Abstract

Hierarchical models are one of the central tools of Bayesian analysis. They offer many advantages, including the ability to borrow strength to estimate individual parameters and the ability to specify complex models that reflect engineering and physical realities. Markov chain Monte Carlo is a set of algorithms that allow Bayesian inference in a variety of models. We illustrate hierarchical models and Markov chain Monte Carlo in a
Bayesian system reliability example.

Keywords: hierarchical model, system reliability, Markov chain Monte Carlo

1 HIERARCHICAL MODELS

Hierarchical models are one of the central tools of Bayesian analysis. Broadly, Bayesian models have two parts (see eqr081): the likelihood function and the prior distribution. We construct the likelihood function from the sampling distribution of the data, which describes the probability of observing the data before the experiment is performed. After we perform the experiment and observe the data, we can consider the sampling distribution as a function of the unknown parameters. This function is called the likelihood function. The prior distribution describes the uncertainty about the parameters of the likelihood function. We update the prior distribution to the posterior distribution after observing data. We use Bayes’ Theorem to perform the update, which shows that the posterior distribution is computed (up to a proportionality constant) by multiplying the likelihood function by the prior distribution.

Denote the sampling distribution as $f(y \mid \theta)$ and the prior distribution as $g(\theta \mid \alpha)$, where $\alpha$ represents the parameters of the prior distribution (often called hyperparameters). It may be the case that we know $g(\theta \mid \alpha)$ completely, including a specific value for $\alpha$. However, suppose that we do not know $\alpha$, and that we choose to quantify our uncertainty about $\alpha$ using a distribution $h(\alpha)$ (often called the hyperprior). This is the most general form of a hierarchical model.
A hierarchical model has three parts [1]:

1. The observational model for the data.

\[(Y_i \mid \theta) \sim f(y_i \mid \theta), \ i = 1, \ldots, k.\]

2. The structural model for the parameters of the likelihood.

\[(\theta \mid \alpha) \sim g(\theta \mid \alpha).\]

3. The hyperparameter model for the parameters of the structural model.

\[\alpha \sim h(\alpha).\]

While this is the most general form of the hierarchial model, one of its most common applications is in problems where there are multiple parameters that are related by the structure of the problem. Consider the data in Table 1, which are analyzed in [2] and [3]. These data summarize the number of failures of pumps \(s_i\) over a period of time \(t_i\) in several systems \(i\) of the nuclear power plant Farley 1.

What is the appropriate observational model for the data in Table 1? We could assume that each pump has an identical failure rate, \(\lambda\). We could then model the number of failures as \(S_i \sim \text{Poisson}(\lambda t_i), \ i = 1, \ldots, 10\) or equivalently, \(\sum_{i=1}^{10} S_i \sim \text{Poisson}(\lambda \sum_{i=1}^{10} t_i)\).
Table 1: Pump Failures ($t_i$ in thousands of hours)

<table>
<thead>
<tr>
<th>System</th>
<th>$s_i$</th>
<th>$t_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>94.320</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>15.720</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>62.880</td>
</tr>
<tr>
<td>4</td>
<td>14</td>
<td>125.760</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>5.240</td>
</tr>
<tr>
<td>6</td>
<td>19</td>
<td>31.440</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>1.048</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>1.048</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>2.096</td>
</tr>
<tr>
<td>10</td>
<td>22</td>
<td>10.480</td>
</tr>
</tbody>
</table>

The assumption of a common failure rate, however, is quite strong, and may be inappropriate. Suppose that instead of a common failure rate, we consider the failure rate for each pump, $\lambda_i$, as a sample from an overall population distribution. We have no information, other than the data, that distinguishes the failure rates, and we have no ordering or grouping of the rates. Formally, this suggests that we treat the rates as exchangeable, which implies that their joint distribution $g(\lambda_1, \ldots, \lambda_{10})$ is invariant to permutations of the indices of the parameters.

By treating the rates as exchangeable, we propose the following hierarchial model:

1. An observational model for the data.

$$(S_i | \lambda_i) \sim \text{Poisson}(\lambda_i t_i), \ i = 1, \ldots, 10.$$
2. A structural model for the parameters of the likelihood.

\[ (\lambda_i | \alpha, \beta) \sim \text{Gamma}(\alpha, \beta), \ i = 1, \ldots, 10. \]

3. A hyperparameter model for the parameters of the structural model.

\[
\begin{align*}
\alpha & \sim \text{Uniform}(0.0, 5.0) \\
\beta & \sim \text{Gamma}(0.1, 1.0).
\end{align*}
\]

This model allows the pump for each system to have its own failure rate, but it also models each failure rate as coming from a common population distribution. This hierarchical structure allows us to borrow strength for the estimation of each \( \lambda_i \). In particular, the estimation of each \( \lambda_i \) is improved by using the failure data from the other pumps.

