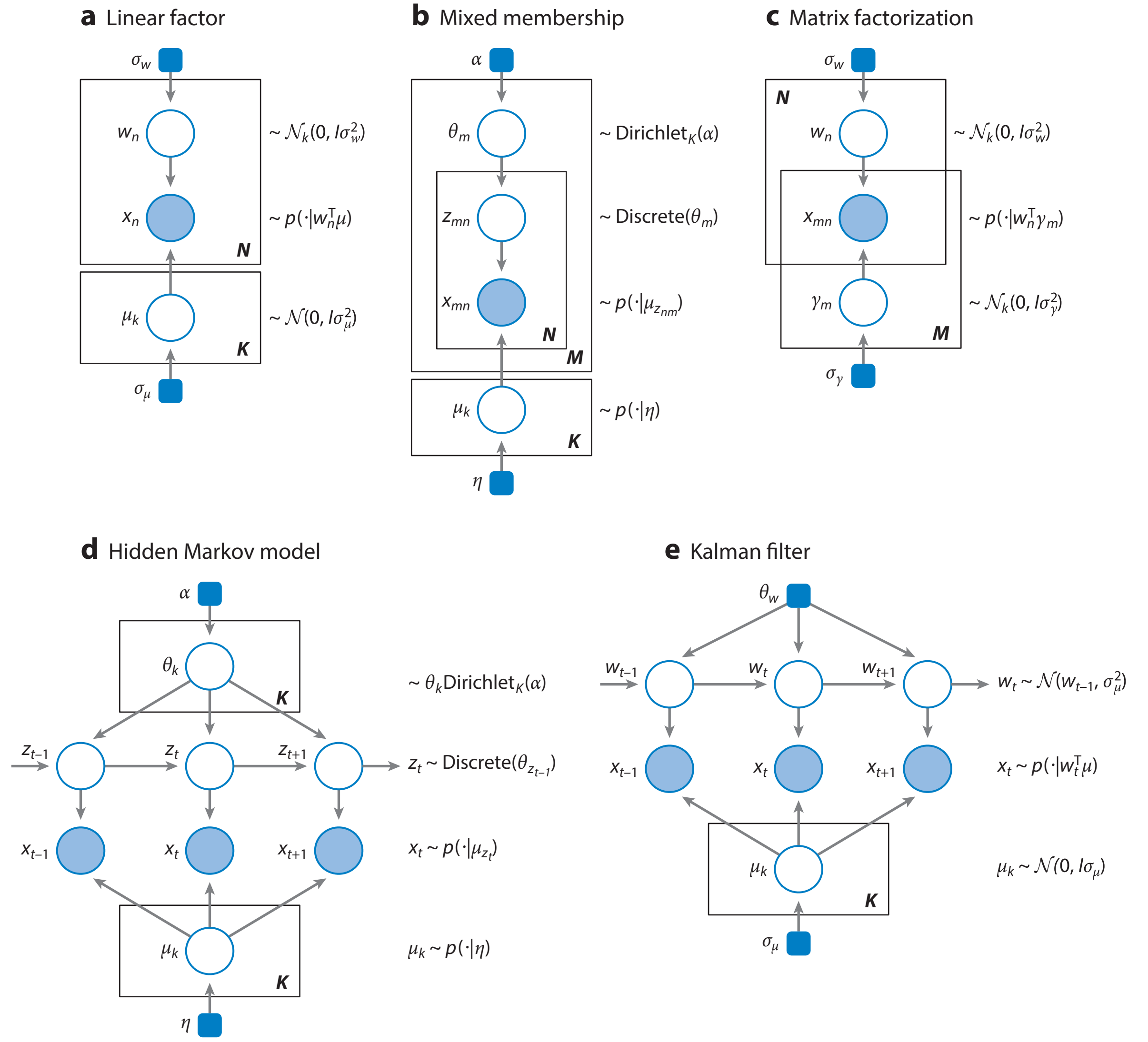

a Linear factor
σ_w
w_n
$\sim \mathcal{N}_k(0, I\sigma_w^2)$
x_n
$\sim p(\cdot|w_n^\top \mu)$
N
μ_k
$\sim \mathcal{N}(0, I\sigma_\mu^2)$
K
σ_μ
b Mixed membership
α
θ_m
$\sim \text{Dirichlet}_K(\alpha)$
z_{mn}
$\sim \text{Discrete}(\theta_m)$
x_{mn}
$\sim p(\cdot|\mu_{z_{nm}})$
N
M
μ_k
$\sim p(\cdot|\eta)$
K
η
c Matrix factorization
σ_w
N
w_n
$\sim \mathcal{N}_k(0, I\sigma_w^2)$
x_{mn}
$\sim p(\cdot|w_n^\top \gamma_m)$
γ_m
$\sim \mathcal{N}_k(0, I\sigma_\gamma^2)$
M
σ_γ
d Hidden Markov model
α
θ_k
K
$\sim \theta_k \text{Dirichlet}_K(\alpha)$
z_{t-1}
z_t
z_{t+1}
$z_t \sim \text{Discrete}(\theta_{z_{t-1}})$
x_{t-1}
x_t
x_{t+1}
$x_t \sim p(\cdot|\mu_{z_t})$
μ_k
K
$\mu_k \sim p(\cdot|\eta)$
η
e Kalman filter
θ_w
w_{t-1}
w_t
w_{t+1}
$w_t \sim \mathcal{N}(w_{t-1}, \sigma_\mu^2)$
x_{t-1}
x_t
x_{t+1}
$x_t \sim p(\cdot|w_t^\top \mu)$
μ_k
K
$\mu_k \sim \mathcal{N}(0, I\sigma_\mu)$
σ_μ

Figure 4

Graphical models for the model components described in Section 3. (*a*) Linear factor model. (*b*) Mixed-membership model. (*c*) Matrix factorization. (*d*) Hidden Markov model. (*e*) Kalman filter.

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 when we first choose a set of mixture proportions and then, for each observation in the group, choose a mixture assignment from the per-group proportions and the data point from its corresponding component. The groups share the same set of components, but each exhibits them with a different proportion. Thus, the mixed-membership posterior uncovers the recurring patterns in the data and the proportion with which they occur in each group. Contrast this situation with that of simple mixture models, in which each data group is associated with a single mixture component. Mixed-membership models give better predictions than do mixture models and provide more interesting exploratory structure.

Figure 4*b* 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.[5] 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}, each of 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—in which each document is a group of observed words—are known as topic models (Blei et al. 2003, Erosheva et al. 2004, Steyvers & 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 place their probability mass on terms that are associated under a single theme—which connects to words that tend to co-occur—and thus are termed topics. The posterior proportions indicate how each document exhibits those topics; for example, one document might be about “sports” and “health,” whereas another may be 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 1.8 million articles from the *New York Times*. This figure was made by estimating the posterior (Section 4) and then plotting the most frequent words from each topic.

