MCMC and Bayesian Modeling

These lecture notes[1] provide an introduction to Bayesian modeling and MCMC algorithms including the Metropolis-Hastings and Gibbs Sampling algorithms. We discuss some of the challenges associated with running MCMC algorithms including the important question of determining when convergence to stationarity has been achieved. To address this issue we introduce the popular convergence diagnostic approach of Gelman and Rubin. Many examples and applications of MCMC are provided.

In the appendix we also discuss various other topics including model checking and model selection for Bayesian models, Hamiltonian Monte-Carlo (an MCMC algorithm that was designed to handle multi-modal distributions and one that forms the basis for many current state-of-the-art MCMC algorithms), empirical Bayesian methods and how MCMC methods can also be used in non-Bayesian applications such as graphical models.

1 Bayesian Modeling

Not surprisingly, Bayes's Theorem is the key result that drives Bayesian modeling and statistics. Let $\mathcal{S}$ be a sample space and let $B_1, \ldots, B_K$ be a partition of $\mathcal{S}$ so that (i) $\bigcup_k B_k = \mathcal{S}$ and (ii) $B_i \bigcap B_j = \emptyset$ for all $i \neq j$.

Theorem 1 (Bayes's Theorem) *Let A be any event. Then for any $1 \leq k \leq K$ we have*

$$P(B_k \mid A) = \frac{P(A \mid B_k)P(B_k)}{P(A)} = \frac{P(A \mid B_k)P(B_k)}{\sum_{j=1}^{K} P(A \mid B_j)P(B_j)}.$$

Of course there is also a continuous version of Bayes's Theorem with sums replaced by integrals. Bayes's Theorem provides us with a simple rule for updating probabilities when new information appears. In Bayesian modeling and statistics this new information is the observed data and it allows us to update our prior beliefs about parameters of interest which are themselves assumed to be random variables.

The Prior and Posterior Distributions

Let $\boldsymbol{\theta}$ be some unknown parameter vector of interest. We assume $\boldsymbol{\theta}$ is random with some distribution, $\pi(\boldsymbol{\theta})$. This is our prior distribution which captures our prior uncertainty regarding $\boldsymbol{\theta}$. There is also a random vector, $\mathbf{X}$, with PDF (or PMF) $p(\mathbf{x} \mid \boldsymbol{\theta})$ – this is the **likelihood**. The joint distribution of $\boldsymbol{\theta}$ and $\mathbf{X}$ is then given by

$$p(\boldsymbol{\theta}, \mathbf{x}) = \pi(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta})$$

and we can integrate the joint distribution to get the marginal distribution of $\mathbf{X}$, namely

$$p(\mathbf{x}) = \int_{\boldsymbol{\theta}} \pi(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta})\, d\boldsymbol{\theta}.$$

We can compute the posterior distribution via Bayes's Theorem so that

$$\pi(\boldsymbol{\theta} \mid \mathbf{x}) = \frac{\pi(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta})}{p(\mathbf{x})} = \frac{\pi(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} \pi(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta})\, d\boldsymbol{\theta}} \tag{1}$$

[1] Many of the figures in these notes were taken from one of the following sources: David Barber's *Bayesian Reasoning and Machine Learning*, Christopher Bishop's *Pattern Recognition and Machine Learning*, Gelman et al.'s *Bayesian Data Analysis* and Ruppert & Matteson *Statistics and Data Analysis for Financial Engineering*.

The mode of the posterior is called the maximum a posterior $(\mathbf{MAP})$ estimator while the mean is of course $\mathsf{E}[\boldsymbol{\theta} \mid \mathbf{X} = \mathbf{x}] = \int \boldsymbol{\theta}\, \pi(\boldsymbol{\theta} \mid \mathbf{x})\, d\boldsymbol{\theta}$. The **posterior predictive** distribution is the distribution of a new as yet unseen data-point, $\mathbf{X}_{new}$:

$$\begin{aligned} p(\mathbf{x}_{new}) := p(\mathbf{x}_{new} \mid \mathbf{x}) &= \int_{\boldsymbol{\theta}} p(\mathbf{x}_{new}, \boldsymbol{\theta} \mid \mathbf{x})\, d\boldsymbol{\theta} \\ &= \int_{\boldsymbol{\theta}} p(\mathbf{x}_{new} \mid \boldsymbol{\theta}, \mathbf{x}) \pi(\boldsymbol{\theta} \mid \mathbf{x})\, d\boldsymbol{\theta} \\ &= \int_{\boldsymbol{\theta}} p(\mathbf{x}_{new} \mid \boldsymbol{\theta}) \pi(\boldsymbol{\theta} \mid \mathbf{x})\, d\boldsymbol{\theta} \end{aligned}$$

where the final equality follows because the data are assumed i.i.d. given $\boldsymbol{\theta}$. Much of Bayesian analysis is concerned with "understanding" the posterior $\pi(\boldsymbol{\theta} \mid \mathbf{x})$. Note that

$$\pi(\boldsymbol{\theta} \mid \mathbf{x}) \propto\ \pi(\boldsymbol{\theta}) p(\mathbf{x} \mid \boldsymbol{\theta})$$

which is what we often work with in practice. Sometimes we can recognize the form of the posterior by simply inspecting $\pi(\boldsymbol{\theta}) p(\mathbf{x} \mid \boldsymbol{\theta})$. But typically we cannot recognize the posterior and cannot compute the denominator in (1) either. In such cases approximate inference techniques such as MCMC are required. We begin with a simple example.

Example 1 (A Beta Prior and Binomial Likelihood)
Let $\theta \in (0, 1)$ represent some unknown probability. We assume a Beta(α, β) prior so that

$$\pi(\theta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha, \beta)}, \quad 0 < \theta < 1.$$

We also assume that $X \mid \theta \sim \mathsf{Bin}(n, \theta)$ so that $p(x \mid \theta) = \binom{n}{x} \theta^x (1-\theta)^{n-x}, \ \ x = 0, \ldots, n.$ The posterior then satisfies

$$\begin{aligned} p(\theta \mid x) &\propto \pi(\theta) p(x \mid \theta) \\ &= \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha, \beta)} \binom{n}{x} \theta^x (1-\theta)^{n-x} \\ &\propto \theta^{\alpha+x-1}(1-\theta)^{n-x+\beta-1} \end{aligned}$$

which we recognize as the Beta$(\alpha + x, \beta + n - x)$ distribution! See Figure 20.1 from *Statistics and Data Analysis for Financial Engineering* by Ruppert and Matteson for a numerical example and a visualization of how the data and prior interact to produce the posterior distribution. ∎

Exercise 1 *How can we interpret the prior distribution in Example 1?*

1.1 Conjugate Priors

Consider the following probabilistic model. The parameter vector $\boldsymbol{\theta}$ has prior $\pi(\,\cdot\,; \boldsymbol{\alpha}_0)$ while the data $\mathbf{X} = (X_1, \ldots, X_N)$ is distributed as $p(\mathbf{x} \mid \boldsymbol{\theta})$. As we saw earlier, the posterior distribution satisfies

$$p(\boldsymbol{\theta} \mid \mathbf{x}) \propto\ p(\boldsymbol{\theta}, \mathbf{x}) = p(\mathbf{x} \mid \boldsymbol{\theta}) \pi(\boldsymbol{\theta}; \boldsymbol{\alpha}_0).$$

We say the prior $\pi(\boldsymbol{\theta}; \alpha)$ is a conjugate prior for the likelihood $p(\mathbf{x} \mid \boldsymbol{\theta})$ if the posterior satisfies

$$p(\boldsymbol{\theta} \mid \mathbf{x}) = \pi(\boldsymbol{\theta}; \boldsymbol{\alpha}(\mathbf{x}))$$

so that the observations influence the posterior *only* via a parameter change $\boldsymbol{\alpha}_0 \to \boldsymbol{\alpha}(\mathbf{x})$. In particular, the form or type of the distribution is unchanged. In Example 1, for example, we saw the beta distribution is conjugate for the binomial likelihood. Here are two further examples.

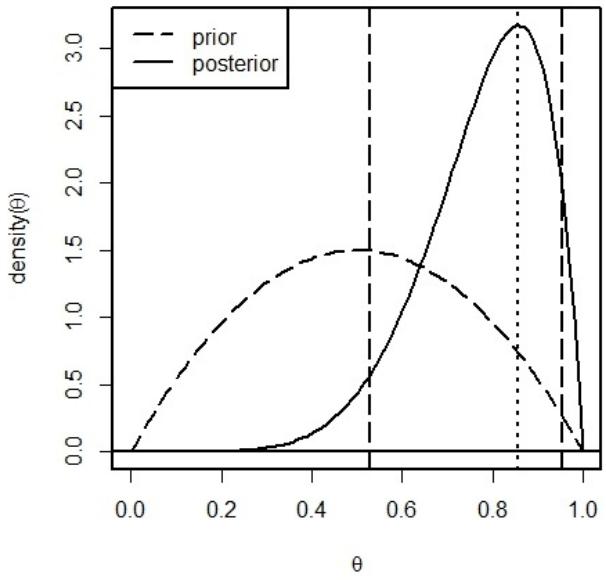

Figure 20.1 (Taken from from Ruppert's *Statistics and Data Analysis for FE*): Prior and posterior densities for $\alpha = \beta = 2$ and $n = x = 5$. The dashed vertical lines are at the lower and upper 0.05-quantiles of the posterior, so they mark off a 90% equal-tailed posterior interval. The dotted vertical line shows the location of the posterior mode at $\theta = 6/7 = 0.857$.

Example 2 (Conjugate Prior for Mean of a Normal Distribution)
Suppose $\theta \sim \mathsf{N}(\mu_0, \gamma_0^2)$ and $p(X_i \mid \theta) = \mathsf{N}(\theta, \sigma^2)$ for $i = 1, \ldots, N$ with σ^2 is assumed known. In this case we have $\boldsymbol{\alpha}_0 = (\mu_0, \gamma_0^2)$. If $\mathbf{X} = (X_1, \ldots, X_N)$ we then have

$$\begin{aligned} p(\theta \mid \mathbf{x}) &\propto p(\mathbf{x} \mid \boldsymbol{\theta})\pi(\theta; \boldsymbol{\alpha}_0) \\ &\propto e^{-\frac{(\theta-\mu_0)^2}{2\gamma_0^2}} \prod_{i=1}^{N} e^{-\frac{(x_i-\theta)^2}{2\sigma^2}} \\ &\propto \exp\left(-\frac{(\theta-\mu_1)^2}{2\gamma_1^2}\right) \end{aligned}$$

where

$$\gamma_1^{-2} := \gamma_0^{-2} + N\sigma^{-2} \quad \text{and} \quad \mu_1 := \gamma_1^2\big(\mu_0\gamma_0^{-2} + \sum_{i=1}^{n} x_i\sigma^{-2}\big).$$

Of course we recognize $p(\theta \mid \mathbf{x})$ as the $\mathsf{N}(\mu_1, \gamma_1^2)$ distribution. ∎

Example 3 (Conjugate Prior for Mean and Variance of a Normal Distribution)
Suppose that $p(X_i \mid \theta) = \mathsf{N}(\mu, \sigma^2)$ for $i = 1, \ldots, N$ and let $\mathbf{X} := (X_1, \ldots, X_N)$. We now assume μ and σ^2 are unknown so that $\boldsymbol{\theta} = (\mu, \sigma^2)$. We assume a *joint* prior of the form

$$\begin{aligned} \pi(\mu, \sigma^2) &= \pi(\mu \mid \sigma^2)\pi(\sigma^2) \\ &= \mathsf{N}\left(\mu_0, \sigma^2/\kappa_0\right) \times \text{Inv-}\chi^2\left(\nu_0, \sigma_0^2\right) \\ &\propto \sigma^{-1}\left(\sigma^2\right)^{-(\nu_0/2+1)} \exp\left(-\frac{1}{2\sigma^2}\left[\nu_0\sigma_0^2 + \kappa_0(\mu_0-\mu)^2\right]\right) \end{aligned}$$

which we recognize as the N-Inv-$\chi^2(\mu_0, \sigma_0^2/\kappa_0, \nu_0, \sigma_0^2)$ PDF. Note that μ and σ^2 are not independent under this joint prior.

