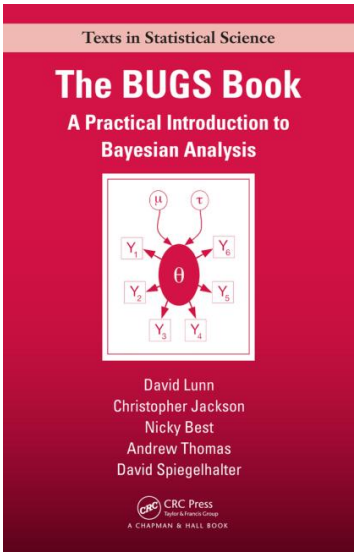
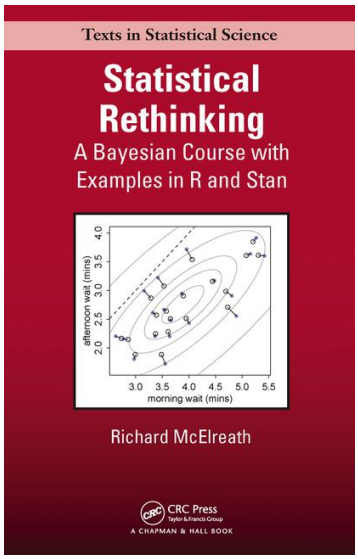
Bayesian FreeBook

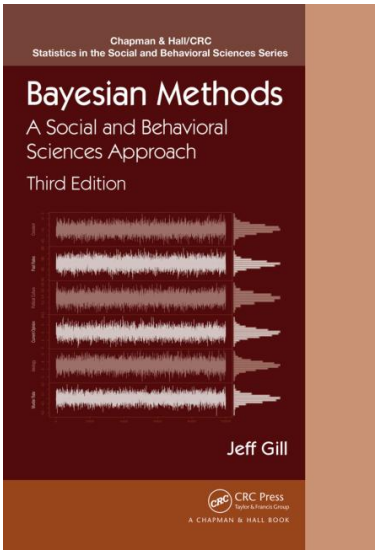
A **CRC PRESS** FREEBOOK

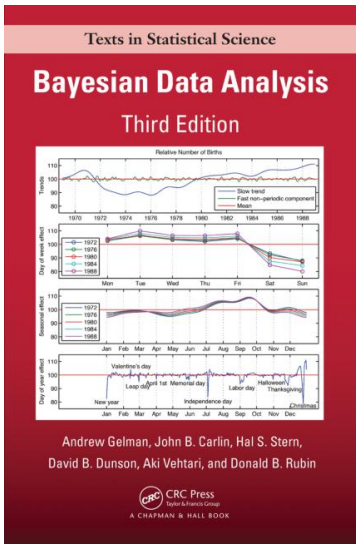
TABLE OF CONTENTS

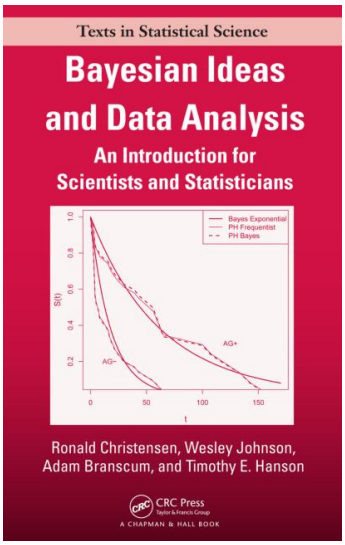
FEATURED TITLES

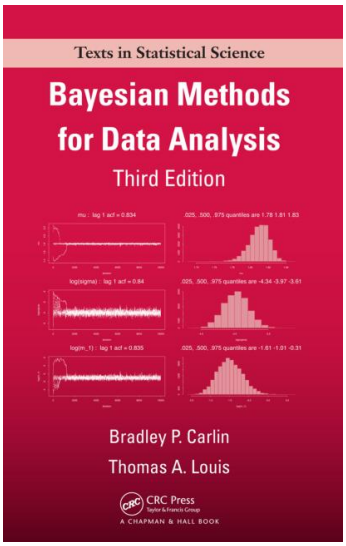












VISIT WWW.CRCPRESS.COM/STATISTICS

TO BROWSE FULL RANGE OF STATISTICS TITLES

Introduction

The Bayesian approach to statistical inference allows the user to update their model based on prior evidence and observed data. The methodology can be complex, and it is only in the last 25 years or so, with the dramatic increase in computing power that they have become more popular. They are now used across the sciences and industry, with applications to medical research and epidemiology, social science, finance, and many other areas.

Bayesian methods are now taught in every graduate program in statistics, sometimes at undergraduate level, and increasingly as a standard course in other fields. CRC Press is the leading publisher of textbooks for Bayesian methods, with books aimed specifically at statistics graduate students, students in the social sciences, and self-studying practitioners. This CRC Press Freebook presents six chapters from some of our leading textbooks in the field.

About the Titles and the Chapters

1 – Prior Distributions (Chapter 5) from The BUGS Book: A Practical Introduction to Bayesian Analysis

Authored by the team that originally developed the software, The BUGS Book provides a practical introduction to the program and its use. The text presents complete coverage of all the functionalities of BUGS, including prediction, missing data, model criticism, and prior sensitivity. It also features a large number of worked examples and a wide range of applications from various disciplines.

In this chapter we introduce basic ideas by focusing on single parameters, and in subsequent chapters consider multi-parameter situations and hierarchical models. Our emphasis is on understanding what is being used and being aware of its (possibly unintentional) influence.

2 – Markov Chain Monte Carlo (Chapter 8) from Statistical Rethinking: A Bayesian Course with Examples in R and Stan

By using complete R code examples throughout, this book provides a practical foundation for performing statistical inference. Designed for both PhD students and seasoned professionals in the natural and social sciences, it prepares them for more advanced or specialized statistical modeling.

This chapter presents an informal introduction to Markov chain Monte Carlo (MCMC) estimation. The goal is to introduce the purpose and approach MCMC algorithms. The major algorithms introduced are the Metropolis, Gibbs sampling, and Hamiltonian Monte Carlo algorithms.

3 – Specifying Bayesian Models (Chapter 3) from Bayesian Methods: A Social and Behavioral Sciences Approach, Third Edition

This bestselling, highly praised text continues to be suitable for a range of courses, including an introductory course or a computing-centered course. It shows students in the social and behavioral sciences how to use Bayesian methods in practice, preparing them for sophisticated, real-world work in the field.

This chapter covers the core idea of Bayesian statistics: updating prior distributions by conditioning on data through the likelihood function. It also looks at repeating this updating process as new information becomes available. There is an additional historical discussion placing this modeling approach into context.

4 – Hierarchical linear models (Chapter 15) from Bayesian Data Analysis, Third Edition.

Winner of the 2015 De Groot Prize from the International Society for Bayesian Analysis. Now in its third edition, this classic book is widely considered the leading text on Bayesian methods, lauded for its accessible, practical approach to analyzing data and solving research problems. Throughout the text, numerous worked examples drawn from real applications and research emphasize the use of Bayesian inference in practice.

Hierarchical or multilevel regression modeling is an increasingly important tool in the analysis of complex data such as arise frequently in modern quantitative research. Hierarchical regression models are useful as soon as there is covariate information at different levels of variation.