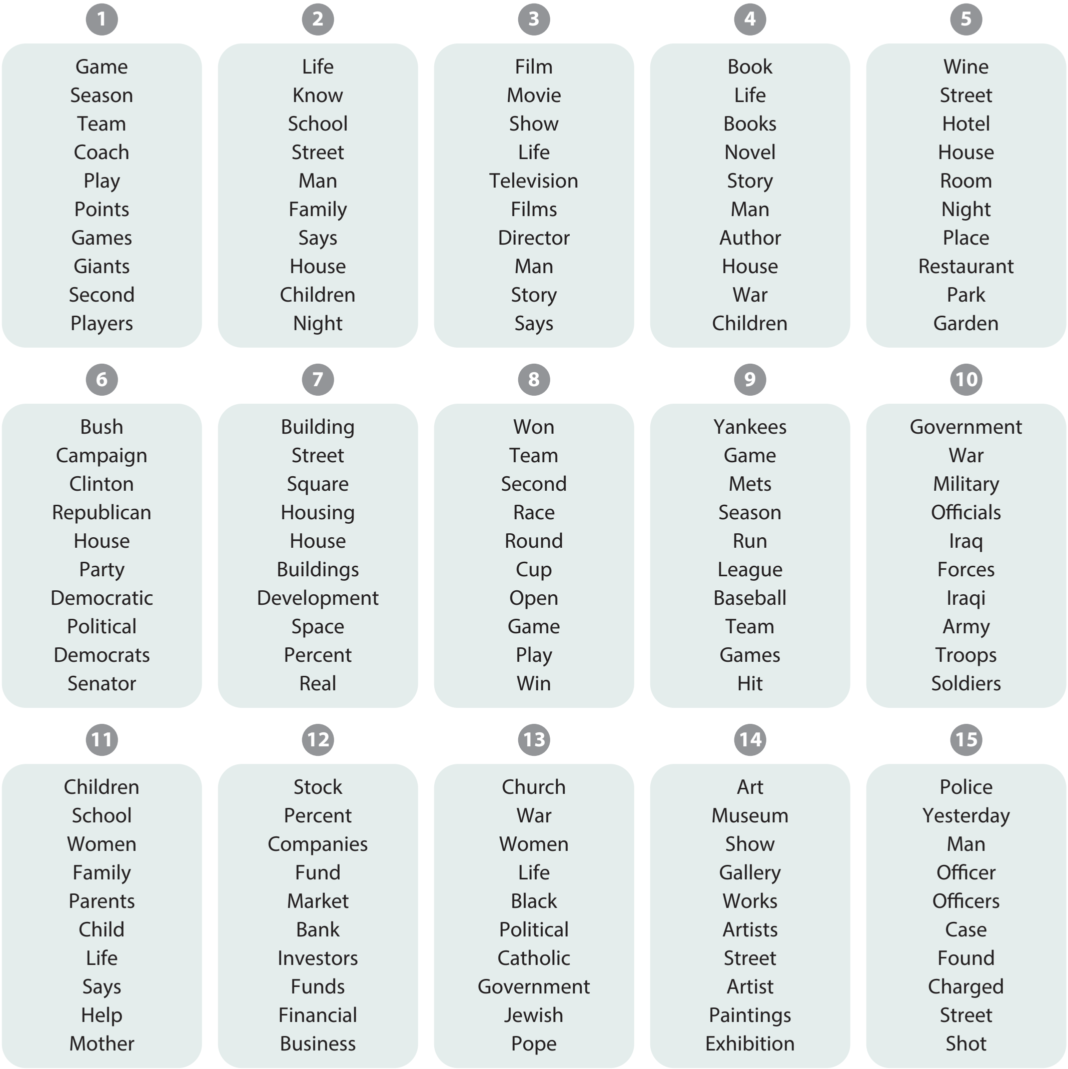
3.3. Matrix Factorization Models

Many data sets are organized into a matrix, in which 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 (Bell & Koren 2007), we observe a matrix in which 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 involves 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 the 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; for example, 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 provides low-dimensional representations of its rows and columns. These representations can be used to form predictions about unseen cells and 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

[5] Each group need not have the same number of observations, but this makes the notation cleaner.



1	2	3	4	5
Game	Life	Film	Book	Wine
Season	Know	Movie	Life	Street
Team	School	Show	Books	Hotel
Coach	Street	Life	Novel	House
Play	Man	Television	Story	Room
Points	Family	Films	Man	Night
Games	Says	Director	Author	Place
Giants	House	Man	House	Restaurant
Second	Children	Story	War	Park
Players	Night	Says	Children	Garden

6	7	8	9	10
Bush	Building	Won	Yankees	Government
Campaign	Street	Team	Game	War
Clinton	Square	Second	Mets	Military
Republican	Housing	Race	Season	Officials
House	House	Round	Run	Iraq
Party	Buildings	Cup	League	Forces
Democratic	Development	Open	Baseball	Iraqi
Political	Space	Game	Team	Army
Democrats	Percent	Play	Games	Troops
Senator	Real	Win	Hit	Soldiers

11	12	13	14	15
Children	Stock	Church	Art	Police
School	Percent	War	Museum	Yesterday
Women	Companies	Women	Show	Man
Family	Fund	Life	Gallery	Officer
Parents	Market	Black	Works	Officers
Child	Bank	Political	Artists	Case
Life	Investors	Catholic	Street	Found
Says	Funds	Government	Artist	Charged
Help	Financial	Jewish	Paintings	Street
Mother	Business	Pope	Exhibition	Shot

Figure 5

Topics found in a corpus of 1.8 million articles from the *New York Times*. Modified from Hoffman et al. (2013).

a particular movie), our prediction of the rating depends on a linear combination of the user's embedding and the 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 4c illustrates the graphical model. This model is closely related to a linear factor model, except that each cell's distribution is determined by hidden variables that depend on the cell's row and column. The overlapping plates show how the observations at the nth row share its embedding w_n but use 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

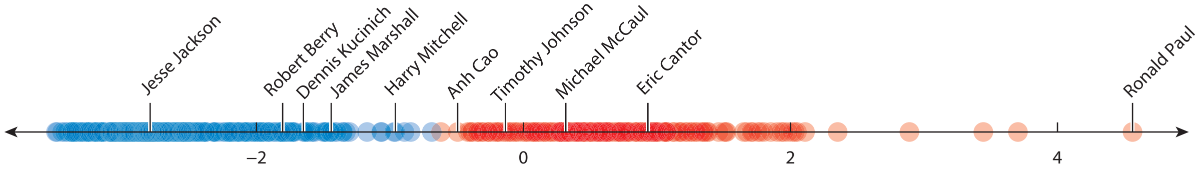


Figure 6

Ideal point models separate Republicans (*red*) from Democrats (*blue*) and try to capture the degree to which each lawmaker is liberal or conservative. Modified courtesy of Sean Gerrish.

as a general probabilistic model is known as probabilistic matrix factorization (Salakhutdinov & Mnih 2008).