Exercise 2 *Show that multiplying this prior by the normal likelihood yields a N-Inv-χ^2 distribution.*

∎

1.2 The Exponential Family of Distributions

The canonical form of the exponential family distribution is

$$p(\mathbf{x} \mid \boldsymbol{\theta}) = h(\mathbf{x}) e^{\boldsymbol{\theta}^\top \mathbf{u}(\mathbf{x}) - \psi(\boldsymbol{\theta})} \tag{2}$$

where $\boldsymbol{\theta} \in \mathbf{R}^m$ is a parameter vector and $\mathbf{u}(\mathbf{x}) = (u_1(\mathbf{x}), \ldots, u_m(\mathbf{x}))$ is the vector of *sufficient statistics*. The exponential family includes Normal, Gamma, Beta, Poisson, Dirichlet, Wishart and Multinomial distributions as special cases. The exponential family is also essentially the only distribution with a non-trivial conjugate prior. This conjugate prior takes the form

$$\pi(\boldsymbol{\theta}; \boldsymbol{\alpha}, \gamma) \propto e^{\boldsymbol{\theta}^\top \boldsymbol{\alpha} - \gamma \psi(\boldsymbol{\theta})}. \tag{3}$$

Combining (2) and (3) we see the posterior takes the form

$$\begin{aligned} p(\boldsymbol{\theta} \mid \mathbf{x}, \boldsymbol{\alpha}, \gamma) &\propto e^{\boldsymbol{\theta}^\top \mathbf{u}(\mathbf{x}) - \psi(\boldsymbol{\theta})} e^{\boldsymbol{\theta}^\top \boldsymbol{\alpha} - \gamma \psi(\boldsymbol{\theta})} = e^{\boldsymbol{\theta}^\top (\alpha + \mathbf{u}(\mathbf{x})) - (\gamma+1)\psi(\boldsymbol{\theta})} \\ &= \pi(\boldsymbol{\theta} \mid \boldsymbol{\alpha} + \mathbf{u}(\mathbf{x}), \gamma + 1) \end{aligned}$$

which (as claimed) has the same form as the prior.

1.3 Computational Issues in Bayesian Modeling

Selecting an appropriate prior is a key component of Bayesian modeling. With only a finite amount of data, the prior can have a very large influence on the posterior. It is important to be aware of this and to understand the sensitivity of posterior inference to the choice of prior. In practice it is common to use **non-informative priors** to limit this influence. When possible conjugate priors are often chosen for tractability reasons.

A common misconception is that the only advantage of the Bayesian approach over the *frequentist approach* is that the choice of prior allows us to express our prior beliefs on quantities of interest. In fact there are many other more important advantages including modeling flexibility via MCMC, exact inference rather than asymptotic inference, the ability to estimate functions of any parameters without "plugging" in MLE estimates, more accurate estimates of parameter uncertainty, etc. Of course there are disadvantages to the Bayesian approach as well. These include the subjectivity induced by choice of prior as well high computational costs. Despite differences between the Bayesian and frequentist approaches we do have the following important and satisfying result.

Theorem 2 (Bernstein-von Mises) *Under suitable assumptions and for sufficiently large sample sizes,the posterior distribution of $\boldsymbol{\theta}$ is approximately normal with mean equal to the true value of $\boldsymbol{\theta}$ and variance equal to the inverse of the Fisher information matrix.*

The Bernstein-von Mises Theorem implies that Bayesian and MLE estimators have the same large sample properties. This is not really surprising since the influence of the prior should diminish with increasing sample sizes. But this is a theoretical result and we often don't have "large" sample sizes so it's quite possible for the posterior to be (very) non-normal and even multi-modal. Moreover, the "suitable assumptions" mentioned in the theorem don't hold in many interesting models, including those for example where the number of parameters grows with the number of data-points.

Most of Bayesian inference is concerned with (simulating from) the posterior

$$\pi(\boldsymbol{\theta} \mid \mathbf{x}) \;\propto\; \pi(\boldsymbol{\theta}) p(\mathbf{x} \mid \boldsymbol{\theta}) \tag{4}$$

without knowing the constant of proportionality in (4). This leads to the general *sampling problem:*

The Sampling Problem

Suppose we are given a distribution function

$$p(\mathbf{z}) = \frac{1}{Z_p}\tilde{p}(\mathbf{z}) \tag{5}$$

where $\tilde{p}(\mathbf{z}) \geq 0$ is easy to compute but Z_p is (too) hard to compute. This very important situation arises in several contexts:

1. In Bayesian models where $\tilde{p}(\boldsymbol{\theta}) := p(\mathbf{x} \mid \boldsymbol{\theta})\pi(\boldsymbol{\theta})$ is easy to compute but $Z_p := p(\mathbf{x}) = \int_{\boldsymbol{\theta}} \pi(\boldsymbol{\theta})p(\mathbf{x} \mid \boldsymbol{\theta})d\boldsymbol{\theta}$ can be very difficult or impossible to compute.
2. In models from statistical physics, e.g. the Ising model, we only know $\tilde{p}(\mathbf{z}) = e^{-\mathcal{E}(\mathbf{z})}$, where $\mathcal{E}(\mathbf{z})$ is an "energy" function. (The Ising model is an example of a *Markov network* or an undirected graphical model.)
3. Dealing with evidence in directed graphical models such as belief networks aka directed acyclic graphs.

The sampling problem is the problem of simulating from $p(\mathbf{z})$ in (5) without knowing the constant Z_p. While the **acceptance-rejection** algorithm can be used, it is very inefficient in high dimensions and an alternative approach is required. That alternative approach is Markov Chain Monte-Carlo (MCMC).

---

2 Markov Chain Monte-Carlo (MCMC)

MCMC algorithms were originally developed in the 1940's by physicists at Los Alamos. These physicists included Ulam (inspired by playing solitaire!), Von Neumann (who developed the acceptance-rejection algorithm) and others. They were interested in modeling the probabilistic behavior of collections of atomic particles. They could not do this analytically but they wondered if they could use simulation. Simulation was difficult as the normalization constant Z_p was not known. Moreover, simulation (as a computational tool) hadn't (why?) been "discovered" yet although simulation ideas had been around for some time – e.g. Buffon's needle (1700's), Lord Kelvin (1901) and Fermi (1930's). (In fact the term "Monte-Carlo" was coined at Los Alamos.)

Ulam and Metropolis overcame this problem by constructing a Markov chain for which the desired distribution was the stationary distribution of the Markov chain. They then only needed to simulate the Markov chain until stationarity was achieved. Towards this end, they introduced the **Metropolis** algorithm and its impact was enormous. Afterwards MCMC was introduced to statistics and generalized with the **Metropolis-Hastings** algorithm (1970) and the **Gibbs sampler** of Geman and Geman (1984).

2.1 Markov Chains

Before describing the basic MCMC algorithm we must first recall some ideas from the theory of Markov chains. We have the following definitions.

Definition 1 *A sequence of random variables $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_t\}$ on a discrete state space Ω is called a (first-order) Markov Chain if*

$$p(\mathbf{X}_t = \mathbf{x}_t \mid \mathbf{X}_{t-1} = \mathbf{x}_{t-1}, \ldots, \mathbf{X}_1 = \mathbf{x}_1) = p(\mathbf{X}_t = \mathbf{x}_t \mid \mathbf{X}_{t-1} = \mathbf{x}_{t-1}).$$

We will restrict ourselves to time-homogeneous Markov chains so that

$$p(\mathbf{X}_t = \mathbf{x}_t \mid \mathbf{X}_{t-1} = \mathbf{x}_{t-1}) = \mathbf{P}(\mathbf{x}_t \mid \mathbf{x}_{t-1}) \in \mathbf{R}^{|\Omega|\times|\Omega|}$$

Note it's easy to check that $[p(\mathbf{X}_{t+1} = \mathbf{x}_t \mid \mathbf{X}_{t-1} = \mathbf{x}_{t-1})]_{(\mathbf{x}_t, \mathbf{x}_{t-1})\in\Omega} = \mathbf{P}^2$.

Definition 2 *A Markov chain is called* **ergodic** *if there exists $r \in \mathbb{N}^+$ such that $\mathbf{P}^r > 0$*

We note that the ergodicity of a Markov chain is equivalent to the Markov chain being:

1. **Irreducible**: For all $\mathbf{x}, \mathbf{y} \in \Omega$, there exists $r(\mathbf{x}, \mathbf{y}) \in \mathbb{N}^+$ s.t. $\mathbf{P}^{r(\mathbf{x},\mathbf{y})}(\mathbf{x}, \mathbf{y}) > 0$
2. **Aperiodic**: For all $\mathbf{x} \in \Omega$, $\text{GCD}\{r \in \mathbb{N}^+ : P^r(\mathbf{x}, \mathbf{x}) > 0\} = 1$.

Definition 3 *A* **stationary distribution** *of a Markov chain is a distribution π on Ω such that*

$$\pi(\mathbf{y}) = \sum_{\mathbf{x} \in \Omega} P(\mathbf{y} \mid \mathbf{x})\pi(\mathbf{x}). \tag{6}$$

We have the following important result.

Theorem 3 *A finite ergodic Markov Chain has a unique stationary distribution.*

Definition 4 *The total variation distance, $d_{TV}(\mu, \nu)$, between two probability measures μ, ν on Ω is defined as*

$$\|\mu - \nu\|_{TV} \quad := \quad \max_{S \subset \Omega} \{\mu(S) - \nu(S)\} \quad = \quad \frac{1}{2} \sum_{\mathbf{z} \in \Omega} |\mu(\mathbf{z}) - \nu(\mathbf{z})| \,.$$

The mixing time function, $\tau_{\text{mix}}(\epsilon)$, is defined as the time until the total variation distance to π is below ϵ. It can be shown to satisfy[2]

$$\begin{aligned} \tau_{\text{mix}}(\epsilon) \quad &:= \quad \max_{\mathbf{x}_0 \in \Omega} \min \left\{ t \in \mathbb{N}^+ : \|P^t(\cdot, \mathbf{x}_0) - \pi(\cdot)\|_{TV} \leq \epsilon \right\} \\ &\sim \quad \ln\left(\frac{1}{\epsilon}\right). \end{aligned}$$

Definition 5 *A Markov chain is said to be* **reversible** *if there exists a probability distribution π on Ω such that*

$$P(\mathbf{x} \mid \mathbf{y})\pi(\mathbf{y}) = P(\mathbf{y} \mid \mathbf{x})\pi(\mathbf{x}) \tag{7}$$

It's easy to check that if π satisfies (7) then it is the stationary distribution of the Markov chain since then we have

$$\sum_{\mathbf{x}} P(\mathbf{y} \mid \mathbf{x})\pi(\mathbf{x}) = \sum_{\mathbf{x}} P(\mathbf{x} \mid \mathbf{y})\pi(\mathbf{y}) = \pi(\mathbf{y})$$

which is (6). Note that (7) implies the chain moves from $\mathbf{x}$ to $\mathbf{y}$ at the same rate as it moves from $\mathbf{y}$ to $\mathbf{x}$ (when in equilibrium). For this reason (7) is often called the **detailed balance equation**. Satisfying the detailed balance equation is a *sufficient* (but not necessary) condition for π to be *a* stationary distribution We will also want to have ergodicity to guarantee that π is *the* stationary distribution.

Exercise 3 *What is the stationary distribution for a reversible symmetric Markov chain?*

There are analogous definitions and results for Markov chains on *continuous* state spaces that we will not state here.

2.2 The Metropolis-Hastings Algorithm

Returning to our sampling problem, suppose we want to sample from a distribution $p(\mathbf{x}) := \tilde{p}(\mathbf{x})/Z_p$. To do this we first construct a (reversible) Markov chain as follows. Let $\mathbf{X}_t = \mathbf{x}$ be the current state. We then perform the following two steps repeatedly:

1. Generate $\mathbf{Y} \sim Q(\cdot \mid \mathbf{x})$ for some Markov transition matrix Q.

   Let $\mathbf{y}$ be the generated value.

[2] We would like to have similar properties for continuous sample spaces!

2. Set $\mathbf{X}_{t+1} = \mathbf{y}$ with probability $\alpha(\mathbf{y} \mid \mathbf{x}) := \min\left\{\frac{\tilde{p}(\mathbf{y})}{\tilde{p}(\mathbf{x})} \cdot \frac{Q(\mathbf{x}|\mathbf{y})}{Q(\mathbf{y}|\mathbf{x})}, 1\right\}$.

   Otherwise set $\mathbf{X}_{t+1} = \mathbf{x}$.

Claim: The resulting Markov chain is reversible with stationary distribution $p(\mathbf{x}) = \tilde{p}(\mathbf{x})/Z_p$. We can therefore sample from $p(\mathbf{x})$ by running the algorithm until stationarity is achieved and then using generated points as our samples. Note that Z_p is not required for the algorithm! Note also that if $\mathbf{Y} = \mathbf{y}$ is rejected then the current state $\mathbf{x}$ becomes the next state so that $\mathbf{X}_t = \mathbf{X}_{t+1} = \mathbf{x}$. (We are assuming that ergodicity is also satisfied — this is generally straightforward to check in a given application.)