5 - Nonparametric Models (Chapter 15) from Bayesian Ideas and Data Analysis: An Introduction for Scientists and Statisticians

Emphasizing the use of WinBUGS and R to analyze real data, this book presents statistical tools to address scientific questions. It highlights foundational issues in

statistics, the importance of making accurate predictions, and the need for scientists and statisticians to collaborate in analyzing data. The WinBUGS code provided offers a convenient platform to model and analyze a wide range of data.

Nonparametric models are anything but nonparametric. These models involve many parameters. In this chapter parameters are added to a basic model to increase the possible shapes for either the density function or the regression function. In Section 1 we discuss more general ways to model distributions. In Section 2 we examine more general ways for defining regression functions. In Section 3 we briefly discuss the application of Section 2 to estimating a baseline hazard for the Proportional Hazards model.

6 - Model criticism and selection (Chapter 4) from Bayesian Methods for Data Analysis, Third Edition

Broadening its scope to nonstatisticians, this new edition provides an accessible introduction to the foundations and applications of Bayesian analysis. Along with a complete reorganization of the material, this edition concentrates more on hierarchical Bayesian modeling as implemented via Markov chain Monte Carlo (MCMC) methods and related data analytic techniques. Focusing on applications from biostatistics, epidemiology, and medicine, this text builds on the popularity of its predecessors by making it suitable for even more practitioners and students.

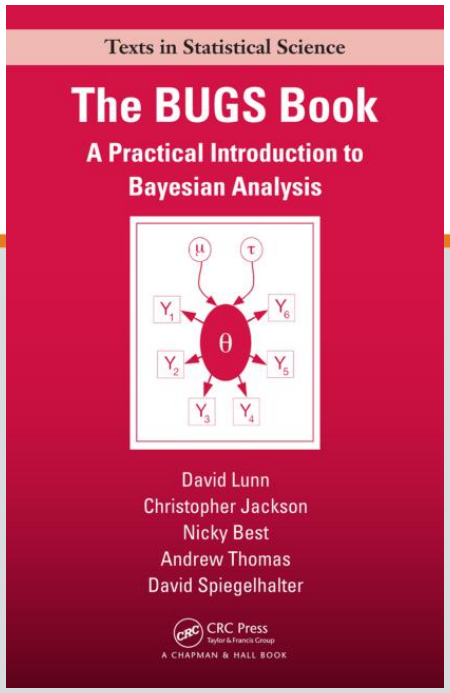
In this chapter, we will attempt to present the model selection and criticism tools most useful for the applied Bayesian, along with sufficient exemplification so that the reader may employ the approaches independently.

Note to readers: References from the original chapters have not been included in this text. For a fully-referenced version of each chapter, including footnotes, bibliographies, references and endnotes, please see the published title. As you read through this FreeBook, you will notice that some excerpts reference subsequent chapters. Please note that these are references to the original text and not the FreeBook.

CHAPTER

1

PRIOR DISTRIBUTIONS



This chapter is excerpted from

The BUGS Book: A Practical Introduction to Bayesian Analysis

by David Lunn, Chris Jackson, Nicky Best, Andrew Thomas, David Spiegelhalter.

Learn more

5

Prior distributions

The prior distribution plays a defining role in Bayesian analysis. In view of the controversy surrounding its use it may be tempting to treat it almost as an embarrassment and to emphasise its lack of importance in particular applications, but we feel it is a vital ingredient and needs to be squarely addressed. In this chapter we introduce basic ideas by focusing on single parameters, and in subsequent chapters consider multi-parameter situations and hierarchical models. Our emphasis is on understanding what is being used and being aware of its (possibly unintentional) influence.

5.1 Different purposes of priors

A basic division can be made between so-called "non-informative" (also known as "reference" or "objective") and "informative" priors. The former are intended for use in situations where scientific objectivity is at a premium, for example, when presenting results to a regulator or in a scientific journal, and essentially means the Bayesian apparatus is being used as a convenient way of dealing with complex multi-dimensional models. The term "non-informative" is misleading, since all priors contain some information, so such priors are generally better referred to as "vague" or "diffuse." In contrast, the use of informative prior distributions explicitly acknowledges that the analysis is based on more than the immediate data in hand whose relevance to the parameters of interest is modelled through the likelihood, and also includes a considered judgement concerning plausible values of the parameters based on external information.

In fact the division between these two options is not so clear-cut — in particular, we would claim that any "objective" Bayesian analysis is a lot more "subjective" than it may wish to appear. First, any statistical model (Bayesian or otherwise) requires qualitative judgement in selecting its structure and distributional assumptions, regardless of whether informative prior distributions are adopted. Second, except in rather simple situations there may not be an agreed "objective" prior, and apparently innocuous assumptions can strongly influence conclusions in some circumstances.

In fact a combined strategy is often reasonable, distinguishing parameters of

primary interest from those which specify secondary structure for the model. The former will generally be location parameters, such as regression coefficients, and in many cases a vague prior that is locally uniform over the region supported by the likelihood will be reasonable. Secondary aspects of a model include, say, the variability between random effects in a hierarchical model. Often there is limited evidence in the immediate data concerning such parameters and hence there can be considerable sensitivity to the prior distribution, in which case we recommend thinking carefully about reasonable values in advance and so specifying fairly informative priors — the inclusion of such external information is unlikely to bias the main estimates arising from a study, although it may have some influence on the precision of the estimates and this needs to be carefully explored through sensitivity analysis. It is preferable to construct a prior distribution on a scale on which one has has a good interpretation of magnitude, such as standard deviation, rather than one which may be convenient for mathematical purposes but is fairly incomprehensible, such as the logarithm of the precision. The crucial aspect is not necessarily to avoid an influential prior, but to be aware of the extent of the influence.

5.2 Vague, "objective," and "reference" priors

5.2.1 Introduction

The appropriate specification of priors that contain minimal information is an old problem in Bayesian statistics: the terms "objective" and "reference" are more recent and reflect the aim of producing a baseline analysis from which one might possibly measure the impact of adopting more informative priors. Here we illustrate how to implement standard suggestions with BUGS. Using the structure of graphical models, the issue becomes one of specifying appropriate distributions on "founder" nodes (those with no parents) in the graph.

We shall see that some of the classic proposals lead to "improper" priors that do not form distributions that integrate to 1: for example, a uniform distribution over the whole real line, no matter how small the ordinate, will still have an infinite integral. In many circumstances this is not a problem, as an improper prior can still lead to a proper posterior distribution. BUGS in general requires that a full probability model is defined and hence forces all prior distributions to be proper — the only exception to this is the `dflat()` distribution (Appendix C.1). However, many of the prior distributions used are "only just proper" and so caution is still required to ensure the prior is not having unintended influence.

5.2.2 Discrete uniform distributions

For discrete parameters it is natural to adopt a discrete uniform prior distribution as a reference assumption. We have already seen this applied to the degrees of freedom of a t-distribution in Example 4.1.2, and in §9.8 we will see how it can be used to perform a non-Bayesian bootstrap analysis within BUGS.