Matrix factorization is widely used. In quantitative political science, ideal point models are one-dimensional matrix factorizations of legislative roll-call data (Clinton et al. 2004). **Figure 6** illustrates the one-dimensional embeddings of lawmakers in the one hundred fifteenth US Congress. The model captured the divide between Democrats and Republicans and identified a finer-grained political spectrum from liberal to conservative. Similar models are used in educational testing scenarios, in which 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). The developers of the winning approach essentially took into account Box's loop; they fitted simple matrix models first and then, as a result of insightful model criticism, embellished the basic model to consider important elements such as 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 are 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 (HMM) 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.

3.4.1. Hidden Markov models. In an HMM, each observation of the time series is drawn from an unobserved mixture component, and that component is drawn conditional on the previous observation's mixture component. The posterior is similar to a mixture model, but one in which 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 (which 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 provides an estimate of what is being said in the audio signal.

Figure 4*d* 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 a mixed-membership model) and the transition probabilities $\theta_{1:K}$. The transition probabilities provide K conditional distributions over the next component, given the value of the previous one.

3.4.2. The Kalman filter.

Whereas an HMM is a time-series adaptation of the mixture model, a 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, a Gaussian whose mean is the previous position's embedding. See West & Harrison (1997) for a general probabilistic perspective on continuous time series.

Kalman filters are influential in radar tracking applications, in which w_t is a vector that 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 object's true position and predict its next position.

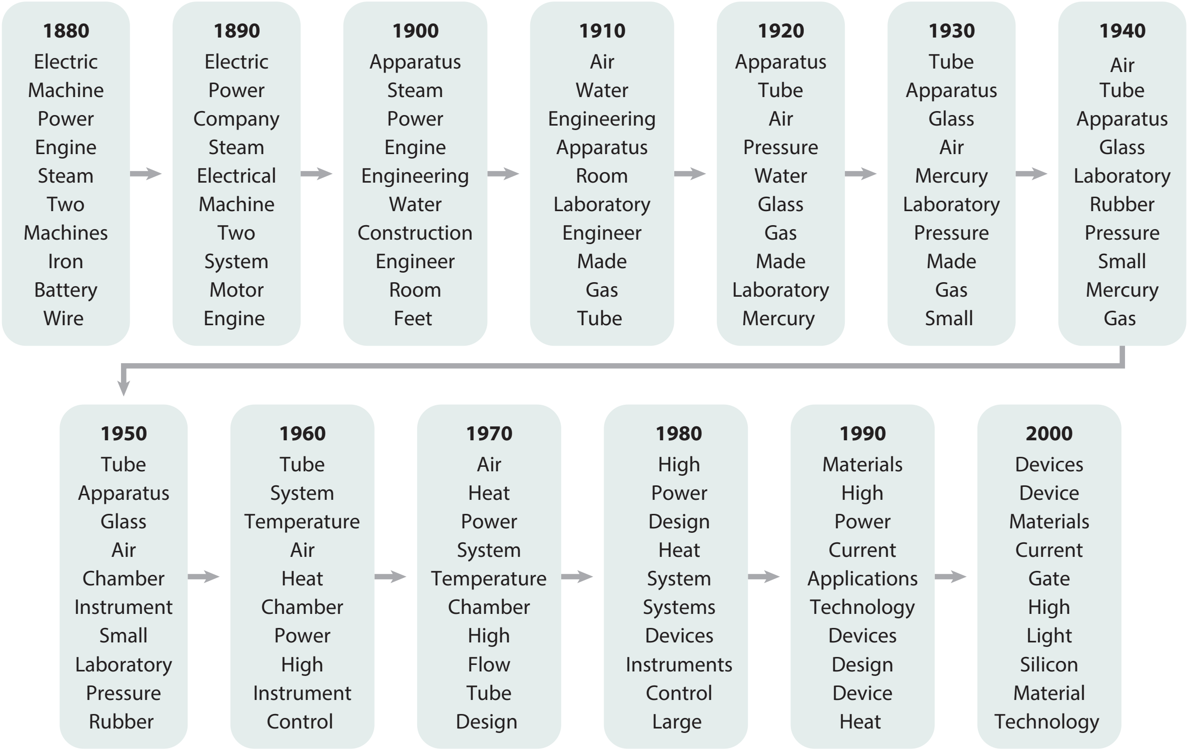
Figure 4*e* illustrates the graphical model. Although the underlying distributions are different—in the Kalman filter the hidden chain of variables is a sequence of continuous variables, rather than discrete ones—the structure of the graphical model is nearly the same as for the HMM. Although the algorithms for the HMM and the Kalman filter were developed independently for different purposes and in different research communities, they are both instances of general algorithms for graphical models. Finding such connections, and developing diverse applications with new models, is one of the advantages of the graphical model formalism.

3.5. The Craft of Latent Variable Modeling

Above, we describe a selection of latent variable models, each of which builds on simpler models. However, we emphasize that our goal is not to deliver a complete catalog of models. In fact, we omit some important types of models, such as Bayesian nonparametric models (Ferguson 1973, Antoniak 1974, Teh & Jordan 2008, Hjort et al. 2010, Gershman & 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 covariates, and hierarchical models (Gelman & 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, the observation is drawn from 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 positive data, or discrete distributions for categorical data. We can even use conditional models, such as generalized linear models (Nelder & Wedderburn 1972, McCullagh & 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. Doing so fundamentally changes models such as Gaussian matrix factorization to a form of nonnegative matrix factorization, a technique that is important in computer vision problems (Lee & Seung



1880	1890	1900	1910	1920	1930	1940
Electric	Electric	Apparatus	Air	Apparatus	Tube	Air
Machine	Power	Steam	Water	Tube	Apparatus	Tube
Power	Company	Power	Engineering	Air	Glass	Apparatus
Engine	Steam	Engine	Apparatus	Pressure	Air	Glass
Steam	Electrical	Engineering	Room	Water	Mercury	Laboratory
Two	Machine	Water	Laboratory	Glass	Laboratory	Rubber
Machines	Two	Construction	Engineer	Gas	Pressure	Pressure
Iron	System	Engineer	Made	Made	Made	Small
Battery	Motor	Room	Gas	Laboratory	Gas	Mercury
Wire	Engine	Feet	Tube	Mercury	Small	Gas

1950	1960	1970	1980	1990	2000
Tube	Tube	Air	High	Materials	Devices
Apparatus	System	Heat	Power	High	Device
Glass	Temperature	Power	Design	Power	Materials
Air	Air	System	Heat	Current	Current
Chamber	Heat	Temperature	System	Applications	Gate
Instrument	Chamber	Chamber	Systems	Technology	High
Small	Power	High	Devices	Devices	Light
Laboratory	High	Flow	Instruments	Design	Silicon
Pressure	Instrument	Tube	Control	Device	Material
Rubber	Control	Design	Large	Heat	Technology

Figure 7

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

1999). Alternatively, we might replace simple distributions with more complex distributions that themselves have latent structure. For example, the idea behind so-called spike and slab modeling (Ishwaran & 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, and second, generate the weights of those factors from an appropriate distribution. This process provides 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 & 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 the magazine *Science*. 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—they find the themes that underlie the collection and reveal how those themes change smoothly through time.