Proof of Claim: We simply check that $p(\mathbf{x})$ satisfies the detailed balance equations. We have

$$\begin{aligned}
\underbrace{\alpha(\mathbf{y} \mid \mathbf{x})Q(\mathbf{y} \mid \mathbf{x})}_{P(\mathbf{y}|\mathbf{x})} p(\mathbf{x}) &= \min\left\{\frac{p(\mathbf{y})}{p(\mathbf{x})} \cdot \frac{Q(\mathbf{x} \mid \mathbf{y})}{Q(\mathbf{y} \mid \mathbf{x})}, 1\right\} Q(\mathbf{y} \mid \mathbf{x})p(\mathbf{x}) \\
&= \min\left\{Q(\mathbf{x} \mid \mathbf{y})p(\mathbf{y}), Q(\mathbf{y} \mid \mathbf{x})p(\mathbf{x})\right\} \\
&= \min\left\{1, \frac{p(\mathbf{x})}{p(\mathbf{y})} \cdot \frac{Q(\mathbf{y} \mid \mathbf{x})}{Q(\mathbf{x} \mid \mathbf{y})}\right\} Q(\mathbf{x} \mid \mathbf{y})p(\mathbf{y}) \\
&= \underbrace{\alpha(\mathbf{x} \mid \mathbf{y})Q(\mathbf{x} \mid \mathbf{y})}_{P(\mathbf{x}|\mathbf{y})} p(\mathbf{y})
\end{aligned}$$

as desired. There are still some important questions that need to be addressed:

1. How do we determine when stationarity is achieved?

   - In general it is difficult to provide a theoretical answer to this question. Instead, we check for convergence to stationarity on a case-by-case basis using **convergence diagnostics**. We will discuss this further in Section 4.2.

2. There are many possible choices of **proposal distribution**, $Q(\cdot \mid \cdot)$. Which one should we use?

   - This is an important question since $Q(\cdot \mid \cdot)$ influences how much time is required to reach stationarity. There appears to be relatively few results on this question although rules of thumb and experience / experimentation do provide (partial) answers. See also the related discussion in Example 5 below.

Exercise 4 *Are the samples produced by the MCMC algorithm independent?*

Example 4 (Simulating from a Bivariate Gaussian Distribution)
Figure 11.9 from Bishop's *Pattern Recognition and Machine Learning* displays samples from a Gaussian distribution that were generated using the Metropolis algorithm with an isotropic Gaussian distribution as the proposal distribution, $Q(\cdot \mid \cdot)$. Specifically $Q(\cdot \mid \mathbf{x}_t) \sim \mathsf{N}_2(\mathbf{x}_t, 0.2 \times \mathbf{I}_2)$ where $\mathbf{I}_n$ denotes the n-dimensional identity matrix. ∎

Exercise 5 *Can you explain the pattern of accepted and rejected samples in Figure 11.9? This is a general phenomenon and is important to understand. See also Example 5 below.*

Example 5 (Simulating from a Multi-Modal Distribution)
Figure 27.8 from Barber's *Bayesian Reasoning and Machine Learning* displays samples from a bi-modal density that were generated using a bivariate normal proposal. In general, simulating from multi-modal distributions using MCMC can be challenging, particularly in high-dimensional problems.

Exercise 6 *Consider carefully the following questions all of which refer to Figure 27.8. (Understanding them is key to understanding the issues that arise when simulating from multi-modal distributions.)*

1. *Suppose instead of using a* $\mathsf{N}(\mathbf{x}' \mid \mathbf{x}, \mathbf{I})$ *proposal we instead used a* $\mathsf{N}(\mathbf{x}' \mid \mathbf{x}, \sigma\mathbf{I})$ *with* $\sigma << 1$ *a constant that is very small. How do you think the algorithm would perform then? Specifically, do you think convergence to stationarity would happen "quickly"?*

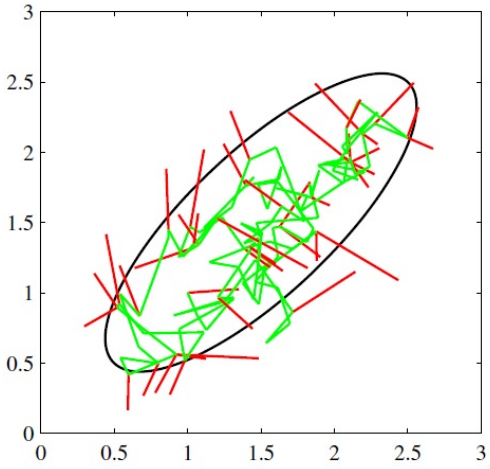

Figure 11.9 (Taken from Bishop's *Pattern Recognition and Machine Learning*): A simple illustration using the Metropolis algorithm to sample from a Gaussian distribution whose one standard-deviation contour is shown by the ellipse. The proposal distribution is an isotropic Gaussian distribution whose standard deviation is 0.2. Steps that are accepted are shown as green lines, and rejected steps are shown in red. A total of 150 candidate samples are generated, of which 43 are rejected.

2. *What might be a solution to the problem outlined in Q1?*
3. *If you increased σ what effect will this have on the Metropolis-Hastings algorithm?*

---

3 Gibbs Sampling

Gibbs sampling[3] is an MCMC sampler introduced by Geman and Geman in 1984. Let $\mathbf{x}^{(t)} \in \mathbf{R}^m$ denote the current sample. Then Gibbs sampling proceeds as follows:

1. Pick an index $k \in \{1, \ldots, m\}$ either via round-robin or uniformly at random
2. Set $\mathbf{x}_j^{(t+1)} = \mathbf{x}_j^{(t)}$, for $j \neq k$, i.e. $\mathbf{x}_{-k}^{(t+1)} = \mathbf{x}_{-k}^{(t)}$
3. Generate $\mathbf{x}_k^{(t+1)} \sim p(\mathbf{x}_k \mid \mathbf{x}_{-k}^{(t)})$

In Gibbs only one component of $\mathbf{x}$ is updated at a time. It is common to simply order the m components and update them sequentially. We can then let $\mathbf{x}_k^{(t+1)}$ be the value of the chain after all m updates rather than each individual update. Gibbs sampling is a very popular method for applications where the conditional distributions, $p(\mathbf{x}_j \mid \mathbf{x}_{-k}^{(t)})$, are easy to simulate from. This is the case for for *conditionally conjugate* models and others.

It is easy to see that Gibbs sampling is a special case of Metropolis-Hastings sampling with

$$Q_k(\mathbf{y} \mid \mathbf{x}) = \begin{cases} p(\mathbf{y}_k \mid \mathbf{x}_{-k}) & \mathbf{y}_{-k} = \mathbf{x}_{-k} \\ 0 & \text{otherwise.} \end{cases}$$

and that each component update will be accepted with probability 1. One must be careful, however, that the component-wise Markov Chain is ergodic as discussed earlier. See Barber's Figure 27.5 in Section 3.1 for an example where the chain is not ergodic in which the Gibbs sampler would fail to converge to the desired stationary distribution.

[3]The algorithm is named after the physicist J. W. Gibbs who died approx. 80 years earlier in 1903.

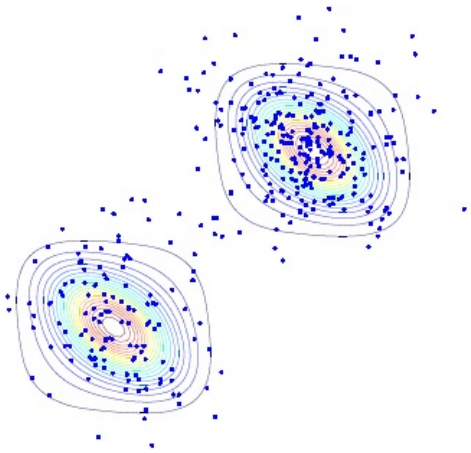

Figure 27.8 (Taken from Barber's *Bayesian Reasoning and Machine Learning*): Metropolis-Hastings samples from a bi-variate distribution $p(x_1, x_2)$ using a proposal $\tilde{q}(\mathbf{x}' \mid \mathbf{x}) = \mathrm{N}(\mathbf{x}' \mid \mathbf{x}, \mathbf{I})$. We also plot the iso-probability contours of p. Although $p(\mathbf{x})$ is multi-modal, the dimensionality is low enough and the modes sufficiently close such that a simple Gaussian proposal distribution is able to bridge the two modes. In higher dimensions, such multi-modality is more problematic.

Example 6 (A Simple Example)
Consider the distribution

$$p(x, y) = \frac{n!}{(n-x)!x!} y^{(x+\alpha-1)}(1-y)^{(n-x+\beta-1)}, \quad x \in \{0, \ldots, n\}, y \in [0, 1]. \tag{8}$$

It is hard to simulate directly from $p(x, y)$ but the conditional distributions are easy to work with. We see that

- $p(x \mid y) \sim \mathsf{Bin}(n, y)$
- $p(y \mid x) \sim \mathsf{Beta}(x + \alpha, n - x + \beta)$

and since it's easy to simulate from each conditional, it's easy to run a Gibbs sampler to simulate from the joint distribution. ∎

Exercise 7 *Can you identify a situation where the distribution of (8) might arise?* Hint: *Refer to one of our earlier examples. (Note that the marginal distribution of x has a* beta-binomial *distribution.)*

Example 7 (Hierarchical Models)

Diet	Measurements
A	62, 60, 63, 59
B	63, 67, 71, 64, 65, 66
C	68, 66, 71, 67, 68, 68
D	56, 62, 60, 61, 63, 64, 63, 59

Table 11-2 (Taken from *Bayesian Data Analysis*, 2^{nd} edition by Gelman et al.): Coagulation time in seconds for blood drawn from 24 animals randomly allocated to four different diets. Different treatments have different numbers of observations because the randomization was unrestricted. From Box, Hunter, and Hunter (1978), who adjusted the data so that the averages are integers, a complication we ignore in our analysis.

Gibbs sampling is particulary suited for *hierarchical* or *multi-level* models, an important class of models throughout statistics and machine learning. We consider here an example from *Bayesian Data Analysis* by Gelman et al. and the data is presented in Table 11-2 above. The data-points y_{ij}, for $i = 1, \ldots, n_j$ and $j = 1, \ldots, J$, are assumed to be independently normally distributed within each of J groups with means θ_j and common variance σ^2. That is,

$$y_{ij} \mid \theta_j \sim N(\theta_j, \sigma^2).$$

The total number of observations is $n = \sum_{j=1}^{J} n_j$. Group means are assumed to follow a normal distribution with unknown mean μ and variance τ^2 so that

$$\theta_j \sim N(\mu, \tau^2).$$

A uniform prior is assumed[4] for $(\mu, \log\sigma, \tau)$ which is equivalent to assuming (why?) that $p(\mu, \log\sigma, \log\tau) \propto \tau$ The posterior is then given by

$$p(\boldsymbol{\theta}, \mu, \log\sigma, \log\tau \mid \mathbf{y}) \propto \tau \prod_{j=1}^{J} \mathsf{N}\left(\theta_j \mid \mu, \tau^2\right) \prod_{j=1}^{J} \prod_{i=1}^{n_j} \mathsf{N}\left(y_{ij} \mid \theta_j, \sigma^2\right). \tag{9}$$

We will see from (9) that all conditional distributions required for Gibbs sampler have simple conjugate forms:

1. **Conditional Posterior Distribution of Each θ_j**

   We simply need to gather the terms (from the posterior in (9)) that only involve θ_j and then simplify to obtain

   $$\theta_j \mid (\boldsymbol{\theta}_{-j}, \mu, \sigma, \tau, \mathbf{y}) \sim \mathsf{N}\left(\widehat{\theta}_j, V_{\theta_j}\right) \tag{10}$$

   where

   $$\widehat{\theta}_j := \frac{\frac{1}{\tau^2}\mu + \frac{n_j}{\sigma^2}\overline{y}_{.j}}{\frac{1}{\tau^2} + \frac{n_j}{\sigma^2}} \quad \text{and} \qquad V_{\theta_j} := \frac{1}{\frac{1}{\tau^2} + \frac{n_j}{\sigma^2}}.$$

   These conditional distributions are independent so generating the θ_j's one at a time is equivalent to drawing $\boldsymbol{\theta}$ all at once.

2. **Conditional Posterior Distribution of μ**

   Again, we simply gather terms from the posterior that only involve μ and then simplify to obtain

   $$\mu \mid (\boldsymbol{\theta}, \sigma, \tau, \mathbf{y}) \sim \mathsf{N}\left(\widehat{\mu}, \frac{\tau^2}{J}\right) \tag{11}$$

   where $\widehat{\mu} := \frac{1}{J}\sum_{j=1}^{J} \theta_j$.

3. **Conditional Posterior Distribution of σ^2**

   Gathering terms from the posterior that only involve σ and then simplifying, we obtain

   $$\sigma^2 \mid (\boldsymbol{\theta}, \mu, \tau, \mathbf{y}) \sim \text{Inv-}\chi^2\left(n, \widehat{\sigma}^2\right) \tag{12}$$

   where $\widehat{\sigma}^2 := \frac{1}{n}\sum_{j=1}^{J}\sum_{i=1}^{n_j}\left(y_{ij} - \theta_j\right)^2$.