5.2.3 Continuous uniform distributions and Jeffreys prior

When it comes to continuous parameters, it is tempting to automatically adopt a uniform distribution on a suitable range. However, caution is required since a uniform distribution for θ does not generally imply a uniform distribution for functions of θ. For example, suppose a coin is known to be biased, but you claim to have "no idea" about the chance θ of it coming down heads and so you give θ a uniform distribution between 0 and 1. But what about the chance (θ^2) of it coming down heads in both of the next two throws? You have "no idea" about that either, but according to your initial uniform distribution on θ, $\psi = \theta^2$ has a density $p(\psi) = 1/(2\sqrt{\psi})$, which can be recognised to be a Beta(0.5, 1) distribution and is certainly not uniform.

Harold Jeffreys came up with a proposal for prior distributions which would be invariant to such transformations, in the sense that a "Jeffreys" prior for θ would be formally compatible with a Jeffreys prior for any 1–1 transformation $\psi = f(\theta)$. He proposed defining a "minimally informative" prior for θ as $p_J(\theta) \propto I(\theta)^{1/2}$ where $I(\theta) = -E[\frac{d^2}{d\theta^2} \log p(Y|\theta)]$ is the Fisher information for θ (§3.6.1). Since we can also express $I(\theta)$ as

$$I(\theta) = E_{Y|\theta}\left[\left(\frac{d \log p(Y|\theta)}{d\theta}\right)^2\right],$$

we have

$$I(\psi) = I(\theta)\left|\frac{d\theta}{d\psi}\right|^2.$$

Jeffreys' prior is therefore invariant to reparameterisation since

$$I(\psi)^{1/2} = I(\theta)^{1/2}\left|\frac{d\theta}{d\psi}\right|,$$

and the Jacobian terms cancel when transforming variables via the expression in §2.4. Hence, a Jeffreys prior for θ transforms to a Jeffreys prior for any 1–1 function $\psi(\theta)$.

As an informal justification, Fisher information measures the curvature of the log-likelihood, and high curvature occurs wherever small changes in parameter values are associated with large changes in the likelihood: Jeffreys' prior gives more weight to these parameter values and so ensures that the

influence of the data and the prior essentially coincide. We shall see examples of Jeffreys priors in future sections.

Finally, we emphasise that if the specific form of vague prior is influential in the analysis, this strongly suggests you have insufficient data to draw a robust conclusion based on the data alone and that you should not be trying to be "non-informative" in the first place.

5.2.4 Location parameters

A location parameter θ is defined as a parameter for which $p(y|\theta)$ is a function of $y - \theta$, and so the distribution of $y - \theta$ is independent of θ. In this case Fisher's information is constant, and so the Jeffreys procedure leads to a uniform prior which will extend over the whole real line and hence be improper. In BUGS we could use `dflat()` to represent this distribution, but tend to use proper distributions with a large variance, such as `dunif(-100,100)` or `dnorm(0,0.0001)`: we recommend the former with appropriately chosen limits, since explicit introduction of these limits reminds us to be wary of their potential influence. We shall see many examples of this use, for example, for regression coefficients, and it is always useful to check that the posterior distribution is well away from the prior limits.

5.2.5 Proportions

The appropriate prior distribution for the parameter θ of a Bernoulli or binomial distribution is one of the oldest problems in statistics, and here we illustrate a number of options. First, both Bayes (1763) and Laplace (1774) suggest using a uniform prior, which is equivalent to Beta$(1, 1)$. A major attraction of this assumption, also known as the Principle of Insufficient Reason, is that it leads to a discrete uniform distribution for the predicted number y of successes in n future trials, so that $p(y) = 1/(n+1)$, $y = 0, 1, ..., n$,* which seems rather a reasonable consequence of "not knowing" the chance of success. On the $\phi = \text{logit}(\theta)$ scale, this corresponds to a standard logistic distribution, represented as `dlogis(0,1)` in BUGS (see code below).

Second, an (improper) uniform prior on ϕ is formally equivalent to the (improper) Beta$(0, 0)$ distribution on the θ scale, i.e., $p(\theta) \propto \theta^{-1}(1-\theta)^{-1}$: the code below illustrates the effect of bounding the range for ϕ and hence making these distributions proper. Third, the Jeffreys principle leads to a Beta$(0.5, 0.5)$ distribution, so that $p_J(\theta) = \pi^{-1}\theta^{\frac{1}{2}}(1-\theta)^{\frac{1}{2}}$. Since it is common to use normal prior distributions when working on a logit scale, it is of interest to consider what normal distributions on ϕ lead to a "near-uniform"

*See Table 3.1 — the posterior predictive distribution for a binomial observation and beta prior is a beta-binomial distribution. With no observed data, $n = y = 0$ in Table 3.1, this posterior predictive distribution becomes the *prior predictive* distribution, which reduces to the discrete uniform for $a = b = 1$.

distribution on θ. Here we consider two possibilities: assuming a prior variance of 2 for ϕ can be shown to give a density for θ that is "flat" at $\theta = 0.5$, while a normal with variance 2.71 gives a close approximation to a standard logistic distribution, as we saw in Example 4.1.1.

```
theta[1]          ~ dunif(0,1)     # uniform on theta
phi[1]            ~ dlogis(0,1)

phi[2]            ~ dunif(-5,5)    # uniform on logit(theta)
logit(theta[2]) <- phi[2]

theta[3]          ~ dbeta(0.5,0.5) # Jeffreys on theta
phi[3]           <- logit(theta[3])

phi[4]            ~ dnorm(0,0.5)   # var=2, flat at theta = 0.5
logit(theta[4]) <- phi[4]

phi[5]            ~ dnorm(0,0.368) # var=2.71, approx. logistic
logit(theta[5]) <- phi[5]
```

We see from Figure 5.1 that the first three options produce apparently very different distributions for θ, although in fact they differ at most by a single implicit success and failure (§5.3.1). The normal prior on the logit scale with variance 2 seems to penalise extreme values of θ, while that with variance 2.71 seems somewhat more reasonable. We conclude that, in situations with very limited information, priors on the logit scale could reasonably be restricted to have variance of around 2.7.

Example 5.2.1. *Surgery (continued): prior sensitivity*
What is the sensitivity to the above prior distributions for the mortality rate in our "Surgery" example (Example 3.3.2)? Suppose in one case we observe 0/10 deaths (Figure 5.2, left panel) and in another, 10/100 deaths (Figure 5.2, right panel). For 0/10 deaths, priors 2 and 3 pull the estimate towards 0, but the sensitivity is much reduced with the greater number of observations.

5.2.6 Counts and rates

For a Poisson distribution with mean θ, the Fisher information is $I(\theta) = 1/\theta$ and so the Jeffreys prior is the improper $p_J(\theta) \propto \theta^{-\frac{1}{2}}$, which can be approximated in BUGS by a `dgamma(0.5, 0.00001)` distribution. The same prior is appropriate if θ is a rate parameter per unit time, so that $Y \sim \text{Poisson}(\theta t)$.