As another example, McAuliffe et al. (2004) and Siepel & Haussler (2004) combined tree-based models and HMMs 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 in the collection.

We emphasize that we have only scratched the surface of the types 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 in probabilistic models, data analysts are able to both formally express the type of latent structure they want to uncover and devise the generative assumptions that can capture it.

4. POSTERIOR INFERENCE WITH MEAN FIELD VARIATIONAL METHODS

In Sections 2 and 3, we describe the probabilistic modeling formalism, discuss several latent variable models, and show 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 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 Markov chain Monte Carlo (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). The reader is referred to Smola et al. (2003) and Rue et al. (2009) for recent innovations.

MCMC sampling methods (Robert & Casella 2004) include fundamental algorithms such as Metropolis–Hastings (Metropolis et al. 1953, Hastings 1970) and Gibbs sampling (Geman & Geman 1984, Gelfand & Smith 1990). Such 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 discuss a particular strategy, mean field variational inference. Variational inference is a deterministic alternative to MCMC in which sampling is replaced by optimization (Jordan et al. 1999, Wainwright & 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 & 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. Mean field variational inference provides a generic way to approximate the posterior for many models.[6]

[6] This section is more mathematically dense and abstract than the rest of this review. 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.

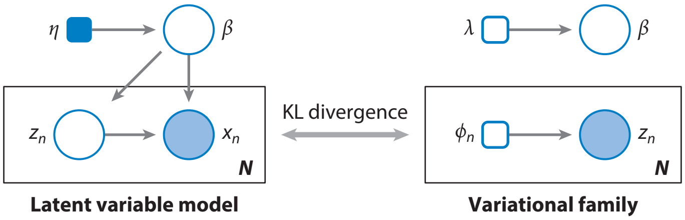


Figure 8

(*Left*) The general graph structure for a conditionally conjugate model with N observations. (*Right*) 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 Kullback–Leibler (KL) divergence to the model posterior.

4.1. Conditionally Conjugate Models

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

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

Figure 8 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. The graphical model in **Figure 8** reveals that local variables are inside the data plate and that global variables are outside the data plate.

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

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

As 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 in computing the posterior stems from 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. If the distribution of x is in an exponential family, then its density has the form

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

The function $t(x)$ is the sufficient statistic; the function $h(x)$ is the base measure; and 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).

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

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

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 use the conditional independencies of the local variables; although they are implicitly conditioned on all the observations and other latent variables, the local variables for the nth data context depend only on the nth data point and global variables.

Requiring complete conditionals in the exponential family is less stringent than requiring full conjugacy (which has given rise to the moniker conditional conjugacy). In contrast, consider the classical Bayesian modeling setting, in which a latent variable is used as a parameter to the observations. The model is fully conjugate (as opposed to conditionally conjugate) 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 & Tiao 1973, Bernardo & 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 remains 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, which 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 types of models, including numerous 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 allow us to quickly build, use, and revise sophisticated latent variable models in many data-analysis settings.

4.2. Mean Field Variational Inference

In the preceding section, we define 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 & Hinton 1999), building on previous work in statistical physics (Peterson & Anderson 1987). The idea is to posit a family of distributions over the latent variables with free parameters (termed variational parameters) and then fit those parameters to find the member of the family that is close to the posterior; closeness is measured by Kullback–Leibler (KL) divergence (Kullback & Leibler 1951).

In the following subsections, we 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 & Beal 2001; Xing et al. 2003). For good reviews of variational inference in general, see Jordan et al. (1999) and Wainwright & Jordan (2008). Here, we follow the treatment in Hoffman et al. (2013).

4.2.1. 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 below.) The goal of variational inference is to find the optimal variational parameters by solving

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

It is through this objective that we tie the variational parameters ν to the observations x (**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:

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

where all expectations are taken with respect to the variational distribution. 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 place mass on latent variable configurations that make the data likely; the second term, which is the entropy of the variational distribution, values diffuse variational distributions.

4.2.2. The mean field variational family. Before optimizing the objective, we must specify the variational family in more detail. We use the mean field variational family, in which 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

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

Note that each variable is independent but that the variables are not identically distributed. Although it cannot capture correlations between variables, this family is very flexible and can focus its mass on any complex configuration of them. We note that the data do not appear in Equation 11; data connect to the variational parameters only when we optimize the variational objective.

To complete the specification, we set each variational factor to be in the same family as the corresponding complete conditional in the model (Equations 7 and 8). If $p(\beta \mid x, z)$ is a Gaussian, then λ are free Gaussian parameters; if $p(\phi_n \mid x_n, z)$ is discrete over K elements, then ϕ_n is a free distribution over K elements. We note that although we assume this structure, Bishop (2006) shows that the optimal mean field variational distribution (Equation 11) will necessarily be in this family.

4.2.3. 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. Again, we emphasize 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:

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

where this expectation is taken with respect to the variational distribution. The reader is referred to 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 understand this point, 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 $\eta_g(z, x)$ involves only 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 depends only 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 proceeds 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. We monitor convergence by tracking the relative change in the variational objective of Equation 10. In practice, one uses multiple random restarts to find a good local optimum.

Note that coordinate-ascent variational inference relates closely to the expectation-maximization algorithm of Dempster et al. (1977). Both are coordinate ascent algorithms that iterate between per–data point computations and data set–wide computations. Furthermore, the EM objective is derived from Jensen's inequality in the same way as the variational objective.

We return briefly to the Gaussian mixture model. The latent variables are the mixture assignments $z_{1:N}$ and the 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 reestimate the locations of the mixture components. This procedure converges to an approximate posterior for the full mixture model. Note that this is how we performed inference on the simulated data of **Figure 2**.