4. **Conditional Posterior Distribution of τ^2**

   Again, we gather terms from the posterior that only involve τ and then simplify to obtain

   $$\tau^2 \mid (\boldsymbol{\theta}, \mu, \sigma, \mathbf{y}) \sim \text{Inv-}\chi^2\left(J - 1, \widehat{\tau}^2\right) \tag{13}$$

   where $\widehat{\tau}^2 := \frac{1}{J-1}\sum_{j=1}^{J}\left(\theta_j - \mu\right)^2$.

To start the Gibbs sampler we only (why?) need starting points for $\boldsymbol{\theta}$ and μ and then we use (10) to (13) to repeatedly generate samples from the conditional distributions. ∎

[4] If a uniform prior was assigned to $\log\tau$ then the posterior would be *improper* as discussed in Gelman et al. This emphasizes the importance of understanding the issues associated with choosing priors. We have not discussed these issues in these lecture notes but they are important.

Other Variations of Gibbs Sampling

Instead of updating just one component at a time we can also split $\mathbf{x}$ into blocks and update one block at a time. This is known as **blocked Gibbs sampling**. And if it's not possible to simulate directly from one or more of the conditional distributions then we can use rejection-sampling or Metropolis-Hastings sampling for those updates. This latter approach is sometimes called **Metropolis-within-Gibbs**.

Finally, suppose for example that we have a posterior defined over three variables $p(\alpha, \beta, \gamma) \propto g(\alpha, \beta, \gamma)$. If γ is not of interest it may be possible to *marginalize*, i.e. integrate, it out to compute $p(\alpha, \beta) \propto h(\alpha, \beta)$ in which case we just need to run the Gibbs sampler on α and β. Such an approach is often called a **collapsed Gibbs sampler**. A common application of collapsed Gibbs sampling is in the LDA model of Example 12 where the β and θ vectors can be marginalized so that the resulting Gibbs sampler just samples the z_i^d's.

Exercise 8 *Does the collapsed Gibbs sampler remind you of any variance reduction technique? If so, which one and why?*

Remark 1 *Sometimes it can be very convenient to* **de-marginalize** *by introducing additional random variables into the problem. See, for example, Example 9.*

3.1 Difficulties With Gibbs Sampling

Gibbs sampling is a very popular MCMC technique that is widely used. It does have some potential drawbacks, however. First, we need to be able to show that the Gibbs sampler Markov chain is ergodic. This will obviously be the case in many circumstances but it may sometimes be an issue. For example, Figure 27.5 from Barber's *BRML* displays a 2-dimensional example where the chain is not irreducible.

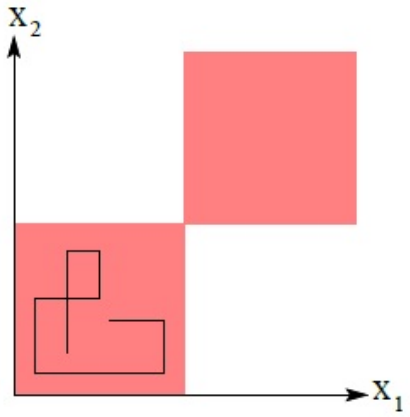

Figure 27.5 (Taken from Barber's *Bayesian Reasoning and Machine Learning*): A two dimensional distribution for which Gibbs sampling fails. The distribution has mass only in the shaded quadrants. Gibbs sampling proceeds from the l^{th} sample state (x_1^l, x_2^l) and then sampling from $p(x_2 \mid x_1^l)$, which we write (x_1^{l+1}, x_2^{l+1}) where $x_1^{l+1} = x_1^l$. One then continues with a sample from $p(x_1 \mid x_2 = x_2^{l+1})$, etc. If we start in the lower left quadrant and proceed this way, the upper right region is never explored.

A second problem that often arises with Gibbs sampling is that the samples might be strongly correlated (negatively or positively). In that event it may take too long to reach the stationary distribution. This phenomenon is discussed in the captions for Figure 27.7 from Barber's *BRML* and Figure 11.11 from Bishop's *PRML*, both of which are displayed below.

When the variables are very correlated a common strategy (to overcome this weakness) is to perform a simple transformation of variables so that the transformed variables are approximately independent.

Exercise 9 *Suppose the random variables $x_1, \ldots, x_d$ are independent. How long do you think it will take the Gibbs sampler to reach stationarity in that case?*

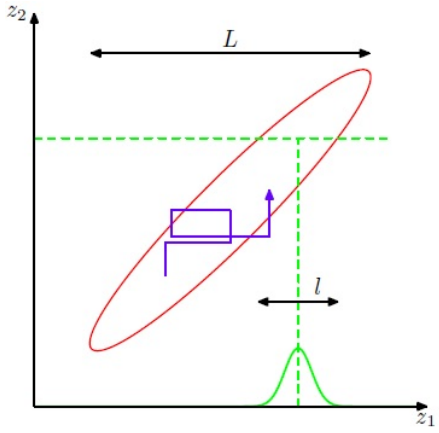

Figure 11.11 (Taken from Bishop's *Pattern Recognition and Machine Learning*): Illustration of Gibbs sampling by alternate updates of two variables whose distribution is a correlated Gaussian. The step size is governed by the standard deviation of the conditional distribution (green curve), and is $O(l)$, leading to slow progress in the direction of elongation of the joint distribution (red ellipse). The number of steps needed to obtain an independent sample from the distribution is $O((L/l)^2)$.

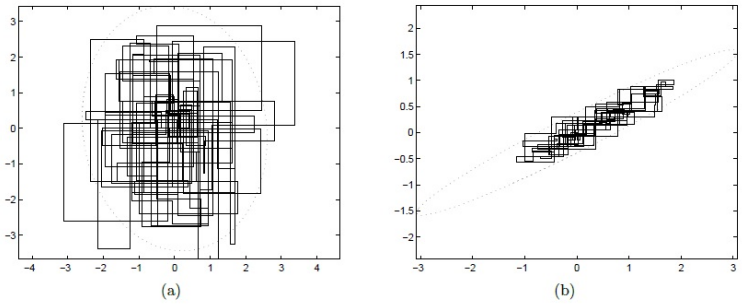

Figure 27.7 (Taken from Barber's *Bayesian Reasoning and Machine Learning*): Two hundred Gibbs samples for a two dimensional Gaussian. At each stage only a single component is updated. (a): For a Gaussian with low correlation, Gibbs sampling can move through the likely regions effectively. (b): For a strongly correlated Gaussian, Gibbs sampling is less effective and does not rapidly explore the likely regions.

The following example provides a cautionary example highlighting the dangers of blindly running a Gibbs sampler for a given set of conditional distributions.

Example 8 (From Casella and George, 1992)
The fact that Gibbs sampling works tells us that the conditional distributions are sufficient to define the joint distribution. But there is a subtle issue here as it is *not* the case that a set of proper well-defined conditional distributions will determine a proper joint distribution. Consider for example the following 2-dimensional example with

$$f(x \mid y) = ye^{-yx}, \quad 0 < x < \infty \tag{14}$$
$$f(y \mid x) = xe^{-xy}, \quad 0 < y < \infty \tag{15}$$

so that both conditionals are exponential distributions (and therefore well-defined). If we apply a Gibbs sampler here, however, we will not obtain a sample from any marginal or joint distribution. This is because (14) and (15) do not correspond to any joint distribution on (x, y). ∎

4 MCMC Convergence and Output Analysis

After running an MCMC we need to analyze the output in order to understand what the data is telling us about the quantities of interest.

4.1 MCMC Output Analysis

We are usually interested in scalar-valued functions of the parameter vector $\boldsymbol{\theta}$. Let $\psi(\boldsymbol{\theta})$ be one such function. If we have n MCMC samples from the stationary distribution then we have n samples of $\psi(\boldsymbol{\theta})$:

$$\{\psi_1 := \psi(\boldsymbol{\theta}_1),\ \ldots\ ,\ \psi_n := \psi(\boldsymbol{\theta}_n)\}\,.$$

The *sample mean* is then given by $\bar{\psi} = n^{-1}\sum_{i=1}^{n}\psi_1$. *Posterior intervals* for $\psi(\boldsymbol{\theta})$ can also be calculated:

1. Let $L(\alpha_1) := \alpha_1$ lower sample quantile and $U(\alpha_2) := \alpha_2$ upper sample quantile of $\psi_1, \ldots, \psi_n$. Then $(L(\alpha_1),\ U(\alpha_2))$ is a $1 - (\alpha_1 + \alpha_2)$ posterior interval.
2. If $\alpha_1 = \alpha_2 = \alpha/2$ then we obtain an equi-tailed $1 - \alpha$ posterior interval.
3. For a highest posterior density interval we solve (numerically) for α_1 and α_2 such that $\alpha = \alpha_1 + \alpha_2$ and $U(\alpha_2) - L(\alpha_1)$ is minimized. Note that this interval could be a *union of intervals* if the posterior of $\psi(\boldsymbol{\theta})$ is not unimodal. (*Kernel density estimates* of the posterior density can be plotted to help determine the number of modes.)

4.2 MCMC Convergence Diagnostics

Before performing the output analysis we must: (1) ensure the Markov chains have reached stationarity and (2) only use those samples that have been generated after stationarity has been reached. But it's impossible to ensure when these two conditions are satisfied since the Markov chain does not begin with the stationary distribution. Instead we can use various methods to assess whether or not stationarity *appears* to have been reached. These methods include:

1. *Visual inspection* where we plot variables (of interest) vs iteration number, plot running means of variables (of interest) etc. This process can be very informative but it also requires "manual" work.
2. *Statistical summaries* of MCMC output which are designed to diagnose convergence / non-convergence. We will consider the popular *Gelman-Rubin* approach here but we will not justify everything.

The Gelman-Rubin Approach to Diagnosing Convergence

Let m, n and $n_0 \in \mathbb{N}^+$ with m even. We run $m/2$ Markov chains for a total of $n_0 + 2n$ iterations each. The chains are begun from *over-dispersed* starting points. These starting points are usually obtained by generating them from some over-dispersed distribution. We discard the first n_0 samples from each chain – these samples constitute the *burn-in* period where the chains are assumed to be in their *transient phase*. It is common to take $n_0 = 2n$ so the first half of each chain is discarded. The remaining component of each chain is then split into two (sub-)chains, each containing n samples. This chain splitting is performed to help us determine if each chain has reached stationarity. At this point we therefore have m chains each containing n samples. Figure 11.2 from Gelman et al. below displays sample points from this process in an example with $m = 10$ corresponding to 5 chains.

We hope these $m \times n$ samples are from the desired stationarity distribution so we check that this appears to be the case by comparing the **between**-chain variance with the **within**-chain variance for all scalar quantities, ψ, of interest. Because the method is based on means and variances it is generally a good idea to *transform* the

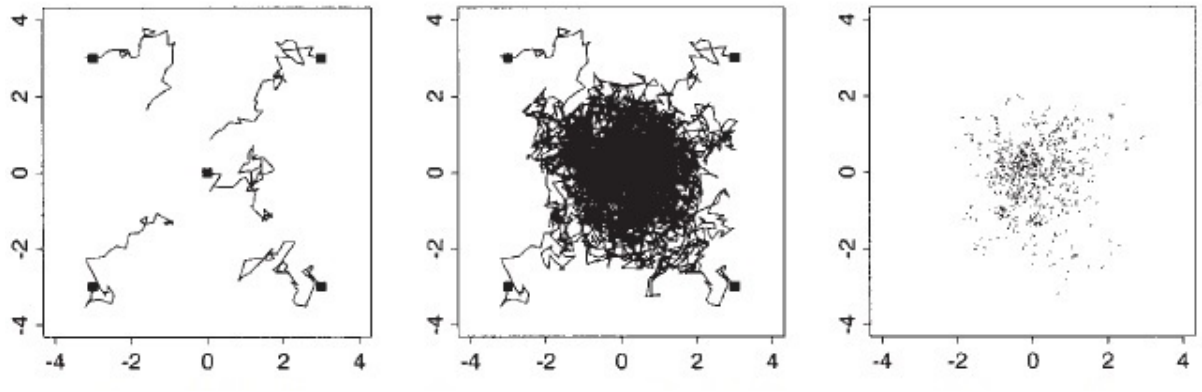

Figure 11.2 (Taken from Gelman et al.'s *BDA*, 2nd ed.): Five independent sequences of a Markov chain simulation for the bivariate unit normal distribution, with over-dispersed starting points indicated by solid squares. (a) After 50 iterations, the sequences are still far from convergence. (b) After 1000 iterations, the sequences are nearer to convergence. Figure (c) shows the iterates from the second halves of the sequences. The points in Figure (c) have been jittered so that steps in which the random walk stood still are not hidden.

scalar estimands so they are approximately normal. We can achieve this by, for example, taking logs of strictly positive quantities and taking logits of quantities that must lie in $(0, 1)$.