2 MARKOV CHAIN MONTE CARLO

Given the hierarchial model for the data in Table 1, we now need to estimate the posterior distributions for \( \lambda_i, i = 1, \ldots, 10, \alpha, \text{ and } \beta. \) The posterior distribution is proportional to

\[
p(\lambda_1, \ldots, \lambda_{10}, \alpha, \beta | s, t) \propto \exp(-\beta - \sum_{i=1}^{10}(\beta + t_i)\lambda_i)(\prod_{i=1}^{10} \lambda_i^{s_i + \alpha - 1})\]

\[\beta^{10\alpha + 0.1 - 1}(\Gamma(\alpha))^{-10} I(\alpha \in (0, 5)).\]
In Bayesian analysis, the posterior distribution provides the “answer;” however, it is not immediately obvious how to use this expression for \( p(\lambda_1, \ldots, \lambda_{10}, \alpha, \beta | s, t) \) to answer questions of interest. In particular, suppose that we would like to know the expected rate of failure for an arbitrary pump from this population given the observed data. Mathematically, we need to calculate

\[
\int \frac{\alpha}{\beta} p(\lambda_1, \ldots, \lambda_{10}, \alpha, \beta | s, t) d\lambda_1 \ldots d\lambda_{10} d\alpha d\beta.
\]

Monte Carlo integration can approximate this integral: we draw a sample \( X^{(j)} = (\lambda_1^{(j)}, \ldots, \lambda_{10}^{(j)}, \alpha^{(j)}, \beta^{(j)}) \), \( j = 1, \ldots, n \) from \( p(\lambda_1, \ldots, \lambda_{10}, \alpha, \beta | s, t) \) and then calculate \( \sum_{j=1}^{n} \alpha^{(j)}/\beta^{(j)} \). Markov chain Monte Carlo (MCMC) provides the algorithms to generate the random sample from the posterior distribution—or more generally, from any distribution, \( \pi(\cdot) \), of interest. The strategy is to construct a Markov chain with \( \pi(\cdot) \) as its stationary distribution.

A Markov chain is a sequence of random variables \( X^{(j)} \), \( j = 0, \ldots \), so that for each \( j \), \( X^{(j)} \) is sampled from a distribution \( P(X^{(j)} | X^{(j-1)}) \). Given \( X^{(j-1)} \), the next state \( X^{(j)} \) depends only on \( X^{(j-1)} \) and not on the rest of the history of the chain. The sequence is called a Markov chain, and \( P(\cdot | \cdot) \) is called the transition kernel of the chain. Subject to certain regularity conditions [4], the chain, regardless of its starting value \( X^{(0)} \), eventually converges (after a burn-in of \( m \) samples) to a unique stationary distribution. As \( j \) increases, the samples \( X^{(j)} \) become dependent samples approximately from \( \pi(\cdot) \).

(For more details on the derivation, see [3].)
Constructing a Markov chain with the appropriate stationary distribution is straightforward using the Metropolis-Hastings algorithm [5]. Suppose that we want to draw $X^{(j+1)}$. We sample a candidate point $Y$ from a proposal distribution $q(\cdot \mid X^{(j)})$. The proposal distribution may depend on the current point $X^{(j)}$. The candidate point is accepted (i.e., $X^{(j+1)} = Y$) with probability $\alpha(X^{(j)}, Y)$, where

$$\alpha(X, Y) = \min \left( 1, \frac{\pi(Y)q(X \mid Y)}{\pi(X)q(Y \mid X)} \right).$$

(2)

Otherwise, $X^{(j+1)} = X^{(j)}$. The rate of convergence depends strongly on the relationship of the stationary distribution and the proposal distribution. See [4] for strategies on choosing good proposal distributions.

There are canonical choices for proposal distributions. The Metropolis algorithm uses symmetric proposals, where $q(Y \mid X) = q(X \mid Y)$. A special case of the Metropolis algorithm is random-walk Metropolis, where $q(Y \mid X) = q(|X - Y|)$. The independence sampler [6] has a proposal density that does not depend on the current state of the chain, $q(Y \mid X) = q(Y)$. Independence samplers work well when $q(\cdot)$ is a good approximation to $\pi(\cdot)$ but with heavier tails.

Two of the most widely used choices for proposal distributions are the single-component Metropolis-Hastings algorithm and the Gibbs sampler. Instead of updating the entire vector $X$ in one step, these algorithms update smaller groups of the components of $X$—often only one component at a time. Let $X = (X_1, \ldots, X_p)$ and let $X_{-i} = (X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_p)$. Define the
full conditional distribution of $X_i$ as the distribution of the $i$th component of $X$ conditional on all of the remaining components, $X_{-i}$. If $X \sim \pi(\cdot)$, the full conditional distribution of $X_i$ is

$$
\pi(X_i \mid X_{-i}) = \frac{\pi(X)}{\int \pi(X) dX_i}.
$$

The single-component Metropolis-Hastings algorithm generates candidate points $Y_i$ from proposal densities $q_i(Y_i, X_i^{(j)}, X_{-i}^{(j)})$, where $X^{(j)} = (X_1^{(j)}, \ldots, X_i^{(j)}, X_{i+1}^{(j-1)}, \ldots, X_p^{(j-1)})$, so that components with indices smaller than $i$ have already been updated $j$ times. The acceptance probability for $Y_i$ is

$$
\alpha(X_{-i}, X_i, Y_i) = \min \left( 1, \frac{\pi(Y_i \mid X_{-i}) q_i(Y_i \mid X_i, X_{-i})}{\pi(X_i \mid X_{-i}) q_i(Y_i \mid X_i, X_{-i})} \right).
$$