4.2.4. Building on mean field inference. The coordinate ascent inference algorithm described above is the simplest variational inference algorithm. Developing faster, more accurate, and more sophisticated variational inference methods are active areas of machine learning and statistics research. Structured variational inference (Saul & Jordan 1996) relaxes the mean field assumption;

the variational distribution allows some dependencies among the variables. Nonconjugate variational inference (Knowles & Minka 2011, Wang & Blei 2013) relaxes the assumption that the complete conditionals are in the exponential family. These algorithms generalize previous research on nonconjugate models, which required specialized approximations for the particular model at hand. In general, variational methods are a powerful tool for approximate posterior inference in complex models.

5. MODEL CRITICISM

With the tools described in 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 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, that is, divide it into groups and structures dictated by the inferences. Examples of exploratory tasks include using topic models to navigate large collections of documents and using clustering models on microarray data to suggest groups of related genes.

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

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

(Equation 3 gives the predictive distribution for a mixture model.) Usually, the posterior $p(\beta \mid x)$ is not available, so we substitute an approximation, $q(\beta)$, which we find by an approximate inference algorithm such as MCMC or variational inference. Examples of predictive tasks include using a matrix factorization to predict which items a user will purchase or using a time-series model to predict future stock prices on the basis of their histories.

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 assess it for the task at hand—for example, to deploy a recommendation system on the Web, trade on predictions about the stock market, or 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 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 misspecified 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 & Hjort 2008), which is 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 allowed for developing only a single model, it would still be useful to know how and where it succeeds and fails. Second, using a model for a chosen task always involves using the posterior, or the approximate posterior. This approximate posterior is a function of both the model and the chosen inference algorithm. Thus, 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 & Raftery 1995, MacKay 2003) to determine which model more likely generated the data.

Model criticism, however, views model formulation as part of the iterative process of Box's loop. Criticism techniques seek to show how a model falls short and indicate where we should make it more complex. In model criticism, 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 & Shalizi 2012). Observations that do not match the predictions of a theory are evidence against it and point us in the directions of improvement. Finally, note that model criticism can be a philosophically controversial idea because it is incoherent under certain theoretical frameworks [see the discussion of Box (1980) and the comments in Lauritzen (2007)].

5.1. Predictive Sample Reuse

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 reuse (PSR), which amounts to using cross-validation to estimate this probability. [Cross-validation was coinvented by Geisser (1975), independently of Stone (1974).]

Let $x_{[n]}$ be the data set with the nth item removed. Suppose our model is $p(\beta, z, x)$ and we use 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

$$\ell_n = \log p(x_n \mid x_{[n]}) \qquad 16.$$

$$= \log \int \left(\int p(x_n \mid z_n) q(z_n) \mathrm{d}z_n \right) q_{[n]}(\beta) \mathrm{d}\beta. \qquad 17.$$

This is the (approximate) posterior predictive probability of the nth data point using data that do 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 with the rest of the data; finally, we compute ℓ_n for each data point, integrating over the approximate posterior estimated from data that do not contain the nth point.

One advantage of PSR is that it does not simply help evaluate the modeling assumptions 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 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 & 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. PSR easily lets us compare two different approximations of the same predictive distribution.

A further advantage of PSR is that it 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 & 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) 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 that our observed data are not typical—that is, they have 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 compromises—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 with respect to 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. These new data come 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 does not look like the observations—the data drawn from the distribution that we are hoping the model will capture—then there may be 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. [The function $T(X)$ is also called a test statistic.] Let x^{rep} be a random set of new hypothetical future observations, a data set drawn from the posterior predictive distribution. Then,

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

Note, the only random variable in this expression is x^{rep}. This PPC is the probability that the replicated data are far away (in terms of T) from the observations.

An important development from 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.) Then,

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

Here, both the hidden variables β and the 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:

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

Because it is written as an interval, this equation reveals how to estimate the PPC 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 the originally observed data $T(x, \beta^{(t)})$.

The posterior predictive p-value is the proportion of cases in which $T(x^{(t)}, \beta^{(t)}) > T(x, \beta^{(t)})$. 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:

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

We sample from the posterior, replicate a data set from the sample, and compute the average log probability of both the observed data and the replicated data. In spirit, this process leads to a check similar to classical goodness-of-fit tests of residuals (Cook & 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 methodology is that it is adaptable to the particular needs of the practitioner. For example, the 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 toward the types of observations that are important to model well in the application at hand. Furthermore, we can look at multiple discrepancies, even reusing computation to do so, to understand trade-offs involved among various models.

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 examples.

The presence of certain emission lines in astrophysical spectra gives clues as to the chemical composition of stars. Excess electromagnetic radiation in a narrow energy range is evidence for these lines. Such evidence is generally sparse, however, and whether it is "significant" is important to forming new astrophysics theories and designing subsequent experiments. Van Dyk & Kang (2004) used PPCs on complex spectral models—with a likelihood ratio as the discrepancy—to determine the existence of emission lines.

Belin & 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 also focus on how PPCs can be used with imputed data, local hidden variables (in the parlance of this review) that are filled in by the model, to directly estimate discrepancies otherwise confounded by missingness.

Belin & Rubin (1995) use the PPC methodology to implement several iterations of Box's loop. They analyze data from an experiment that collected response times by schizophrenics and nonschizophrenics 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. Note that 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, these authors 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 modified from their