Let ψ_{ij} for $i = 1, \ldots n$ and $j = 1, \ldots, m$ be the MCMC samples computed after the burn-in period and after splitting the non-burn-in component of each chain in two. The **between**- and **within**-sequence variances, B and W, are computed as[5]

$$\begin{aligned} B &:= \frac{n}{m-1} \sum_{j=1}^{m} \left(\bar{\psi}_{\cdot j} - \bar{\psi}_{\cdot\cdot}\right)^2 \\ W &:= \frac{1}{m} \sum_{j=1}^{m} s_j^2 \quad \text{where} \quad s_j^2 := \frac{1}{n-1} \sum_{i=1}^{n} \left(\psi_{ij} - \bar{\psi}_{\cdot j}\right)^2 \end{aligned}$$

and where $\bar{\psi}_{\cdot j} := \frac{1}{n} \sum_{i=1}^{n} \psi_{ij}$ and $\bar{\psi}_{\cdot\cdot} := \frac{1}{m} \sum_{j=1}^{m} \bar{\psi}_{\cdot j}$. We can estimate $\mathrm{Var}\,(\psi \mid \mathbf{X})$ as a weighted average of W and B with

$$\widehat{\mathrm{Var}}^{+}\,(\psi \mid \mathbf{X}) = \frac{n-1}{n} W \; + \; \frac{1}{n} B. \tag{16}$$

Note that $\widehat{\mathrm{Var}}^{+}\,(\psi \mid \mathbf{X})$ *overestimates* the marginal posterior variance (of ψ) since the starting distribution is over-dispersed. But it will be unbiased when sampling from the desired stationary distribution.

We also note that for any finite n, it should be the case that W is an *underestimate* of $\mathrm{Var}\,(\psi \mid \mathbf{X})$. This follows since each individual sequence may not yet have had time to explore all of the target, i.e. stationary, distribution. But W should approach $\mathrm{Var}\,(\psi \mid \mathbf{X})$ in the limit as $n \to \infty$. We therefore monitor convergence through

$$\widehat{R} := \sqrt{\frac{\widehat{\mathrm{Var}}^{+}\,(\psi \mid \mathbf{X})}{W}}$$

Note that by the above argument, we should have $\widehat{R} > 1$ for any finite n but we also have $\widehat{R} \to 1$ as $n \to \infty$. This leads to the following rule of thumb for diagnosing convergence:

Rule of Thumb: Values of $\widehat{R} < 1.1$ are acceptable but the closer $\widehat{R}$ is to 1 the better. We then monitor $\widehat{R}$ for all quantities ψ of interest.

[5] B contains a factor of n because it is based on the variance of the within-sequence means, $\bar{\psi}_{\cdot j}$, each of which is an average of n values.

The Effective Sample Size

Note that B/n is the sample variance of the m chain means so that B/mn therefore estimates the Monte-Carlo variance of $\bar{\psi}_{\cdot\cdot}$. Suppose now that we could take an *independent* sample of size n_{eff}. The variance of the mean of this sample would be estimated as $\widehat{\mathrm{Var}}^{+}(\psi \mid \mathbf{X})/n_{eff}$. Equating the two estimates yields the *effective sample size*, n_{eff}, as

$$n_{eff} := mn \, \frac{\widehat{\mathrm{Var}}^{+}(\psi \mid \mathbf{X})}{B} \tag{17}$$

Generally $n_{eff} < mn$ since samples within each sequence will be *auto-correlated*. We can therefore interpret n_{eff}/mn as a measure of the simulation *efficiency*. Note that if m is small then B will have high sampling variability in which case n_{eff} will be a crude estimate. In this case we might prefer to report $\min(n_{eff}, mn)$.

---

5 Further Examples and Applications

Inference in (complex) Bayesian models is typically done via: (1) sampling from the posterior using MCMC algorithms such as Metropolis-Hastings, Gibbs sampling or *auxiliary variable* MCMC methods such as *slice sampling* and *Hamiltonian Monte-Carlo* (HMC) or (2) approximating the posterior with more tractable distributions – a process known as *deterministic inference*. These deterministic inference methods include *variational Bayes* and *expectation propagation*.

Over the past couple of decades software such as `WinBugs`, `OpenBugs` and `JAGS` have been made freely available. These software packages use Gibbs sampling to simulate from the posterior and also perform various convergence diagnostics. More recently `STAN` has been developed (mainly by researchers at Columbia University) and this package largely relies on (a version of) HMC[6] to overcome the slow mixing / convergence of Gibbs for very complex models. Because of the development of such software as well as increased computing power, Bayesian models are now ubiquitous throughout the sciences. In this section we describe some additional applications while Section 5.1 shows how Gibbs sampling arises naturally when performing inference in directed graphical models.

Example 9 (Data Augmentation for Binary Response Regression with a Probit Link[7])
We have binary response variables $\mathbf{y} := (y_1, \ldots, y_m)$ and corresponding to the i^{th} response we have a covariate vector $\mathbf{x}_i := (x_{i1}, \ldots, x_{ik})$. Like logistic regression, the probit regression model is a generalized linear model (GLM). The probability that $y_i = 1$ satisfies

$$p_i := P(y_i = 1) = \Phi\left(x_{i1}\beta_1 + \cdots + x_{ik}\beta_k\right)$$

where Φ is the CDF of the standard normal distribution. The goal is to estimate $\boldsymbol{\beta} := (\beta_1, \ldots, \beta_k)$ and this can be done using standard GLM software using the 'probit' *link function*. We will use a Bayesian approach here, however. If we assume a prior $\pi(\boldsymbol{\beta})$ on $\boldsymbol{\beta}$ then the posterior density is given by

$$\begin{aligned} g(\boldsymbol{\beta} \mid \mathbf{y}) &\propto \pi(\boldsymbol{\beta}) \prod_{i=1}^{n} p_i^{y_i} (1-p_i)^{1-y_i} \\ &= \pi(\boldsymbol{\beta}) \prod_{i=1}^{n} \Phi\left(\mathbf{x}_i^{\top}\boldsymbol{\beta}\right)^{y_i} \left(1 - \Phi\left(\mathbf{x}_i^{\top}\boldsymbol{\beta}\right)\right)^{1-y_i}. \end{aligned} \tag{18}$$

It is not clear how to generate samples of $\boldsymbol{\beta}$ from the posterior in (18) in a Gibbs sampling framework. A clever way to resolve this problem is to define *latent*, i.e. unobserved, variables

$$z_i := x_{i1}\beta_1 + \cdots + x_{ik}\beta_k + \epsilon_i$$

[6] See Appendix A.3 for a description of HMC.
[7] This example is taken from *Bayesian Analysis of Binary and Polychotomous Response Data* by Albert and Chib (1993).

where the ϵ_i's are IID $\mathsf{N}(0,1)$ for $i = 1, \ldots, n$. Note that (why?) $p_i = P(z_i > 0) = \Phi(\mathbf{x}_i^\top \boldsymbol{\beta})$. We can now regard the problem as a *missing data* problem where instead of observing the z_i's we only observe the indicators $y_i := 1_{\{z_i > 0\}}$ and our posterior distribution is now over $\boldsymbol{\beta}$ and $\mathbf{z} := (z_1, \ldots, z_n)$. This posterior is given by

$$\begin{aligned} g(\boldsymbol{\beta}, \mathbf{z} \mid \mathbf{y}) &\propto g(\boldsymbol{\beta}, \mathbf{z}, y) \\ &= \pi(\boldsymbol{\beta}) \prod_{i=1}^{n} \left[1_{\{z_i > 0\}} 1_{\{y_i = 1\}} + 1_{\{z_i \leq 0\}} 1_{\{y_i = 0\}} \right] \phi(z_i \,;\, \mathbf{x}_i^\top \boldsymbol{\beta}, 1) \end{aligned} \tag{19}$$

where $\phi(\cdot \,;\, \mu, \sigma^2)$ denotes the PDF for a normal random variable with mean μ and variance σ^2. The posterior in (19) is in a particularly convenient form for Gibbs sampling if we assume $\pi(\boldsymbol{\beta}) \equiv 1$, i.e. a uniform prior on $\boldsymbol{\beta}$. In that case we can use a block Gibbs sampler where we simulate successively from $g(\boldsymbol{\beta} \mid \mathbf{z}, \mathbf{y})$ and $g(\mathbf{z} \mid \boldsymbol{\beta}, \mathbf{y})$. When $\pi(\boldsymbol{\beta}) \equiv 1$ it is relatively(!) easy to see that

$$g(\boldsymbol{\beta} \mid \mathbf{z}, \mathbf{y}) \sim \mathsf{MVN}_k \left((\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{z},\, (\mathbf{X}^\top \mathbf{X})^{-1} \right) \tag{20}$$

where $\mathsf{MVN}_k(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a k-dimensional multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, and $\mathbf{X}$ is the design matrix for the problem.

Exercise 10 *Justify (20) and then explain how we can also simulate from* $g(\mathbf{z} \mid \boldsymbol{\beta}, \mathbf{y})$.

As a specific example, we consider the data-set on the *Donner party*, a group of wagon trail emigrants who struggled to cross the Sierra Nevada mountains in California in 1846-47 with the result being that a large number of them starved to death. We are interested in estimating the model

$$P(y_i = 1) = \Phi\left(\beta_0 + \beta_1 \mathsf{Male}_i + \beta_2 \mathsf{Age}_i\right) \tag{21}$$

where $y_i = 1$ denotes the death of the i^{th} person in the party and $y_i = 0$ denotes their survival. We have two covariates, Male (1 for males, 0 for females) and Age (in years). Figure 1 displays estimated percentile survival

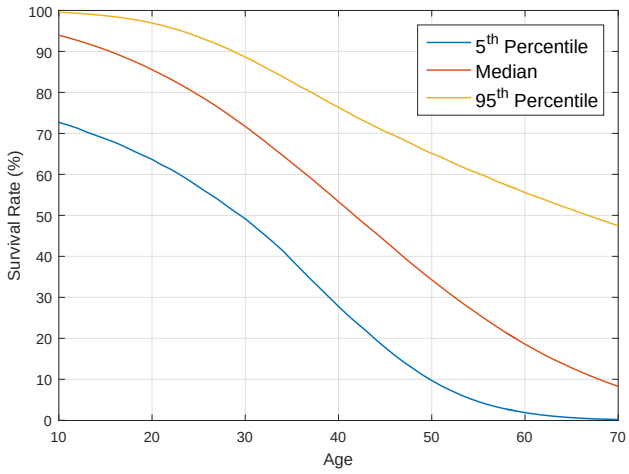

Figure 1: Median, 5th and 95% percentile survival rates as a function of age for men

rates for men of various ages based in the Donner party. These quantities were computed by running the block Gibbs sampler as described above and using the $\boldsymbol{\beta}$ samples (after convergence had been diagnosed) together with (21).

In addition to demonstrating ing the power of *data augmentation*, it is also worth noting that the survival curves of Figure 1 would be extremely difficult to construct in a non-Bayesian framework, especially when there are relatively few data-points so that large n asymptotic results do not apply. ∎

Remark 2 *In non-Bayesian problems with latent / hidden variables it is very common to estimate parameters via the* **EM algorithm**. *In Bayesian versions of these problems it is typically the case that a 2-stage Gibbs sampler can easily be implemented. The first stage simulates the unknown parameters given the data (observed and hidden) while the second stage simulates the unobserved data given the parameters and observed data.*

Example 10 (Asset Allocation with Views)
In finance one can use sophisticated statistical / time series techniques to construct an *objective* model of security returns or risk factors. Assuming such a model has been constructed, we let $\mathbf{X}_{t+1}$ denote the (random) change in *risk factors* between dates t and $t+1$. Then all security returns from period t to $t+1$ depend on $\mathbf{X}_{t+1}$ only plus idiosyncratic noise. We let $f(\cdot)$ then denote the (objective) distribution of $\mathbf{X}_{t+1}$ based on all information available in the market place at date t. The investor would like to construct an optimal portfolio based on the distribution $f(\cdot)$ as well as her own **subjective** views of what will happen in the market between dates t and $t+1$.

Question: How can she do this?

Solution: Let $\mathbf{V} = g(\mathbf{X}_{t+1}) + \boldsymbol{\epsilon}$ be a random vector where $g(\cdot)$ is a function representing how these views depend on $\mathbf{X}_{t+1}$ and $\boldsymbol{\epsilon}$ is a noise vector reflecting how certain the investor is in her views. We assume $\boldsymbol{\epsilon}$ is independent of $\mathbf{X}_{t+1}$ with distribution MVN$(\mathbf{0}, \boldsymbol{\Sigma})$ say. Suppose now that the investor believes that $g(\mathbf{X}_{t+1})$ will equal $\mathbf{v}$. Then we construct the conditional distribution of $\mathbf{X}_{t+1}$ given $\mathbf{V} = \mathbf{v}$ and obtain

$$\begin{aligned} f(\mathbf{X}_{t+1} \mid \mathbf{V} = \mathbf{v}) &\propto f(\mathbf{X}_{t+1}, \mathbf{v}) \\ &= f(\mathbf{v} \mid \mathbf{X}_{t+1})\, f(\mathbf{X}_{t+1}) \end{aligned} \tag{22}$$

where $f(\mathbf{v} \mid \mathbf{X}_{t+1})$ is easily computed given the (user-specified) distribution of $\boldsymbol{\epsilon}$ and $f(\mathbf{X}_{t+1})$ is the objective distribution of the risk-factor returns discussed above. We can use MCMC to simulate many samples from (22) which can then be used to construct an optimal portfolio.