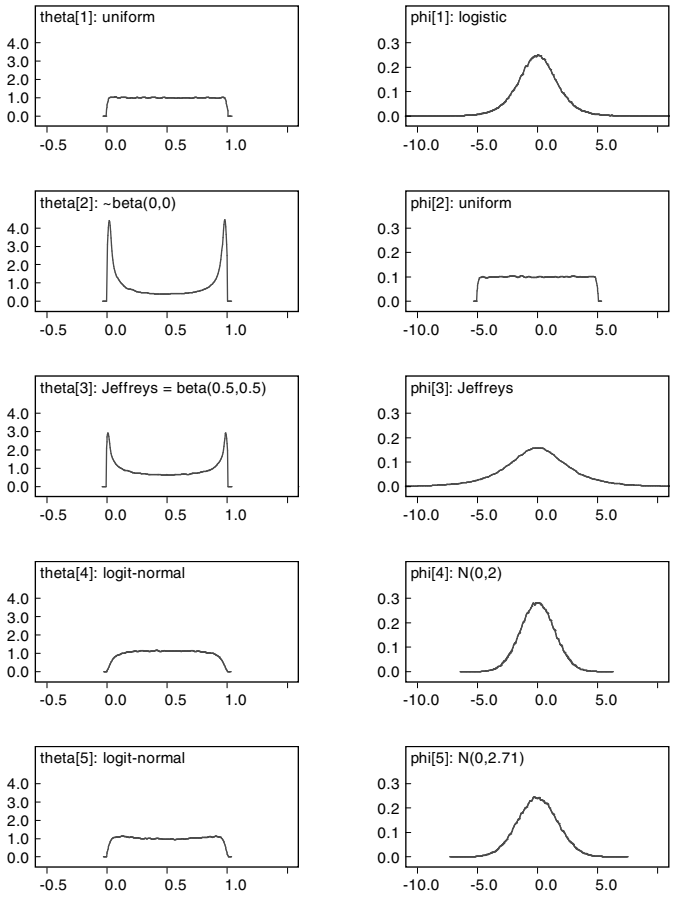


FIGURE 5.1
Empirical distributions (based on 100,000 samples) corresponding to various different priors for a proportion parameter.

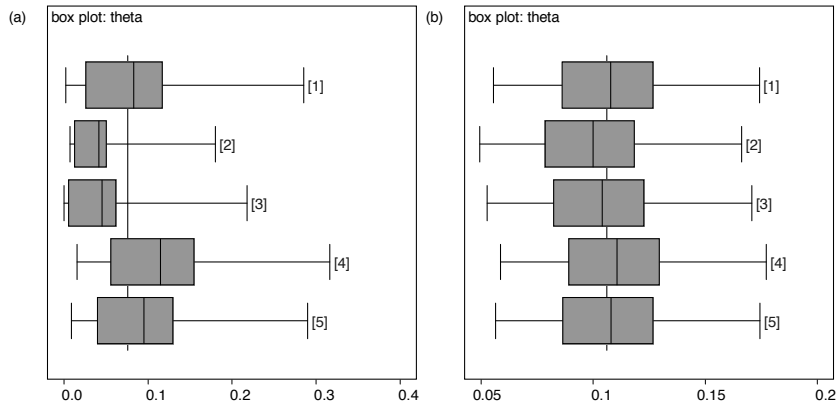


FIGURE 5.2
Box plots comparing posterior distributions arising from the five priors discussed above for mortality rate: (a) 0/10 deaths observed; (b) 10/100 deaths observed.

5.2.7 Scale parameters

Suppose σ is a scale parameter, in the sense that $p(y|\sigma) = \sigma^{-1} f(y/\sigma)$ for some function f, so that the distribution of Y/σ does not depend on σ. Then it can be shown that the Jeffreys prior is $p_J(\sigma) \propto \sigma^{-1}$, which in turn means that $p_J(\sigma^k) \propto \sigma^{-k}$, for any choice of power k. Thus for the normal distribution, parameterised in BUGS in terms of the precision $\tau = 1/\sigma^2$, we would have $p_J(\tau) \propto \tau^{-1}$. This prior could be approximated in BUGS by, say, a `dgamma(0.001,0.001)`, which also can be considered an "inverse-gamma distribution" on the variance σ^2. Alternatively, we note that the Jeffreys prior is equivalent to $p_J(\log \sigma^k) \propto$ const, i.e., an improper uniform prior. Hence it may be preferable to give $\log \sigma^k$ a uniform prior on a suitable range, for example, `log.tau ~ dunif(-10, 10)` for the logarithm of a normal precision. We would usually want the bounds for the uniform distribution to have negligible influence on the conclusions.

We note some potential conflict in our advice on priors for scale parameters: a uniform prior on $\log \sigma$ follows Jeffreys' rule but a uniform on σ is placing a prior on an interpretable scale. There usually would be negligible difference between the two — if there is a noticeable difference, then there is clearly little information in the likelihood about σ and we would recommend a weakly informative prior on the σ scale.

Note that the advice here applies only to scale parameters governing the variance or precision of *observable* quantities. The choice of prior for the variance of *random effects* in a hierarchical model is more problematic — we discuss this in §10.2.3.

5.2.8 Distributions on the positive integers

Jeffreys (1939) [p. 238] suggested that a suitable prior for a parameter N, where $N = 0, 1, 2, ...$, is $p(N) \propto 1/N$, analogously to a scale parameter.

Example 5.2.2. *Coin tossing: estimating number of tosses*
Suppose we are told that a fair coin has come up heads $y = 10$ times. How many times has the coin been tossed? Denoting this unknown quantity by N we can write down the likelihood as

$$p(y|N) = \text{Binomial}(0.5, N) \propto \frac{N!}{(N-y)!}0.5^N.$$

As N is integer-valued we must specify a *discrete* prior distribution.

Suppose we take Jeffreys' suggestion and assign a prior $p(N) \propto 1/N$, which is improper but could be curtailed at a very high value. Then the posterior distribution is

$$p(N|y) \propto \frac{N!}{(N-y)!}0.5^N/N \propto \frac{(N-1)!}{(N-y)!}0.5^N, \;\; N \geq y,$$

which we can recognise as the kernel of a negative binomial distribution with mean $2y = 20$. This has an intuitive attraction, since if instead we had fixed $y = 10$ in advance and flipped a coin until we had y heads, then the sampling distribution for the random quantity N would be just this negative binomial. However, it is notable that we were *not* told that this was the design — we have no idea whether the final flip was a head or not.

Alternatively, we may wish to assign a uniform prior over integer values from 1 to 100, i.e., $\Pr(N = n) = 1/100$, $n = 1, ..., 100$. Then the posterior for N is proportional to the likelihood, and its expectation, for example, is given by

$$E[N|y] = \sum_{n=1}^{100} n \Pr(N = n|y) = A \sum_{n=1}^{100} \frac{n \times n!}{(n-y)!}0.5^n, \tag{5.1}$$

where A is the posterior normalising constant. The right-hand side of (5.1) cannot be simplified analytically and so is cumbersome to evaluate (although this is quite straightforward with a little programming). In BUGS we simply specify the likelihood and the prior as shown below.

```
y <- 10
y  ~ dbin(0.5, N)
N  ~ dcat(p[])
for (i in 1:100) {p[i] <- 1/100}
```

BUGS can use the resulting samples to summarise the posterior graphically as well as numerically. Numeric summaries, such as the one shown below, allow us to make formal inferences; for example, we can be 95% certain that the coin has been tossed between 13 and 32 times. Graphical summaries, on the other hand,

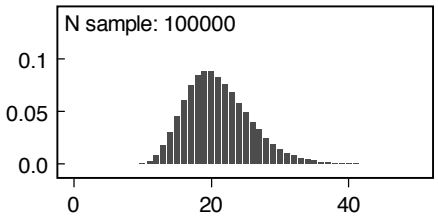


FIGURE 5.3
Approximate posterior distribution for number of (unbiased) coin tosses leading to ten heads.

might reveal interesting features of the posterior. Figure 5.3 shows the posterior density for N. Note that the mode is 20, which is the intuitive answer, as well as being the MLE and the posterior mean using the Jeffreys prior. Note also that although the uniform prior supports values in $\{1, ..., 9\}$, which are impossible in light of the observed data (10 heads), the posterior probability for these values is, appropriately, zero.