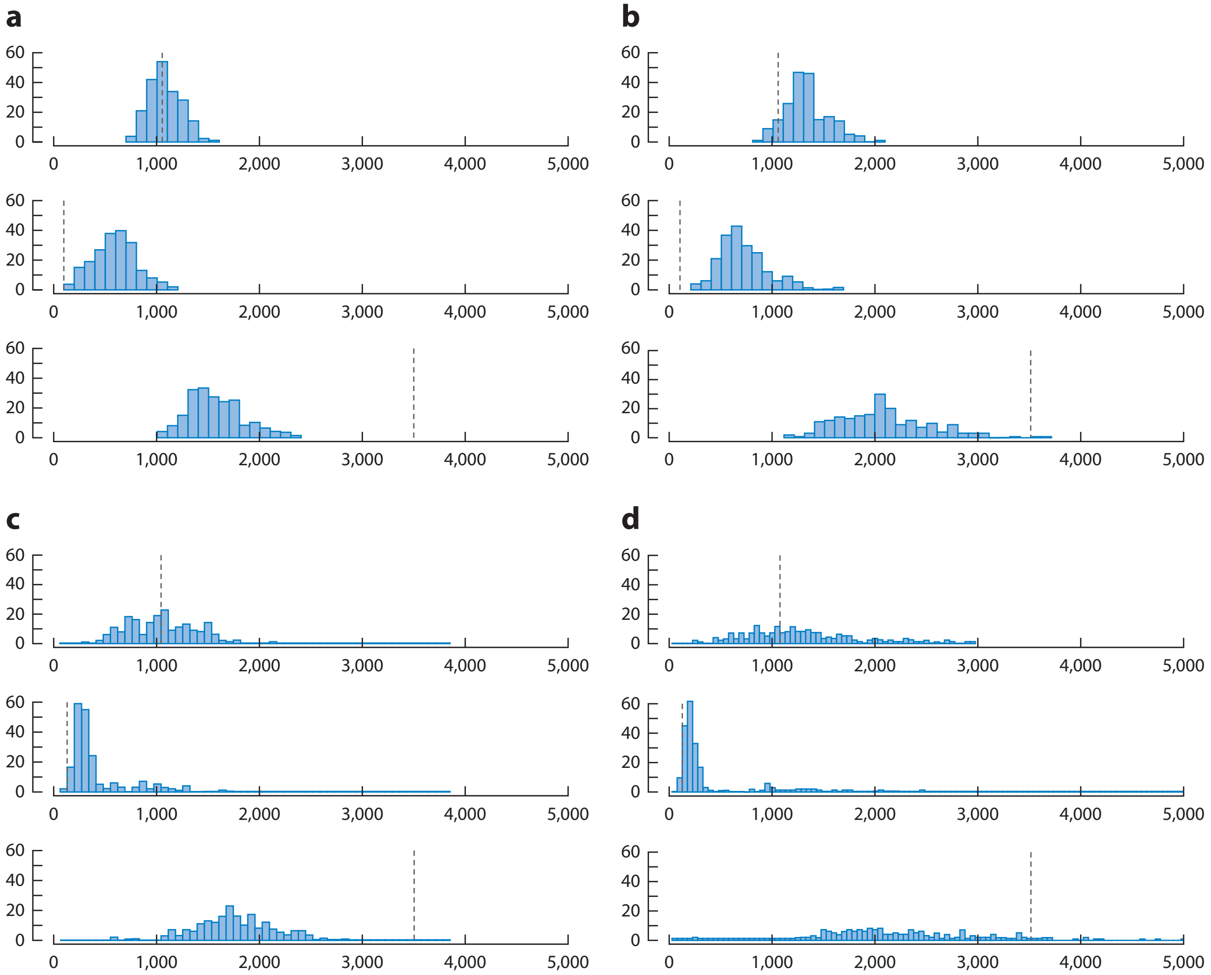


Figure 9

Posterior predictive checks for the four models developed by Belin & Rubin (1995), in the order the authors build them. Each panel contains the checks for one model, and each check is a histogram of a discrepancy applied to replicated data sets. The three discrepancies in each panel are the largest observed variance for schizophrenics (*top*), the smallest observed variance for schizophrenics (*middle*), and the within-person variance across all subjects (*bottom*). The dashed line in each plot indicates the discrepancy of the observed data. (Note that this discrepancy does not depend on the model under study.) Modified from figure 2 of Belin & Rubin (1995).

article. It is clear that the sequence of models provides increasingly better predictive distributions of the observed data.

As a final example, we summarize our research on PPCs for topic models (Mimno & 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 conditional mutual information between document labels and observations. This pair of variables should have low mutual information because, under the modeling assumptions, they are conditionally independent. This research 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 can use PPCs to filter out aspects of the model that are not capturing the data well and to 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—both methodological and technological—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. First, 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. A promising avenue of research in this vein is probabilistic programming. Researchers in probabilistic programming are developing software that allows users to easily specify models and compute approximate posteriors (Gilks & Spiegelhalter 1992, Bishop et al. 2003, McCallum et al. 2009, Stan Development Team 2013; also see **http://research.microsoft.com/infernet**). 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 recently become increasingly important because scientists and other data consumers want to discover, understand, and exploit patterns in streams of observational data. (Such goals are outside the traditional scientific goal of drawing conclusions from a designed experiment.) In this spirit, we should continue to develop methods such as PPCs to help navigate large data sets with complex models. We should further develop the foundations of data exploration, along the lines of Tukey (1962), Good (1983), and Diaconis (1985), to develop principled methods for exploring observational data.

However, the most important challenges are those we cannot yet identify. The limitations of our methods will be revealed only 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, direct us to where and how our methods can be improved.

DISCLOSURE STATEMENT

This material is partly based on research sponsored by DARPA under agreement number FA8750-14-2-0009. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA or the U.S. Government.

ACKNOWLEDGMENTS

My work is supported by National Science Foundation (NSF) CAREER grant IIS-0745520, NSF BIGDATA grant IIS-1247664, NSF NEURO grant IIS-1009542, Office of Naval Research grant N00014-11-1-0651, and the Alfred P. Sloan foundation. The author is grateful to Martha Benco, Steve Fienberg, Andrew Gelman, Sean Gerrish, Jeremy Manning, Xiao-Li Meng, David Mimno, Donald Rubin, Hal Stern, Chris Wiggins, and Chong Wang for useful comments and discussions.

LITERATURE CITED

Airoldi EM, Blei DM, Fienberg SE, Xing EP. 2008. Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.* 9:1981–2014

Antoniak CE. 1974. Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *Ann. Stat.* 2:1152–74

Asuncion A, Welling M, Smyth P, Teh YW. 2009. On smoothing and inference for topic models. *Proc. Conf. Uncertain. Artif. Intell.* 25:27–34

Attias H. 1999. Inferring parameters and structure of latent variable models by variational Bayes. *Proc. Conf. Uncertain. Artif. Intell.* 15:21–30

Attias H. 2000. A variational Bayesian framework for graphical models. *Adv. Neural Inf. Process. Syst.* 12:209–15

Baker FB. 1992. *Item Response Theory*. New York: Marcel Dekker

Bar-Shalom Y, Li XR, Kirubarajan T. 2004. *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. New York: Wiley

Bartholomew DJ, Knott M, Moustaki I. 2011. *Latent Variable Models and Factor Analysis*, Vol. 899: *A Unified Approach*. New York: Wiley

Belin TR, Rubin DB. 1995. The analysis of repeated-measures data on schizophrenic reaction times using mixture models. *Stat. Med.* 14:747–68

Bell RM, Koren Y. 2007. Lessons from the Netflix prize challenge. *ACM SIGKDD Explor. Newsl.* 9:75–79

Bernardo JM, Smith AFM. 1994. *Bayesian Theory*. Chichester, UK: Wiley

Bishop CM. 2006. *Pattern Recognition and Machine Learning*. New York: Springer

Bishop CM. 2013. Model-based machine learning. *Philos. Trans. R. Soc. A* 371:20120222

Bishop CM, Spiegelhalter D, Winn J. 2003. VIBES: a variational inference engine for Bayesian networks. *Adv. Neural Inf. Process. Syst.* 15:793–800

Blei DM. 2012. Probabilistic topic models. *Commun. ACM* 55:77–84

Blei DM, Lafferty JD. 2006. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, ed. W Cohen, A Moore, pp. 113–20. New York: Assoc. Comput. Mach.