Note that we obtain the famous **Black-Litterman** model when $\mathbf{X}_{t+1}$ is the vector of security returns, $g(\cdot)$ is linear, and all distributions are multivariate normal. In this case the posterior can be calculated analytically. ∎

Example 11 (Optimization via MCMC and Code Breaking)
One day[8] a psychologist from California's state prison system showed up at the consulting service of Stanford's Statistics department. The problem was to decode a collection of coded messages – one such sample is displayed in the figure below. A student in the consulting service guessed (correctly) that it was a simple *substitution cipher* so that each symbol represented a letter, number, punctuation mark or a space.

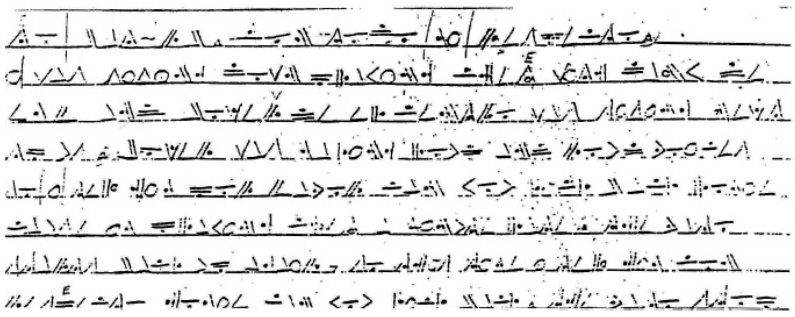

Figure taken from "The Markov Chain Monte Carlo Revolution", by Persi Diaconis in the *Bulletin of the American Mathematical Society* (2008).

[8]This example is based on the paper "The Markov Chain Monte Carlo Revolution", by Persi Diaconis in the *Bulletin of the American Mathematical Society* (2008).

The goal then was to crack this cipher and find the function

$$f : \{\text{code space}\} \to \{\text{usual alphabet}\}. \tag{23}$$

The following solution approach was adopted:

1. Find a text, e.g. *War and Peace*, and record the first-order transitions, i.e. the proportion of consecutive text symbols from x to y. This yields a matrix $M(x, y)$ of transition frequencies.
2. We can then define a *plausibility* to any function $f(\cdot)$ via

$$\text{Pl}(f) := \prod_i M\left(f(s_i), f(s_{i+1})\right)$$

   where s_i runs over all the symbols that appear in the coded message. The idea here is that functions with high values of $\text{Pl}(f)$ are good candidates for the decryption code in (23).
3. We therefore search for maximal $f(\cdot)'s$ by running the following MCMC algorithm:
   - Start with an initial guess f.
   - Compute $\text{Pl}(f)$.
   - Change to f_* by making a random transposition of the values f assigns to two symbols.
   - Compute $\text{Pl}(f_*)$; if this is larger than $\text{Pl}(f)$ accept f_*.
   - If not, flip a coin where the probability of heads is $\text{Pl}(f_*)/\text{Pl}(f)$. If the coin toss comes up heads accept f_*. Otherwise stay at f.

Exercise 11 *What type of MCMC algorithm is described in Step 3? Explain what each step is doing.*

By running the algorithm for sufficiently many iterations and possibly from randomly chosen starting points we hope that the algorithm will identify regions of high probability, i.e plausibility.

∎

Example 12 (Topic Modeling and LDA)
Latent Dirichlet Allocation (LDA) is a hierarchical model used to model collections of text documents. Each document is modeled as a mixture of topics and each topic is then defined as a distribution over the words in the vocabulary / dictionary. Specifically, we assume there are a total of K topics, a total of D documents and a total of M words in the dictionary with words numbered from 1 to M. The LDA topic model is then obtained in the following *generative* fashion:

1. A topic mixture $\boldsymbol{\theta}_d$ for each document is drawn independently from a $\text{Dir}_K(\alpha\mathbf{1})$ distribution, where $\text{Dir}_K(\boldsymbol{\phi})$ is a Dirichlet distribution over the K-dimensional simplex with parameters $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_K)$.
2. Each of the K topics $\{\boldsymbol{\beta}_k\}_{k=1}^K$ are drawn independently from a $\text{Dir}_M(\gamma\mathbf{1})$ distribution.
3. Then for each of the $i = 1 \ldots, N_d$ words in document d, an assignment variable z_i^d is drawn from $\text{Mult}(\boldsymbol{\theta}_d)$.
4. Conditional on the assignment variable z_i^d, word i in document d, denoted as w_i^d, is drawn independently from $\text{Mult}(\boldsymbol{\beta}_{z_i^d})$

This is a hierarchical model and it's straightforward to write out the joint distribution of all the data. Only the w_i^d's are observed, however, and so we need to use the corresponding conditional distribution to learn the topic mixtures for each document, the K topic distributions and the latent variables z_i^d. This is typically done via Gibbs sampling or variational Bayes. Figure 2 displays some of the main topics found in a sample from the conditional distribution.

∎

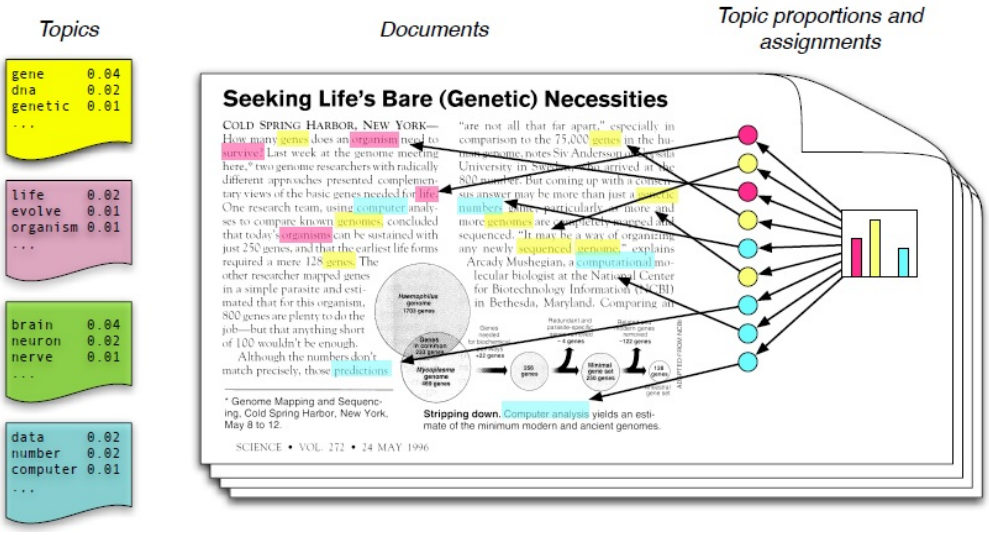

Figure 2: Taken from *Introduction to Probabilistic Topic Models* by D.M. Blei (2011).

5.1 An Extremely Brief Detour on Directed Graphical Models

Graphical models are used to describe *dependence / independence* relationships between random variables and these models are now very popular in the machine learning community. There are two main types of graphical models:

1. **Undirected graphical models** which are also known as **Markov networks**.
2. **Directed graphical models** which are also known as **Bayesian networks**. **Belief networks**, also known as **directed acyclic graphs (DAG's)**, are an important subclass.

A graphical model contains nodes and (directed or undirected) edges. Each node in the graph corresponds to a random variable with the edge structure of the graph (and edge direction in case of directed graphs) determining the various *conditional independence / dependence relationships* between the random variables. These relationships often enable inference, e.g. computation of conditional distributions, to be performed very efficiently. We only consider directed graphical models here.

Directed Acyclic Graphs (DAGs)

There are no *directed cycles* in a DAG. This implies there is a *node numbering* such that any edge in the graph is always directed from a node to a higher numbered node. Many efficient algorithms exist for performing inference in belief networks.

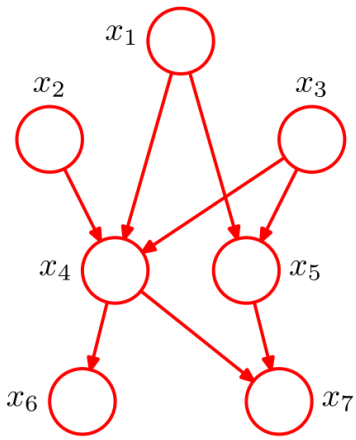

Figure 8.2 ***Pattern Recognition and Machine Learning*** **by C. Bishop**

Note the ordering of nodes in the DAG of Figure 8.2 which was taken from Bishop's *PRML*. This ordering can be used to write

$$\begin{aligned} p(x_1, x_2, \ldots, x_7) &= p(x_7 \mid x_4, x_5) \cdot p(x_6 \mid x_4) \cdot \\ & \quad p(x_5 \mid x_1, x_3) \cdot p(x_4 \mid x_1, x_2, x_3) \\ & \quad p(x_3) \cdot p(x_2) \cdot p(x_1). \end{aligned}$$

More generally for any DAG we have

$$p(\mathbf{x}) = \prod_{k=1}^{K} p(x_k \mid \mathsf{pa}(x_k)) \tag{24}$$

where $\mathsf{pa}(x)$ denotes the "parents" of node x_k.

Note that it is by definition that (24) must hold for any DAG representing $p(\mathbf{x})$. Specifically, the DAG structure models the fact for all k we have $p(x_k \mid x_1, \ldots, x_{k-1}) = p(x_k \mid \mathsf{pa}(x_k))$. It's easy (why?) to simulate from a DAG using (24). Indeed simulating using the representation in (24) is called **ancestral sampling**. It is not so easy, however, to simulate from the joint conditional distribution when some nodes are observed but we will see that *Gibbs sampling* is easy to implement in that case.

Using Gibbs Sampling to Deal with Evidence in a Belief Network

Suppose now that x_3, x_5 and x_6 have been observed and we want to compute the conditional distribution of the unobserved variables. Using (24) this conditional distribution satisfies

$$\begin{aligned} p(x_1, x_2, x_4, x_7 \mid x_3, x_5, x_6) &= \frac{p(x_1, x_2, x_3, x_4, x_5, x_6, x_7)}{p(x_3, x_5, x_6)} \\ &= \frac{p(x_1, x_2, x_3, x_4, x_5, x_6, x_7)}{\sum_{x_1, x_2, x_4, x_7} p(x_1, x_2, x_3, x_4, x_5, x_6, x_7)} \\ &= \frac{\prod_{k=1}^{7} p(x_k \mid \mathsf{pa}(x_k))}{\sum_{x_1, x_2, x_4, x_7} \prod_{k=1}^{7} p(x_k \mid \mathsf{pa}(x_k))} \end{aligned} \tag{25}$$

where x_3, x_5 and x_6 are "clamped" at their observed values in (25). Computing the normalizing factor, i.e. the denominator, in (25) can be computationally demanding — especially for very large DAGs. Note also that the ordering of the original DAG (with no observed variables) is now lost. e.g. x_1 and x_3 are no longer independent once x_5 has been observed.