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
N	21.01	4.702	0.01445	13.0	20.0	32.0	1	100000

In Example 5.5.2 we consider a further example of a prior over the positive integers which reveals the care that can be required.

5.2.9 More complex situations

Jeffreys' principle does not extend easily to multi-parameter situations, and additional context-specific considerations generally need to be applied, such as assuming prior independence between location and scale parameters and using the Jeffreys prior for each, or specifying an ordering of parameters into groups of decreasing interest.

5.3 Representation of informative priors

Informative prior distributions can be based on pure judgement, a mixture of data and judgement, or data alone. Of course, even the selection of relevant data involves a substantial degree of judgement, and so the specification of an informative prior distribution is never an automatic procedure.

We summarise some basic techniques below, emphasising the mapping of relevant data and judgement onto appropriate parametric forms, ideally representing "implicit" data.

5.3.1 Elicitation of pure judgement

Elicitation of subjective probability distributions is not a straightforward task due to a number of potential biases that have been identified. O'Hagan et al. (2006) provide some "*Guidance for best practice*," emphasising that probability assessments are constructed by the questioning technique, rather than being "pre-formed quantifications of pre-analysed belief" (p. 217). They say it is best to interview subjects face-to-face, with feedback and continual checking for biases, conducting sensitivity analysis to the consequence of the analysis, and avoiding verbal descriptions of uncertainty. They recommend eliciting intervals with moderate rather than high probability content, say by focusing on 33% and 67% quantiles: indeed one can simply ask for an interval and afterwards elicit a 'confidence' in that assessment (Kynn, 2005). They suggest using multiple experts and reporting a simple average, but it is also important to acknowledge imperfections in the process, and that even genuine "expertise" cannot guarantee a suitable subject. See also Kadane and Wolfson (1998) for elicitation techniques for specific models.

In principle any parametric distribution can be elicited and used in BUGS. However, it can be advantageous to use conjugate forms since, as we have seen in Chapter 3, the prior distribution can then be interpreted as representing "implicit data," in the sense of a prior estimate of the parameter and an "effective prior sample size." It might even then be possible to include the prior information as "data" and use standard classical methods (and software) for statistical analysis.

Below we provide a brief summary of situations: in each case the "implicit data" might be directly elicited, or measures of central tendency and spread requested and an appropriate distribution fitted. A simple moment-based method is to ask directly for the mean and standard deviation, or elicit an approximate 67% interval (i.e., the parameter is assessed to be twice as likely to be inside the interval as outside it) and then treat the interval as representing the mean $\pm$ 1 standard deviation, and solve for the parameters of the prior distribution. In any case it is good practice to iterate between alternative representations of the prior distribution, say as a drawn distribution, percentiles, moments, and interpretation as "implicit data," in order to check the subject is happy with the implications of their assessments.

- Binomial proportion θ. Suppose our prior information is equivalent to having observed y events in a sample size of n, and we wanted to derive a corresponding Beta(a, b) prior for θ. Combining an improper Beta(0,0) "pre-prior" with these implicit data gives a conjugate "posterior" of Beta$(y, n - y)$, which we can interpret as our elicited prior. The mean

of this elicited prior is $a/(a+b) = y/n$, the intuitive point estimate for θ, and the implicit sample size is $a+b = n$. Using a uniform "pre-prior" instead of the Beta(0,0) gives $a = y+1$ and $b = n-y+1$.

Alternatively, a moment-based method might proceed by eliciting a prior standard deviation as opposed to a prior sample size, and by then solving the mean and variance formulae (Appendix C.3) for a and b: $a = mb/(1-m)$, $b = m(1-m)^2/v + m - 1$, for an elicited mean $m = \hat{\theta}$ and variance v.

- Poisson rate θ: if we assume θ has a Gamma(a, b) distribution we can again elicit a prior estimate $\hat{\theta} = a/b$ and an effective sample size of b, assuming a Gamma(0,0) pre-prior (see Table 3.1, Poisson-gamma conjugacy), or we can use a moment-based method instead.

- Normal mean μ: a normal distribution can be obtained be eliciting a mean γ and standard deviation ω directly or via an interval. By conditioning on a sampling variance σ^2, we can calculate an effective prior sample size $n_0 = \sigma^2/\omega^2$ which can be fed back to the subject.

- Normal variance σ^2: $\tau = \sigma^{-2}$ may be assumed to have a Gamma(a, b) distribution, where a/b is set to an estimate of the precision, and $2a$ is the effective number of prior observations, assuming a Gamma(0,0) pre-prior (see Table 3.1, normal y with unknown variance σ^2).

- Regression coefficients: In many circumstances regression coefficients will be unconstrained parameters in standard generalised linear models, say log-odds ratios in logistic regression, log-rate-ratios in Poisson regression, log-hazard ratios in Cox regression, or ordinary coefficients in standard linear models. In each case it is generally appropriate to assume a normal distribution. Kynn (2005) described the elicitation of regression coefficients in GLMs by asking an expert for expected responses for different values of a predictor. Lower and upper estimates, with an associated degree of confidence, were also elicited, and the answers used to derive piecewise-linear priors.

Example 5.3.1. *Power calculations*
A randomised trial is planned with n patients in each of two arms. The response within each treatment arm is assumed to have between-patient standard deviation σ, and the estimated treatment effect Y is assumed to have a Normal$(\theta, 2\sigma^2/n)$ distribution. A trial designed to have two-sided Type I error α and Type II error β in detecting a true difference of θ in mean response between the groups will require a sample size per group of

$$n = \frac{2\sigma^2}{\theta^2}(z_{1-\beta} + z_{1-\alpha/2})^2,$$

where $\Pr(Z < z_p) = p$ for a standard normal variable $Z \sim \text{Normal}(0, 1)$. Alternatively, for fixed n, the power of the study is

$$\text{Power} = \Phi\left(\sqrt{\frac{n\theta^2}{2\sigma^2}} - z_{1-\alpha/2}\right).$$

If we assume $\theta = 5, \sigma = 10$, $\alpha = 0.05, \beta = 0.10$, so that the power of the trial is 90%, then we obtain $z_{1-\beta} = 1.28$, $z_{1-\alpha/2} = 1.96$, and $n = 84$.