Blei DM, Lafferty JD. 2007. A correlated topic model of science. *Ann. Appl. Stat.* 1:17–35

Blei DM, Ng AY, Jordan MI. 2003. Latent Dirichlet allocation. *J. Mach. Learn. Res.* 3:993–1022

Box GEP. 1976. Science and statistics. *J. Am. Stat. Assoc.* 71:791–99

Box GEP. 1980. Sampling and Bayes' inference in scientific modelling and robustness. *J. R. Stat. Soc. A* 143:383–430

Box GEP, Draper NR. 1987. *Empirical Model-Building and Response Surfaces*. New York: Wiley

Box GEP, Hill WJ. 1967. Discrimination among mechanistic models. *Technometrics* 9:57–71

Box GEP, Hunter WG. 1962. A useful method for model-building. *Technometrics* 4:301–18

Box GEP, Hunter WG. 1965. The experimental study of physical mechanisms. *Technometrics* 7:23–42

Box GEP, Tiao GC. 1973. *Bayesian Inference in Statistical Analysis*. New York: Wiley

Brown LD. 1986. *Fundamentals of Statistical Exponential Families with Applications in Statistical Decision Theory*. Hayward, CA: Inst. Math. Stat.

Claeskens G, Hjort NL. 2008. *Model Selection and Model Averaging*. New York: Cambridge Univ. Press

Clinton J, Jackman S, Rivers D. 2004. The statistical analysis of roll call data. *Am. Polit. Sci. Rev.* 98:355–70

Collins M, Dasgupta S, Schapire R. 2002. A generalization of principal component analysis to the exponential family. *Adv. Neural Inf. Process. Syst.* 14:617–24

Cook RD, Weisberg S. 1982. *Residuals and Influence in Regression*. London: Chapman & Hall

Dawid AP, Lauritzen SL. 1993. Hyper Markov laws in the statistical analysis of decomposable graphical models. *Ann. Stat.* 21:1272–317

Dempster AP, Laird NM, Rubin DB. 1977. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. B.* 36:1–38

Diaconis P. 1985. Theories of data analysis: from magical thinking through classical statistics. In *Exploring Data: Tables, Trends, and Shapes*, ed. DC Hoaglin, F Mosteller, JW Tukey, pp. 1–36. New York: Wiley

Durbin R, Eddy SR, Krogh A, Mitchison G. 1998. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. New York: Cambridge Univ. Press

Efron B. 2013. *Empirical Bayes modeling, computation, and accuracy*. Tech. rep. 263, Div. Biostat., Stanford Univ., Stanford, CA. **http://statweb.stanford.edu/~ckirby/brad/papers/2013EBModeling.pdf**

Efron B, Morris C. 1973. Combining possibly related estimation problems. *J. R. Stat. Soc. B* 35:379–421

Erosheva E, Fienberg S, Lafferty J. 2004. Mixed-membership models of scientific publications. *Proc. Natl. Acad. Sci. USA* 101(Suppl. 1):5220–27

Erosheva EA, Fienberg SE, Joutard C. 2007. Describing disability through individual-level mixture models for multivariate binary data. *Ann. Appl. Stat.* 1:502–37

Ferguson TS. 1973. A Bayesian analysis of some nonparametric problems. *Ann. Stat.* 1:209–30

Geisser S. 1975. The predictive sample reuse method with applications. *J. Am. Stat. Assoc.* 70:320–28

Gelfand AE, Smith AFM. 1990. Sampling-based approaches to calculating marginal densities. *J. Am. Stat. Assoc.* 85:398–409

Gelman A, Carlin JB, Stern HS, Rubin DB. 1995. *Bayesian Data Analysis*. London: Chapman & Hall

Gelman A, Hill J. 2007. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. New York: Cambridge Univ. Press

Gelman A, Meng X-L, Stern H. 1996. Posterior predictive assessment of model fitness via realized discrepancies. *Stat. Sin.* 6:733–807

Gelman A, Shalizi CR. 2012. Philosophy and the practice of Bayesian statistics. *Br. J. Math. Stat. Psychol.* 66:8–38

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

Geman S, Geman D. 1984. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Mach. Intell.* 6:721–41

Gershman SJ, Blei DM. 2012. A tutorial on Bayesian nonparametric models. *J. Math. Psychol.* 56:1–12

Ghahramani Z. 2012. Bayesian nonparametrics and the probabilistic approach to modelling. *Philos. Trans. R. Soc. A* 371:1984

Ghahramani Z, Beal MJ. 2001. Propagation algorithms for variational Bayesian learning. *Adv. Neural Inf. Process. Syst.* 13:507–13

Gilks WR, Thomas A, Spiegelhalter DJ. 1992. A language and program for complex Bayesian modelling. *Statistician* 43:169–77

Good IJ. 1983. The philosophy of exploratory data analysis. *Philos. Sci.* 50:283–95

Good IJ. 2009 (1983). Subjective probability as the measure of a non-measurable set. In *Good Thinking: The Foundations of Probability and Its Applications*, pp. 73–82. Mineola, NY: Dover

Guttman I. 1967. The use of the concept of a future observation in goodness-of-fit problems. *J. R. Stat. Soc. B* 29:83–100

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

Hjort NL, Holmes C, Müller P, Walker SG, eds. 2010. *Bayesian Nonparametrics*. New York: Cambridge Univ. Press

Hoffman MD, Blei DM, Wang C, Paisley J. 2013. Stochastic variational inference. *J. Mach. Learn. Res.* 14:1303–47

Hotelling H. 1933. Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol.* 24:417–41

Hotelling H. 1936. Relations between two sets of variates. *Biometrika* 28:321–77

Ishwaran H, Rao JS. 2005. Spike and slab variable selection: frequentist and Bayesian strategies. *Ann. Stat.* 33:730–73

Jordan MI, ed. 1999. *Learning in Graphical Models*. Cambridge, MA: MIT Press

Jordan MI. 2004. Graphical models. *Stat. Sci.* 19:140–55

Jordan MI, Ghahramani Z, Jaakkola TS, Saul LK. 1999. An introduction to variational methods for graphical models. *Mach. Learn.* 37:183–233

Kalman RE. 1960. A new approach to linear filtering and prediction problems. *Trans. ASME J. Basic Eng.* 82:35–45

Kass RE, Raftery AE. 1995. Bayes factors. *J. Am. Stat. Assoc.* 90:773–95

Knowles DA, Minka TP. 2011. Non-conjugate variational message passing for multinomial and binary regression. *Adv. Neural Inf. Process. Syst.* 24:1701–9

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

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

Krnjajić M, Kottas A, Draper D. 2008. Parametric and nonparametric Bayesian model specification: a case study involving models for count data. *Comput. Stat. Data Anal.* 52:2110–28