Exercise 12 *Can we use still ancestral sampling to simulate from* $p(x_1, x_2, x_4, x_7 \mid x_3, x_5, x_6)$*? If so, is it efficient?*

In fact we can simulate efficiently from $p(x_1, x_2, x_4, x_7 \mid x_3, x_5, x_6)$ using Gibbs sampling. To see this note that at each step of the Gibbs sampler we need to simulate from $p(x_i \mid \mathbf{x}_{-i})$ where any observed values in $\mathbf{x}_{-i}$ are clamped at these values throughout the simulation. But it's easy to see (why?) that

$$p(x_i \mid \mathbf{x}_{-i}) = \frac{1}{Z} p(x_i \mid pa(x_i)) \prod_{j \in ch(i)} p(x_j \mid pa(x_j))$$

where $pa(x_i)$ and $ch(i)$ are the parent and children nodes, respectively, of x_i, and Z is the (usually easy to compute) normalization constant

$$Z = \sum_{x_i} p(x_i \mid pa(x_i)) \prod_{j \in ch(i)} p(x_j \mid pa(x_j)).$$

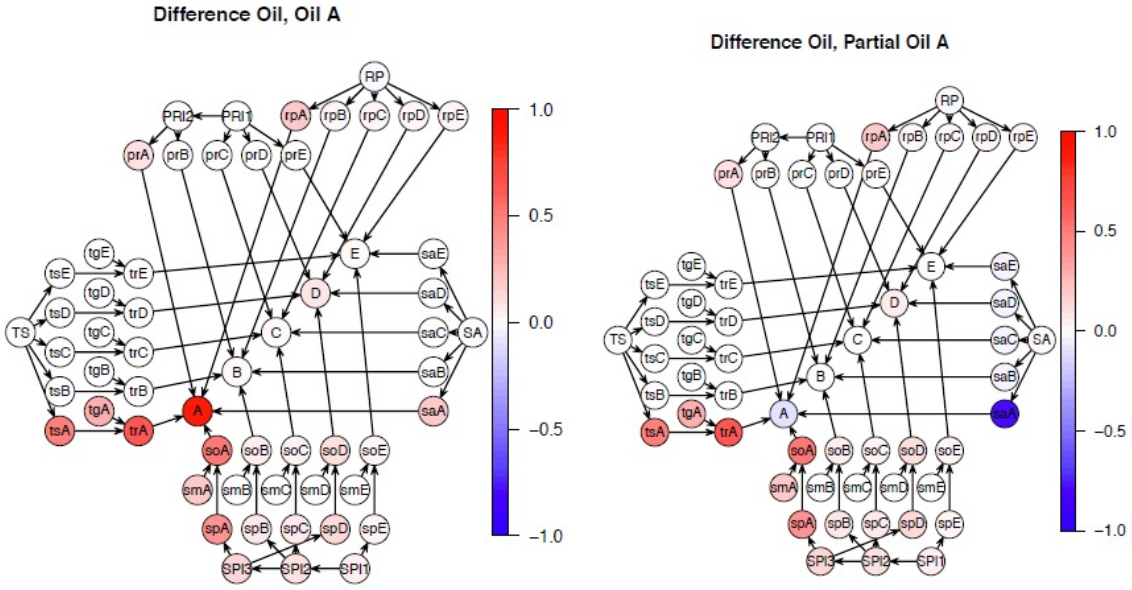

Figure 4: Difference between conditional probabilities given evidence and prior probabilities. Evidence is observed in prospect A, and follows the explanations in section 3.4 . Figure shows the effect of Evidence 1 (left) and 2 (right).

Figure 3: Taken from "Strategies for Petroleum Exploration Based on Bayesian Networks: a Case Study", by Martinelli et al. (2012).

Note that $x_i \in pa(x_j)$ for each $j \in ch(i)$ and so the product term in the above expression for Z is required. The parents of x_i, the children of x_i and the parents of the children of x_i are known collectively as the **Markov blanket** of x_i.

Exercise 13 *When using a Gibbs sampler to simulate from a DAG given some nodes have been observed, is the sampler guaranteed to succeed? If not, what can go wrong?*

Example 13 (Oil Exploration and Inference Using a DAG)
A directed graphical model is used to model the geology of a particular area below the seabed of the North Sea. This geology is complex and locating oil requires both exploration *and* inference. A specific example of such an oil exploration network is displayed in Figure 3. A decision has been made to drill at node **A** and the figures display the changes in probabilities of oil being present at every other node conditional on:

(i) Oil being found at `A` (left-hand heat-map).

(ii) Only partial oil being found at A (right-hand heat-map).

The probabilities, and therefore all of the changes in probabilities, can be estimated using Gibbs sampling as described above. (It's worth mentioning that for relatively small networks, there are algorithms, e.g. the **junction tree** algorithm, that can compute these (conditional) probabilities exactly.) ∎

A Appendix

We briefly discuss a few other important topics in Bayesian modeling and MCMC here.

A.1 Bayesian Model Checking

After (successfully) confirming the stationarity of the Markov chains, we can use the samples to estimate various quantities of interest. But often this is just part of a bigger analysis. In particular we often need to: (1) *assess* the model's performance and (2) *choose* among competing models. There are many ways to assess model performance including:

1. Comparing posterior distributions of parameters to domain knowledge.
2. Simulating samples from the posterior predictive distribution and checking them for "reasonableness". We can do this by first simulating $\boldsymbol{\theta}$ from the posterior distribution (we already have these samples from the MCMC!) and then simulating $\mathbf{X}_{rep} \mid \boldsymbol{\theta}$.
3. **Posterior predictive checking:** in this case we design test statistics of interest and compare their posterior predictive distributions (using simulated samples) to observed values of these test statistics. This can be viewed as a form of internal model validation.

Example 14 (From *Bayesian Data Analysis* by Gelman et al.)
Consider a sequence of coin tosses $\mathbf{y} = [y_1, \ldots, y_n]$. We model them as a specified number of IID Bernoulli trials with a uniform prior on the probability of *heads*, θ. If we let $s := \sum y_i$ then the posterior is given by

$$\begin{aligned} p(\theta \mid \mathbf{y}) &\propto p(\mathbf{y} \mid \theta) p(\theta) \\ &= \theta^s (1-\theta)^{n-s} \end{aligned}$$

which we recognize as the Beta $(\sum y_i + 1, n - \sum y_i + 1)$ distribution. Suppose now that the data was obtained in the following order

$$y = [1\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$$

so $n = 20$ and $s = 7$.

Questions: Is this a good model? Do the data look IID (given θ) as we have assumed?

We first note that the sequence is strongly *autocorrelated* with $T(\mathbf{y}) = 3$ where $T(\cdot)$ counts the number of switches between 0 and 1. So let's simulate m samples $T(\mathbf{y}_1^{rep}), \ldots, T(\mathbf{y}_m^{rep})$ from the posterior predictive distribution and compare them with $T(\mathbf{y})$. The results are displayed in Figure 4 below where we took $m = 10k$ and found only 2.8% of the samples were less than or equal to $T(\mathbf{y}) = 3$. This constitutes pretty strong evidence against the model, in particular against the assumption of IID observations given θ. Posterior predictive checks are a form of *internal model validation* and in this case suggests the model is inadequate and should be improved / expanded. ∎

Bayesian Data Analysis (BDA) by Gelman et al. should be consulted for a far more detailed introduction to model checking as well as many more examples.

A.2 Bayesian Model Selection

Suppose now that we have several "good" models that have "passed" various posterior predictive checks etc. How should we pick the "best" model? There are also several approaches to this model selection problem:

1. **Information criteria** approaches that estimate an in-sample error and penalize the effective number of parameters, p_D. Two common criteria are:

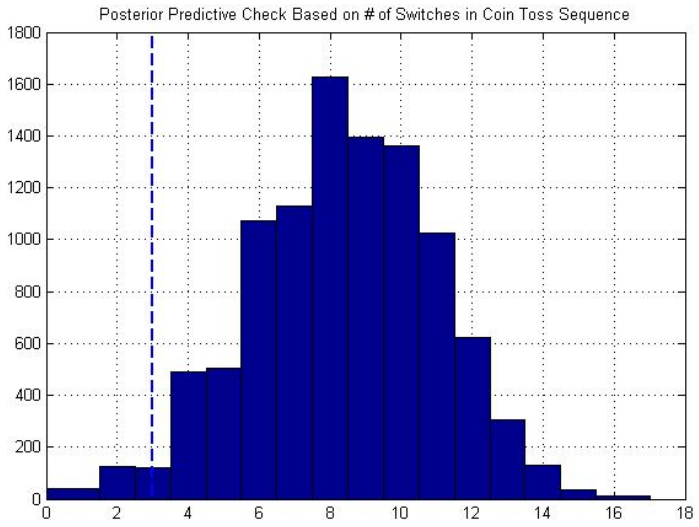

Figure 4: Posterior predictive checking

(i) The *deviance information criterion (DIC)*. This is only suitable for certain types of Bayesian models.

(ii) The *Watanabe-Akaike information criterion (WAIC)*. This is a recently developed criterion and is more generally applicable than DIC. It is not suitable, however, for models where the data is dependent (given θ) like time-series and spatial models.

Note that p_D is a random variable that depends on the data and it's estimated differently for DIC and WAIC. When comparing models, a smaller DIC or WAIC is "better". Both DIC and WAIC are easily estimated from the output of an MCMC which is a useful feature given the computational demands of Bayesian modeling.

2. **Bayesian cross-validation** where the data is divided into K folds. The error on each fold is computed by fitting the model on the remaining $K-1$ folds. The error can be computed using either of:

   (i) The *mean-squared prediction error* which requires the predicted values of the hold-out data. We can use the posterior predictive mean which can often be estimated from MCMC.

   (ii) The $\log$ posterior predictive distribution evaluated at the hold-out data.

   Cross-validation can clearly be computationally very demanding.

3. **Bayes factors** can also be useful when choosing among competing models. Specifically, given two models H_1 and H_2, the Bayes factor, $B(H_2; H_1)$, is

$$B(H_2; H_1) := \frac{p(\mathbf{X} \mid H_2)}{p(\mathbf{X} \mid H_1)} = \frac{\int_{\boldsymbol{\theta}_2} p(\mathbf{X} \mid \boldsymbol{\theta}_2, H_2) p(\boldsymbol{\theta}_2 \mid H_2)\, d\boldsymbol{\theta}_2}{\int_{\boldsymbol{\theta}_1} p(\mathbf{X} \mid \boldsymbol{\theta}_1, H_1) p(\boldsymbol{\theta}_1 \mid H_1)\, d\boldsymbol{\theta}_1} \tag{26}$$

   Note that the Bayes factor is not defined if the priors $p(\boldsymbol{\theta}_i \mid H_i)$ are not *proper*. In general we need to estimate the two integrals in (26) in order to estimate $B(H_2; H_1)$.

   Bayesian Model Averaging (BMA) is a related technique that performs inference using a weighted average of several "good" models with the weights computed via Bayes factors.

It is perhaps worth emphasizing that Bayesian methods and classical *frequentist* methods differ[9] significantly from each other on the topic of model comparison and selection. In contrast, Bayesian and frequentist approaches often lead to similar results when evaluating a *fixed and given* model.

[9]See, for example, Chapters 28 and 37 of David MacKay's excellent text *Information Theory, Inference, and Learning Algorithms* which is freely available online from Cambridge University Press.

A.3 Hamiltonian Monte-Carlo

A real concern with MCMC methods is that the Markov chains move through all areas of significant probability. This is guaranteed in theory but i*n practice* too many iterations may be required. Consider, for example, a Metropolis-Hastings algorithm with a *local* proposal distribution, i.e. a proposal that's unlikely to propose a candidate point x_{t+1} that's far from x_t. If the target distribution has many modes or "islands" of high density, then it will take a long time to move from one island to another. But if we use a *global* proposal distribution, i.e. one with very large variance, then the chance of landing on a high-density island is small[10]. *Auxiliary variable* MCMC methods such as *Hamiltonian Monte-Carlo* (HMC) or the *slice sampler* have been developed to address these concerns. These latter methods have become very popular in recent years and (with Gibbs sampling) have begun to render (basic) Metropolis-Hastings almost obsolete. In this subsection we will discuss the HMC approach, an MCMC method for continuous variables. It makes non-local jumps possible so that we can more easily jump from one mode to another. To begin, we write the target distribution as

$$p(\mathbf{x}) = \frac{1}{Z_x} e^{H_x(\mathbf{x})}$$

where as usual Z_x is unknown. We now introduce a new *auxiliary* variable / vector $\mathbf{y}$ with

$$p(\mathbf{y}) = \frac{1}{Z_y} e^{H_y(\mathbf{y})}.$$

We typically choose $\mathbf{y}$ to be Gaussian so that $H_y(\mathbf{y}) = -\frac{1}{2}\mathbf{y}^\top\mathbf{y}$. We also assume $\mathbf{x}$ and $\mathbf{y}$ are independent so that

$$p(\mathbf{x},\mathbf{y}) = p(\mathbf{x})p(\mathbf{y}) = \frac{1}{Z_x Z_y} e^{H_x(\mathbf{x})+H_y(\mathbf{y})} = \frac{1}{Z} e^{H(\mathbf{x},\mathbf{y})}$$

where $Z := Z_x Z_y$ and $H(\mathbf{x},\mathbf{y}) := H_x(\mathbf{x}) + H_y(\mathbf{y})$. The goal is to define an MCMC algorithm for generating samples of $(\mathbf{x},\mathbf{y})$ with the stationary distribution $p(\mathbf{x},\mathbf{y})$. Then once stationarity is reached we can simply discard the $\mathbf{y}$ samples. The *"trick"* is to define the proposal distribution so that we can easily jump from one mode (of $p(\mathbf{x})$) to another.

We can achieve this as follows: given a current sample $(\mathbf{x},\mathbf{y})$ we:

1. Simulate $\mathbf{y}'$ from $p(\mathbf{y})$
2. And then simulate $\mathbf{x}'$ from $p(\mathbf{x} \mid \mathbf{y}')$ using a Metropolis-Hastings sampler.