Suppose we wish to acknowledge uncertainty about the alternative hypothesis θ and the standard deviation σ. First, we assume past evidence suggests θ is likely to lie anywhere between 3 and 7, which we choose to interpret as a 67% interval ($\pm$ 1 standard deviation), and so $\theta \sim \text{Normal}(5, 2^2)$. Second, we assess our estimate of $\sigma = 10$ as being based on around 40 observations, from which we assume a Gamma(a, b) prior distribution for $\tau = 1/\sigma^2$ with mean $a/b = 1/10^2$ and effective sample size $2a = 40$, from which we derive $\tau \sim \text{Gamma}(20, 2000)$.

```
tau    ~ dgamma(20, 2000)
sigma <- 1/sqrt(tau)
theta  ~ dnorm(5, 0.25)
n     <- 2*pow((1.28 + 1.96)*sigma/theta, 2)  # n for 90% power
power <- phi(sqrt(84/2)*theta/sigma - 1.96)   # power for n = 84
p70   <- step(power - 0.7)                    # Pr(power > 70%)
```

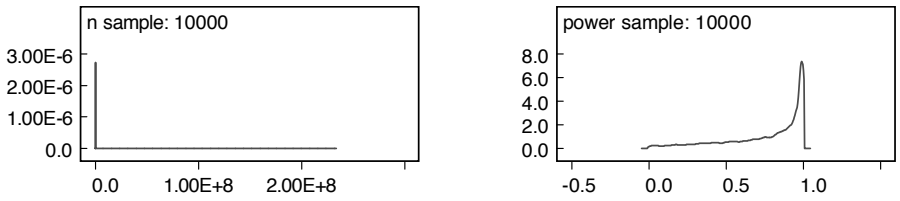


FIGURE 5.4
Empirical distributions based on 10,000 simulations for: `n`, the number of subjects required in each group to achieve 90% power, and `power`, the power achieved with 84 subjects in each group.

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
n	38740.0	2.533E+6	25170.0	24.73	87.93	1487.0	1	10000
p70	0.7012	0.4577	0.004538	0.0	1.0	1.0	1	10000
power	0.7739	0.2605	0.002506	0.1151	0.8863	1.0	1	10000

Note that the median values for `n` (88) and `power` (0.89) are close to the values derived by assuming fixed θ and σ (84 and 0.90, respectively), but also note the

huge uncertainty. It is quite plausible, under the considered prior for θ and σ, that to achieve 90% power the trial may need to include nearly 3000 subjects. Then again, we might get away with as few as 50! A trial involving 84 subjects in each group could be seriously underpowered, with 12% power being quite plausible. Indeed, there is a 30% chance that the power will be less than 70%.

5.3.2 Discounting previous data

Suppose we have available some historical data and we could obtain a prior distribution for the parameter θ based on an empirical estimate $\hat{\theta}_H$, say, by matching the prior mean and standard deviation to $\hat{\theta}_H$ and its estimated standard error. If we were to use this prior directly then we would essentially be pooling the data in a form of meta-analysis (see §11.4), in which case it would be preferable (and essentially equivalent) to use a reference prior and include the historical data directly in the model.

If we are reluctant to do this, it must be because we do not want to give the historical data full weight, perhaps because we do not consider it to have the same relevance and rigour as the data directly being analysed. We may therefore wish to *discount* the historical data using one of the methods outlined below.

- *Power prior:* this uses a prior mean based on the historical estimate $\hat{\theta}_H$, but discounts the "effective prior sample size" by a factor κ between 0 and 1: for example, a fitted Beta(a, b) would become a Beta$(\kappa a, \kappa b)$, a Gamma(a, b) would become a Gamma$(\kappa a, \kappa b)$, a Normal(γ, ω^2) would become a Normal$(\gamma, \omega^2/\kappa)$ (Ibrahim and Chen, 2000).

- *Bias modelling:* This explicitly considers that the historical data may be biased, in the sense that the estimate $\hat{\theta}_H$ is estimating a slightly different quantity from the θ of current interest. We assume that $\theta = \theta_H + \delta$, where δ is the bias whose distribution needs to be assessed. We further assume $\delta \sim [\mu_\delta, \sigma_\delta^2]$, where $[,]$ indicates a mean and variance but otherwise unspecified distribution. Then if we assume the historical data give rise to a prior distribution $\theta_H \sim [\gamma_H, \omega_H^2]$, we obtain a prior distribution for θ of
$$\theta \sim [\gamma_H + \mu_\delta, \omega_H^2 + \sigma_\delta^2].$$
Thus the prior mean is shifted and the prior variance is increased.

The power prior only deals with variability — the discount factor κ essentially represents the "weight" on a historical observation, which is an attractive concept to communicate but somewhat arbitrary to assess. In contrast, the bias modelling approach allows biases to be added, and the parameters can be defined in terms of the size of potential biases.

Example 5.3.2. *Power calculations (continued)*
We consider the power example (Example 5.3.1) but with both prior distributions discounted. We assume each historical observation informing the prior distribution for σ is only worth half a current observation, so that the prior for σ is only based on 10 rather than 20 observations. This discounts the parameters in the gamma distribution for τ by a factor of 2. For the treatment effect, we assume that the historical experiment could have been more favourable than the current one, so that the historical treatment effect had a bias with mean -1 and SD 2, and so would be expected to be between -5 and 3. Thus an appropriate prior distribution is $\theta \sim \text{Normal}(5-1, 2^2+2^2)$ or $\text{Normal}(4, 8)$ — this has been constrained to be > 0 using the `I(,)` construct (see Appendix A.2.2 and §9.6). This leads to the code:

```
# tau    ~ dgamma(20, 2000)
  tau    ~ dgamma(10, 1000)      # discounted by 2
# theta  ~ dnorm(5, 0.25)
  theta  ~ dnorm(4, 0.125)I(0,)  # 4 added to var and shifted
                                 # by -1, constrained to be >0
```

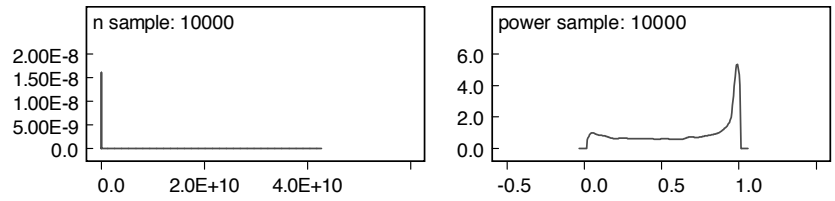


FIGURE 5.5
Empirical distributions based on 10,000 simulations for: `n`, the number of subjects required in each group to achieve 90% power, and `power`, the power achieved with 84 subjects in each group. Discounted priors for `tau` and `theta` used.

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
n	4.542E+6	4.263E+8	4.26E+6	20.96	125.6	14270.0	1	10000
p70	0.5398	0.4984	0.005085	0.0	1.0	1.0	1	10000
power	0.6536	0.3315	0.003406	0.04353	0.7549	1.0	1	10000

This has raised the median sample size to 126, but with huge uncertainty. There is a 46% probability that the power is less than 70% if the sample size stays at 84.

5.4 Mixture of prior distributions

Suppose we want to express doubt about which of two or more prior distributions is appropriate for the data in hand. For example, we might suspect that *either* a drug will produce a similar effect to other related compounds, *or* if it doesn't behave like these compounds we are unsure about its likely effect.