The Gibbs sampler [7] is a special case of the single-component Metropolis-Hastings algorithm where the proposal distribution is the full conditional distribution,

$$
q_i(Y_i, X_i^{(j)}, X_{-i}^{(j)}) = \pi(Y_i \mid X_{-i}).
$$

The acceptance probabilities for the Gibbs sampler are all one.
3 APPLICATION TO BAYESIAN SYSTEM RELIABILITY

For illustration, suppose that we have a simple parallel system (see eqr341) that comprises the 10 pumps described in Table 1. The system works if at least one of the pumps is working. We would like to estimate the probability that the system is working at 10,000 hours. (Remember that the times in Table 1 are in thousands of hours.)

Because we describe the number of failures as a Poisson process (see eqr055), Poisson($\lambda_i t_i$), the time to the first failure of a pump is distributed Exponential($\lambda_i$). Assume for this example that once a pump fails, it is not repaired. (Of course, in a real nuclear power plant, pumps are always repaired.) We would like to estimate the probability, $R_S(t)$, that the system is working at 10,000 hours. The probability that the system is working at 10,000 hours is $R_S(10) = 1 - \prod_{i=1}^{10} (1 - \exp(-10\lambda_i))$. (See [8] for the derivation of the expression for system reliability.)

We use a combination of the Gibbs sampler and single-component Metropolis-Hastings algorithm to estimate the failure rates for each pump. This requires the full conditional distributions for each parameter.

$$[\lambda_i \mid \lambda_{-i}, \alpha, \beta] \sim \text{Gamma}(s_i + \alpha, \beta + t_i)$$

$$[\beta \mid \alpha, \lambda_1, \ldots, \lambda_{10}] \sim \text{Gamma}(10\alpha + 0.1, 1.0 + \sum_{i=1}^{10} \lambda_i)$$

$$[\alpha \mid \beta, \lambda_1, \ldots, \lambda_{10}] \propto (\prod_{i=1}^{10} \lambda_i^{s_i + \alpha - 1})\beta^{10\alpha - 0.9}(\Gamma(\alpha))^{-10}I(\alpha \in (0, 5))$$
For the \((j+1)st\) iteration of MCMC, for \(i = 1, \ldots, 10\), we set \(\lambda_i^{(j+1)}\) equal to a random draw from a Gamma\((s_i + \alpha^{(j)}, \beta^{(j)} + t_i)\) distribution, and we set \(\beta^{(j+1)}\) equal to a random draw from a Gamma\((10\alpha^{(j)} + 0.1, 1.0 + \sum_{i=1}^{10} \lambda_i^{(j+1)})\).

To sample \(\alpha^{(j+1)}\), we set a candidate point \(Y\) equal to a random draw from a proposal distribution: for example, Normal\((\alpha^{(j)}, 0.25)I(0 < \alpha < 5)\). As a general heuristic, we choose the standard deviation of the proposal distribution so that the candidate acceptance probability is between 0.25 and 0.45 [9]. For this proposal distribution, the candidate acceptance probability is

\[
\min \left(1, \frac{\prod_{i=1}^{10} (\lambda_i^{(j+1)})^{s_i + Y - 1} (\beta^{(j+1)})^{10Y - 0.9} \Gamma(Y)^{-10}}{\prod_{i=1}^{10} (\lambda_i^{(j+1)})^{s_i + \alpha^{(j)} - 1} (\beta^{(j+1)})^{10\alpha^{(j)} - 0.9} \Gamma(\alpha^{(j)})^{-10}} \right).
\]

We could also use a MCMC package like WinBUGS [10] to perform the calculations.

Once we have a sample from the posterior distribution, we evaluate \(R_S(10)\) for each draw. The posterior mean of \(R_S(10)\) is 0.92, with a 95% credible interval of \((0.77, 0.99)\). Figure 1 is a plot of the posterior distribution of the system reliability at 10,000 hours. (For more information on creating summaries using MCMC draws, see [11].)

4 SUMMARY

Hierarchical models offer many advantages including the ability to borrow strength to estimate individual parameters and the ability to specify complex models that reflect engineering and physical realities. MCMC provides a general set of
Figure 1: The posterior distribution of $R_S(10)$, the system reliability at 10,000 hours.
algorithms that enable Bayesian inference in a variety of complex models.

5 RELATED ARTICLES

eqr055, eqr081, eqr086, eqr110, eqr341, eqr353

References