We want the new sample $(\mathbf{x}',\mathbf{y}')$ to satisfy

$$H(\mathbf{x}',\mathbf{y}') \approx H(\mathbf{x},\mathbf{y})$$

so that it will be accepted with high probability in the M-H algorithm. We can achieve this by moving (approximately) along a contour of H from $(\mathbf{x},\mathbf{y})$ to $(\mathbf{x}',\mathbf{y}')$ where $(\mathbf{x}',\mathbf{y}') = (\mathbf{x}+\Delta\mathbf{x},\, \mathbf{y}+\Delta\mathbf{y})$. A first-order Taylor approximation implies

$$\begin{aligned} H(\mathbf{x}',\mathbf{y}') &= H(\mathbf{x}+\Delta\mathbf{x},\, \mathbf{y}+\Delta\mathbf{y}) \\ &\approx H(\mathbf{x},\mathbf{y}) + \nabla_x H_x(\mathbf{x})^\top \Delta\mathbf{x} + \nabla_y H_y(\mathbf{y})^\top \Delta\mathbf{y} \end{aligned} \tag{27}$$

To move (approximately) along a contour of H we would like to set the sum of the last two terms in (27) to 0. This is a 1-dimensional constraint so many solutions are possible. To identify a particular solution it is customary to use so-called **Hamiltonian dynamics** whereby

$$\Delta\mathbf{x} := \epsilon\nabla_y H(\mathbf{y}) \qquad \text{and} \qquad \Delta\mathbf{y} := -\epsilon\nabla_x H(\mathbf{x})$$

[10] These observations are the "solution" to the questions we posed in Exercise 6.

so that $H(\mathbf{x}', \mathbf{y}') \approx H(\mathbf{x}, \mathbf{y})$ as desired. We take L such Hamiltonian steps all with the same value of ϵ which is drawn randomly according to

$$\epsilon = \begin{cases} +\epsilon_0, & \text{with prob. } 0.5 \\ -\epsilon_0, & \text{with prob. } 0.5 \end{cases}$$

so that the proposal distribution, $Q(\cdot \mid \cdot)$, is *symmetric*.

The variable $\mathbf{x}$ has the interpretation of *position* and the auxiliary variable $\mathbf{y}$ has the interpretation of *momentum*. Typically, $\mathbf{y}$ has the same dimension as $\mathbf{x}$ so there is one momentum variable for each space variable. The Hamiltonian dynamics, i.e movement along a contour of H, can be implemented in a more sophisticated way than (27) via so-called **leapfrog discretization**. See, for example, Bishop's *PRML* for details. In order to implement the algorithm we need to specify the parameters L and ϵ_0. The success[11] of the algorithm is quite sensitive to these choices. A high-level version of the HMC algorithm is given in Algorithm 27.4 below which is taken from Barber's *BRML*.

Algorithm 27.4 Hybrid Monte Carlo sampling

1: Start from $\mathbf{x}^1$
2: **for** $i = 1$ to L **do**
3: Draw a new sample $\mathbf{y}$ from $p(\mathbf{y})$.
4: Choose a random (forwards or backwards) trajectory direction.
5: Starting from $\mathbf{x}^i, \mathbf{y}$, follow Hamiltonian dynamics for a fixed number of steps, giving a candidate $\mathbf{x}', \mathbf{y}'$.
6: Accept the candidate $\mathbf{x}^{i+1} = \mathbf{x}'$ if $H(\mathbf{x}', \mathbf{y}') > H(\mathbf{x}, \mathbf{y})$, otherwise accept it with probability $\exp(H(\mathbf{x}', \mathbf{y}') - H(\mathbf{x}, \mathbf{y}))$.
7: If rejected, we take the sample as $\mathbf{x}^{i+1} = \mathbf{x}^i$.
8: **end for**

Figure 27.9 (also taken from from Barber's *BRML*) displays HMC in action in a one-dimensional example where the distribution is bimodal. The distribution becomes bivariate with the addition of the auxiliary variable y and we see in part (c) how the Hamiltonian dynamics enables the sampler to easily cross between the two islands of high probability, i.e. the two modes.

A.4 Empirical Bayes

We now briefly discuss the empirical Bayes approach to the selection of prior distributions. Note that a full Bayesian approach first specifies a prior (including hyper-priors as necessary), then specifies the likelihood and finally combines the two via Bayes Theorem to construct the posterior. What is relevant for our discussion here is that in a full Bayesian approach the data plays no role in specifying the prior. This is *not* the case with empirical Bayes as we shall see in the following[12] example.

Example 15 (Robbins' Formula)
Table 1 displays one year's worth of claims data for a European insurance company. There were a total of 9461 policy holders of whom 7840 made 0 claims, 1317 made 1 claim, 239 made 2 claims etc. We are concerned with estimating the number of claims each policy holder will make *next* year. Towards this end we let X_k denote the number of claims made in a single year by policy holder k and we assume it follows a Poisson distribution with parameter θ_k so that

$$P(X_k = x) = p_{\theta_k}(x) := \frac{e^{-\theta_k}\theta_k^x}{x!}, \qquad x = 0, 1, 2, \ldots. \tag{28}$$

[11] Indeed improved versions of Hamiltonian MC choose these parameters *adaptively* and these versions are implemented in the new and popular `STAN` software which was developed mainly by a team at Columbia University.

[12] Our brief development of empirical Bayes follows Section 6.1 of the recent text *Computer Age Statistical Inference: Algorithms, Evidence, and Data Science* by Efron and Hastie. The rest of that chapter as well as Chapter 21 and other more advanced applications elsewhere in the text demonstrate the now widespread applicability of the empirical Bayes approach.

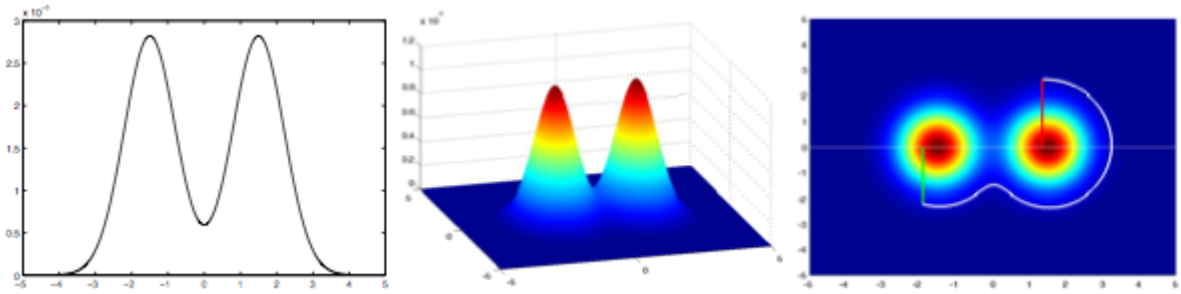

Figure 27.9 (Taken from Barber's *BRML*): Hybrid Monte Carlo. (a): Multi-modal distribution $p(x)$ for which we desire samples. (b): HMC forms the joint distribution $p(x)p(y)$ where $p(y)$ is Gaussian. (c): This is a plot of (b) from above. Starting from the point x, we first draw a y from the Gaussian $p(y)$, giving a point (x, y), given by the green line. Then we use Hamiltonian dynamics (white line) to traverse the distribution at roughly constant energy for a fixed number of steps, giving x', y'. We accept this point if $H(x', y') > H(x, y')$ and make the new sample x' (red line). Otherwise this candidate is accepted with probability $\exp(H(x', y') - H(x, y'))$. If rejected the new sample x' is taken as a copy of x.

Claims x	0	1	2	3	4	5	6	7
Counts y_x	7840	1317	239	42	14	4	4	1
Formula (31)	.168	.363	.527	1.33	1.43	6.00	1.25	-
Gamma MLE	.164	.398	.633	.87	1.10	1.34	1.57	-

Table 1: Counts y_x of number of claims x made in a single year by 9461 automobile insurance policy holders. Robbins' formula (31) estimates the number of claims expected in a succeeding year, for instance 0.168 for a customer in the $x = 0$ category. Parametric maximum likelihood analysis based on a gamma prior gives less noisy estimates.

We also assume that the θ_k's are random with prior $g(\theta)$. Consider now an individual customer who made x claims last year. Then we have (why?)

$$\mathbb{E}[\theta \mid x] = \frac{\int_0^\infty \theta p_\theta(x) g(\theta)\, d\theta}{\int_0^\infty p_\theta(x) g(\theta)\, d\theta}. \tag{29}$$

Note that (29) would also yield the expected number of claims made by the customer next year since (why?) $\mathbb{E}[\theta \mid x] = \mathbb{E}[X \mid x]$. So formula (29) is what the insurance company needs to answer its question *if* it already knows the prior $g(\cdot)$. For example, if the company assumes g is Gamma(ν, σ) with ν and σ known, then there is no problem calculating (29). But how would we choose "good" values of ν and σ? A typical Bayesian approach would in fact assume they are unknown and would therefore place a hyper-prior (with known parameters) on (ν, σ). In that case considerably more work would be required to compute g and calculate (29).

Alternatively we can be a little clever! Using (28) and (29) we have

$$\begin{aligned} \mathbb{E}[\theta \mid x] &= \frac{\int_0^\infty \left[e^{-\theta}\theta^{x+1}/x!\right] g(\theta)\, d\theta}{\int_0^\infty \left[e^{-\theta}\theta^{x}/x!\right] g(\theta)\, d\theta} \\ &= \frac{(x+1)\int_0^\infty \left[e^{-\theta}\theta^{x+1}/(x+1)!\right] g(\theta)\, d\theta}{\int_0^\infty \left[e^{-\theta}\theta^{x}/x!\right] g(\theta)\, d\theta} \\ &= (x+1)\frac{f(x+1)}{f(x)} \end{aligned} \tag{30}$$

where $f(x) = \int_0^\infty p_\theta(x) g(\theta)\, d\theta$ is the *marginal* density of X. From (30) it is clear that to answer the insurance company's question we only need $f(\cdot)$ and not $g(\cdot)$. But we have a lot of data and can easily estimate $f(\cdot)$

directly to obtain *Robbins' approximation*

$$\begin{aligned}\widehat{\mathbb{E}}[\theta \mid x] &= (x+1)\frac{\widehat{f}(x+1)}{\widehat{f}(x)} \\ &= (x+1)\frac{y_{x+1}}{y_x} \end{aligned} \tag{31}$$

with y_x denoting the number of observations with x claims. That is, we estimate $f(x)$ with $\widehat{f}(x) = y_x/N$ where $N = 9461$. We see the values of $\widehat{\mathbb{E}}[\theta \mid x]$ in the third row of Table 1. ∎

Note that the values at the end of the third row in Table 1 seem to go awry. This is because formula (31) has become unstable at that point due to the small count numbers in the data for policies that had 5 or more claims. We can help resolve this issue by using a *parametric* empirical Bayesian approach in contrast to the *non-parametric* approach outlined above.

Example 16 (Parametric Empirical Bayes)
We continue on from Example 15 but now we assume that g is Gamma(ν, σ) with

$$g(\theta) = \frac{\theta^{\nu-1}e^{-\theta/\sigma}}{\sigma^{\nu}\Gamma(\nu)}, \qquad \theta \geq 0$$

with (ν, σ) unknown. Instead of placing a (hyper-) prior on (ν, σ) we can estimate them from the data by explicitly computing (how?) the marginal density $f(x)$ which now has parameters ν and σ. We then simply compute[13] the maximum likelihood estimators $\widehat{\nu}$ and $\widehat{\sigma}$ to obtain

$$\widehat{\mathbb{E}}[\theta \mid x] = (x+1)\frac{f_{\widehat{\nu},\widehat{\sigma}}(x+1)}{f_{\widehat{\nu},\widehat{\sigma}}(x)} \tag{32}$$

as our estimator. The fourth row of Table 1 was obtained using (32). ∎

Exercise 14 *Explain how you would compute an explicit expression for $f_{\nu,\sigma}(x)$ In Example 16.*

According to Efron and Hastie, Robbins' formula came as a surprise to the statistical world since $\widehat{\mathbb{E}}[\theta_k \mid x_k]$, unavailable without the prior g, suddenly became available by leveraging the information in data from (a large number of) similar cases. It's interesting to note that many eminent statisticians including Robbins, Fisher, Von Mises and others developed empirical Bayesian estimators but the approach, which was often criticized for being neither Bayesian nor frequentist, is now quite standard and has grown in popularity in the "big-data" era where massive *parallel* data-sets are not quite common.

Section 6.2 of Efron and Hastie describes the first known application of empirical Bayes. It was developed by Ronald Fisher and he solved a *missing-species* problem concerned with estimating the number of butterfly species in Malaysia during World War II. They then go on to describe how the same methods can be (and have been) used to estimate the total number of words in Shakespeare's vocabulary.

[13] We could also use other estimation techniques such as simple moment-matching.