For two possible prior distributions $p_1(\theta)$ and $p_2(\theta)$ for a parameter θ, the overall prior distribution is then a *mixture*

$$p(\theta) = qp_1(\theta) + (1-q)p_2(\theta),$$

where q is the assessed probability that p_1 is "correct." If we now observe data y, it turns out that the posterior for θ is

$$p(\theta|y) = q'p_1(\theta|y) + (1-q')p_2(\theta|y)$$

where

$$p_i(\theta|y) \propto p(y|\theta)p_i(\theta),$$
$$q' = \frac{qp_1(y)}{qp_1(y) + (1-q)p_2(y)},$$

where $p_i(y) = \int p(y|\theta)p_i(\theta)\,d\theta$ is the predictive probability of the data y assuming $p_i(\theta)$. The posterior is a mixture of the respective posterior distributions under each prior assumption, with the mixture weights adapted to support the prior that provides the best prediction for the observed data.

This structure is easy to implement in BUGS for any form of prior assumptions. We first illustrate its use with a simple example and then deal with some of the potential complexities of this formulation. In the example, `pick` is a variable taking the value j when the prior assumption j is selected in the simulation.

Example 5.4.1. *A biased coin?*
Suppose a coin is either unbiased or biased, in which case the chance of a "head" is unknown and is given a uniform prior distribution. We assess a prior probability of 0.9 that it is unbiased, and then observe 15 heads out of 20 tosses — what is the chance that the coin is biased?

```
r <- 15; n <- 20          # data
#####################################
r         ~ dbin(p, n)    # likelihood
p        <- theta[pick]
pick      ~ dcat(q[])     # 2 if biased, 1 otherwise
q[1]     <- 0.9
```

```
q[2]     <- 0.1
theta[1] <- 0.5           # if unbiased
theta[2]  ~ dunif(0, 1)   # if biased
biased   <- pick - 1      # 1 if biased, 0 otherwise
```

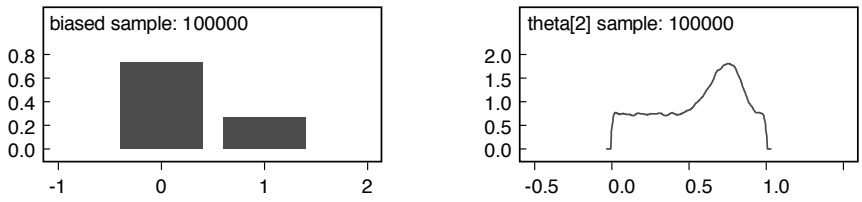


FIGURE 5.6
Biased coin: empirical distributions based on 100,000 simulations.

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
biased	0.2619	0.4397	0.002027	0.0	0.0	1.0	1	100000
theta[2]	0.5594	0.272	9.727E-4	0.03284	0.6247	0.9664	1	100000

So the probability that the coin is biased has increased from 0.1 to 0.26 on the basis of the evidence provided. The rather strange shape of the posterior distribution for `theta[2]` is explained below.

If the alternative prior assumptions for `theta` in Example 5.4.1 were from the same parametric family, e.g., beta, then we could formulate this as `p` $\sim$ `dbeta(a[pick], b[pick])`, say, with specified values of `a[1]`, `a[2]`, `b[1]`, and `b[2]`. However, the more general formulation shown in the example allows prior assumptions of arbitrary structure.

It is important to note that when `pick=1`, `theta[1]` is sampled from its *posterior* distribution, but `theta[2]` is sampled from its *prior* as `pick=1` has essentially "cut" the connection between the data and `theta[2]`. At another MCMC iteration, we may have `pick=2` and so the opposite will occur, and this means that the posterior for each `theta[j]` recorded by BUGS is a mixture of "true" (model specific) posterior and its prior. This explains the shape of the posterior for `theta[2]` in the example above. If we are interested in the posterior distribution under each prior assumption individually, then we could do a separate run under each prior assumption, or only use those values for `theta[j]` simulated when `pick=j`: this "post-processing" would have to be performed outside BUGS.

We are essentially dealing with alternative model formulations, and our q's above correspond to posterior probabilities of models. There are well-known difficulties with these quantities both in theory, due to their potential

dependence on the within-model prior distributions, and in particular when calculating within MCMC: see §8.7. In principle we can use the structure above to handle a list of arbitrary alternative models, but in practice considerable care is needed if the sampler is not to go "off course" when sampling from the prior distribution at each iteration when that model is not being "picked." It is possible to define "pseudo-priors" for these circumstances, where `pick` also dictates the prior to be assumed for `theta[j]` when `pick` $\neq j$ — see §8.7 and Carlin and Chib (1995).

5.5 Sensitivity analysis

Given that there is no such thing as the *true* prior, sensitivity analysis to alternative prior assumptions is vital and should be an integral part of Bayesian analysis. The phrase "community of priors" (Spiegelhalter et al., 2004) has been used in the clinical trials literature to express the idea that different priors may reflect different perspectives: in particular, the concept of a "sceptical prior" has been shown to be valuable. Sceptical priors will typically be centred on a "null" value for the relevant parameter with the spread reflecting plausible but small effects. We illustrate the use of sceptical and other prior distributions in the following example, where the evidence for an efficacious intervention following myocardial infarction is considered under a range of priors for the treatment effect, namely, "vague," "sceptical," "enthusiastic," "clinical," and "just significant."

Example 5.5.1. *GREAT trial*

Pocock and Spiegelhalter (1992) examine the effect of anistreplase on recovery from myocardial infarction. 311 patients were randomised to receive either anistreplase or placebo (conventional treatment); the number of deaths in each group is given in the table below.

		Treatment		total
		anistreplase	placebo	
Event	death	13	23	36
	no death	150	125	275
total		163	148	311

Let r_j, n_j, and π_j denote the number of deaths, total number of patients, and underlying mortality rate, respectively, in group $j \in \{1, 2\}$ (1 = anistreplase; 2 = placebo). Inference is required on the log-odds ratio (log(OR)) for mortality in the anistreplase group compared to placebo, that is,

$$\delta = \log\left\{\frac{\pi_1/(1-\pi_1)}{\pi_2/(1-\pi_2)}\right\} = \text{logit}\,\pi_1 - \text{logit}\,\pi_2. \tag{5.2}$$

A classical maximum likelihood estimator and approximate variance are given by

$$\hat{\delta} = \log\left\{\frac{r_1/(n_1 - r_1)}{r_2/(n_2 - r_2)}\right\}, \quad V(\hat{\delta}) \approx s^2 = \frac{1}{r_1} + \frac{1}{r_2} + \frac{1}{n_1 - r_1} + \frac{1}{n_2 - r_2}.$$

For the above data these give $\hat{\delta} = -0.753$ with $s = 0.368$. An approximate Bayesian analysis might proceed via the assumption $\hat{\delta} \sim$ Normal(δ, s^2) with a locally uniform prior on δ, e.g., $\delta \sim$ Uniform$(-10, 10)$. A more appropriate likelihood is a binomial assumption for each observed number of deaths: $r_j \sim$ Binomial(π_j, n_j), $j = 1, 2$. In this case we could be "vague" by specifying Jeffreys priors for the mortality rates, $\pi_j \sim$ Beta$(0.5, 0.5)$, $j = 1, 2$, and then deriving the posterior for δ via (5.2). Alternatively we might parameterise the model directly in terms of δ:

$$\text{logit}\,\pi_1 = \alpha + \delta/2, \quad \text{logit}\,\pi_2 = \alpha - \delta/2,$$

which facilitates the specification of informative priors for δ. Here α is a nuisance parameter and is assigned a vague normal prior: $\alpha \sim$ Normal$(0, 100^2)$. Our first informative prior for δ is a "clinical" prior based on expert opinion: a senior cardiologist, informed by one unpublished and two published trials, expressed belief that *"an expectation of 15–20% reduction in mortality is highly plausible, while the extremes of no benefit and a 40% relative reduction are both unlikely."* This is translated into a normal prior with a 95% interval of -0.51 to 0 (0.6 to 1.0 on the OR scale): $\delta \sim$ Normal$(-0.26, 0.13^2)$. We also consider a "sceptical" prior, which is designed to represent a reasonable expression of doubt, perhaps to avoid early stopping of trials due to fortuitously positive results. For example, a hypothetical sceptic might find treatment effects more extreme than a 50% reduction or 100% increase in mortality largely implausible, giving a 95% prior interval (assuming normality) of -0.69 to 0.69 (0.5 to 2 on the OR scale): $\delta \sim$ Normal$(0, 0.35^2)$.