Kullback S, Leibler RA. 1951. On information and sufficiency. *Ann. Math. Stat.* 22:79–86

Lauritzen SL. 2007. Discussion of some aspects of model selection for prediction: article of Chakrabarti and Ghosh. In *Bayesian Statistics 8*, ed. JM Bernardo, MJ Bayarri, JO Berger, AP David, D Heckerman, et al., pp. 84–90. Oxford, UK: Oxford Univ. Press

Lee DD, Seung HS. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature* 401:788–91

Lehmann EL. 1990. Model specification: the views of Fisher and Neyman, and later developments. *Stat. Sci.* 5(2):160–68

MacKay DJC. 2003. *Information Theory, Inference, and Learning Algorithms*. New York: Cambridge Univ. Press

McAuliffe JD, Pachter L, Jordan MI. 2004. Multiple-sequence functional annotation and the generalized hidden Markov phylogeny. *Bioinformatics* 20:1850–60

McCallum A, Schultz K, Singh S. 2009. FACTORIE: probabilistic programming via imperatively defined factor graphs. *Adv. Neural Inf. Process. Syst.* 22:1249–57

McCullagh P, Nelder JA. 1989. *Generalized Linear Models*. London: Chapman & Hall

Meng X-L. 1994. Posterior predictive *p*-values. *Ann. Stat.* 22:1142–60

Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. 1953. Equations of state calculations by fast computing machines. *J. Chem. Phys.* 21:1087–92

Mimno D, Blei DM. 2011. Bayesian checking for topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, ed. P Merlo, pp. 227–37. Stroudsburg, PA: Assoc. Comput. Linguist.

Mohamed S, Ghahramani Z, Heller K. 2008. Bayesian exponential family PCA. *Adv. Neural Inf. Process. Syst.* 21:1089–96

Morris CN. 1983. Parametric empirical Bayes inference: theory and applications. *J. Am. Stat. Assoc.* 78:47–65

Murphy KP. 2013. *Machine Learning: A Probabilistic Approach*. Cambridge, MA: MIT Press

Neal RM, Hinton GE. 1999. A view of the EM algorithm that justifies incremental, sparse, and other variants. In *Learning in Graphical Models*, ed. MI Jordan, pp. 355–68. Cambridge, MA: MIT Press

Nelder JA, Wedderburn RWM. 1972. Generalized linear models. *J. R. Stat. Soc. A* 135:370–84

Pearl J. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. New York: Morgan Kaufmann Publ.

Pearson K. 1901. On lines and planes of closest fit to systems of points. *Lond. Edinb. Dublin Philos. Mag. J. Sci.* 6:559–72

Peterson C, Anderson JR. 1987. A mean field theory learning algorithm for neural networks. *Complex Syst.* 1:995–1019

Popper KR. 1959. *The Logic of Scientific Discovery*. London: Hutchinson

Pritchard JK, Stephens M, Donnelly P. 2000. Inference of population structure using multilocus genotype data. *Genetics* 155:945–59

Rabiner LR. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* 77:257–86

Robbins H. 1980. An empirical Bayes estimation problem. *Proc. Natl. Acad. Sci. USA* 77:6988–89

Robbins H, Monro S. 1951. A stochastic approximation method. *Ann. Math. Stat.* 22:400–7

Robert CP, Casella G. 2004. *Monte Carlo Statistical Methods*. New York: Springer

Roweis S. 1998. EM algorithms for PCA and SPCA. *Adv. Neural Inf. Process. Syst.* 10:626–32

Rubin DB. 1984. Bayesianly justifiable and relevant frequency calculations for the applied statistician. *Ann. Stat.* 12:1151–72

Rue H, Martino S, Chopin N. 2009. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *J. R. Stat. Soc. B* 71:319–92

Salakhutdinov R, Mnih A. 2008. Probabilistic matrix factorization. *Adv. Neural Inf. Process. Syst.* 20:1257–64

Saul LK, Jordan MI. 1996. Exploiting tractable substructures in intractable networks. *Adv. Neural Inf. Process. Syst.* 8:486–92

Siepel A, Haussler D. 2004. Combining phylogenetic and hidden Markov models in biosequence analysis. *J. Comput. Biol.* 11:413–28

Skrondal A, Rabe-Hesketh S. 2007. Latent variable modelling: a survey. *Scand. J. Stat.* 34:712–45

Smola AJ, Vishwanathan SVN, Eskin E. 2003. Laplace propagation. *Adv. Neural Inf. Process. Syst.* 16:441–48

Stan Development Team. 2013. *Stan: A C++ Library for Probability and Sampling*, *Version 1.3*. **http://mc-stan.org**

Steyvers M, Griffiths T. 2006. Probabilistic topic models. In *Latent Semantic Analysis: A Road to Meaning*, ed. T Landauer, D McNamara, S Dennis, W Kintsch, pp. 424–40. London: Laurence Erlbaum

Stone M. 1974. Cross-validatory choice and assessment of statistical predictions. *J. R. Stat. Soc. B* 36:111–47

Teh YW, Jordan MI. 2008. Hierarchical Bayesian nonparametric models with applications. In *Bayesian Nonparametrics*, ed. NL Hjort, C Holmes, P Müller, SG Walker, pp. 158–207. New York: Cambridge Univ. Press

Thomson G. 1939. The factorial analysis of human ability. *Br. J. Educ. Psychol.* 9:188–95

Thurstone LL. 1931. Multiple factor analysis. *Psychol. Rev.* 38:406–27

Thurstone LL. 1938. *Primary Mental Abilities*. Chicago: Univ. Chicago Press

Tierney L, Kass RE, Kadane JB. 1989. Fully exponential Laplace approximations to expectations and variances of nonpositive functions. *J. Am. Stat. Assoc.* 84:710–16

Tipping ME, Bishop CM. 1999. Probabilistic principal component analysis. *J. R. Stat. Soc. B* 61:611–22

Tukey JW. 1962. The future of data analysis. *Ann. Math. Stat.* 33:1–67

van Dyk DA, Kang H. 2004. Highly structured models for spectral analysis in high-energy astrophysics. *Stat. Sci.* 19:275–93

Wainwright MJ, Jordan MI. 2008. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.* 1:1–305

Wang C, Blei DM. 2013. Variational inference in nonconjugate models. *J. Mach. Learn. Res.* 14:1005–1031

West M, Harrison J. 1997. *Bayesian Forecasting and Dynamic Models*. Berlin: Springer

Wiegerinck W. 2000. Variational approximations between mean field theory and the junction tree algorithm. *Proc. Conf. Uncertain. Artif. Intell.* 16:626–33

Xing EP, Jordan MI, Russell S. 2003. A generalized mean field algorithm for variational inference in exponential families. *Proc. Conf. Uncertain. Artif. Intell.* 19:583–91