As a counterbalance to the sceptical prior we might specify an "enthusiastic" or "optimistic" prior, as a basis for conservatism in the face of early negative results, say. Such a prior could be centred around some appropriate beneficial treatment effect with a small prior probability (e.g., 5%) assigned to negative treatment benefits. We do not construct such a prior in this example, however, since the clinical prior described above also happens to be "enthusiastic" in this sense. Another prior of interest is the "just significant" prior. Assuming that the treatment effect is significant under a vague prior, it is instructive to ask how sceptical we would have to be for that significance to vanish. Hence we assume $\delta \sim$ Normal$(0, \sigma_\delta^2)$ and we search for the largest value of σ_δ such that the 95% posterior credible interval (just) includes zero. BUGS code for performing such a search is presented below along with code to implement the clinical, sceptical, and vague priors discussed above. (Note that a preliminary search had been run to identify the approximate value of σ_δ as somewhere between 0.8 and 1, though closed form approximations exist for this "just signficant" prior (Matthews, 2001; Spiegelhalter et al., 2004)).

```
model {
  for (i in 1:nsearch) {               # search for "just
    pr.sd[i]         <- start + i*step # significant" prior
    pr.mean[i]       <- 0
  }
  pr.mean[nsearch+1] <- -0.26
  pr.sd[nsearch+1]   <- 0.13           # clinical prior
  pr.mean[nsearch+2] <- 0
  pr.sd[nsearch+2]   <- 0.35           # sceptical prior

  # replicate data for each prior and specify likelihood...
  for (i in 1:(nsearch+3)) {
    for (j in 1:2) {
      r.rep[i,j]     <- r[j]
      n.rep[i,j]     <- n[j]
      r.rep[i,j]      ~ dbin(pi[i,j], n.rep[i,j])
    }
  }
  delta.mle          <- -0.753
  delta.mle           ~ dnorm(delta[nsearch+4], 7.40)

  # define priors and link to log-odds...
  for (i in 1:(nsearch+2)) {
    logit(pi[i,1])   <- alpha[i] + delta[i]/2
    logit(pi[i,2])   <- alpha[i] - delta[i]/2
    alpha[i]          ~ dnorm(0, 0.0001)
    delta[i]          ~ dnorm(pr.mean[i], pr.prec[i])
    pr.prec[i]       <- 1/pow(pr.sd[i], 2)
  }
  pi[nsearch+3,1]     ~ dbeta(0.5, 0.5)
  pi[nsearch+3,2]     ~ dbeta(0.5, 0.5) # Jeffreys prior
  delta[nsearch+3]   <- logit(pi[nsearch+3,1])
                      - logit(pi[nsearch+3,2])
  delta[nsearch+4]    ~ dunif(-10, 10) # locally uniform prior
}

list(r = c(13, 23), n = c(163, 148),
start = 0.8, step = 0.005, nsearch = 40)
```

The derived value of σ_δ is ~ 0.925, corresponding to the 25th element of `delta[]` above. Selected posterior and prior distributions are summarised below. We note the essentially identical conclusions of the classical maximum likelihood approach and the two analyses with vague priors. The results suggest we should conclude that anistreplase is a superior treatment to placebo if we are either (a priori) completely ignorant of possible treatment effect sizes, or we trust the senior cardiologist's expert opinion, or perhaps if we are otherwise enthusiastic about the

new treatment's efficacy. If, on the other hand, we wish to claim prior indifference as to the sign of the treatment effect but we believe "large" treatment effects to be implausible, we should be more cautious. The "just significant" prior has a 95% interval of $(\exp(-1.96 \times 0.925), \exp(1.96 \times 0.925)) = (0.16, 6.1)$ on the OR scale, corresponding to reductions/increases in mortality as extreme as 84%/610%. These seem quite extreme, implying that only a small degree of scepticism is required to render the analysis "non-significant." We might conclude that the GREAT trial alone does not provide "credible" evidence for superiority, and larger-scale trials are required to quantify the treatment effect precisely.

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
delta[25]	-0.6635	0.3423	5.075E-4	-1.343	-0.6609	3.598E-4	1001	500000
delta[41]	-0.317	0.1223	1.741E-4	-0.5562	-0.317	-0.07745	1001	500000
delta[42]	-0.3664	0.2509	3.497E-4	-0.8608	-0.366	0.1245	1001	500000
delta[43]	-0.7523	0.367	5.342E-4	-1.487	-0.7479	-0.04719	1001	500000
delta[44]	-0.7534	0.3673	5.432E-4	-1.475	-0.7529	-0.0334	1001	500000

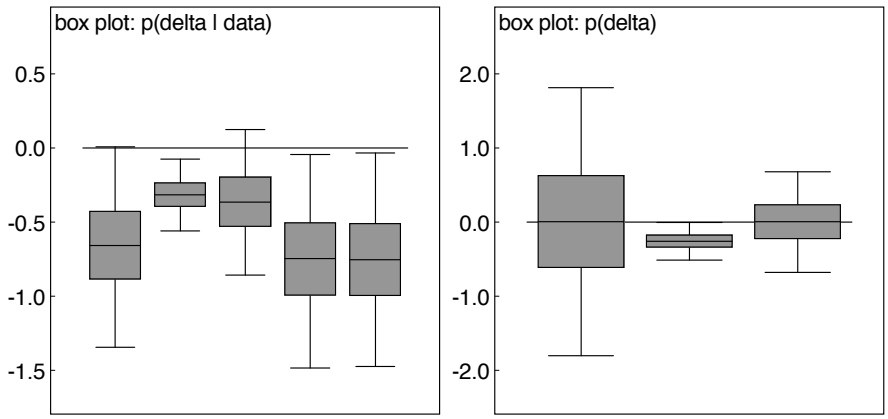


FIGURE 5.7
Left-hand side: Posterior distributions for δ from analysis of GREAT trial data. From left to right: corresponding to "just significant," "clinical," "sceptical," "Jeffreys" and "locally uniform" priors. Right-hand side: Prior distributions for analysis of GREAT trial data. From left to right: "just significant," "clinical" and "sceptical."

A primary purpose of trying a range of reasonable prior distributions is to find unintended sensitivity to apparently innocuous "non-informative" assumptions. This is reflected in the following example.

Example 5.5.2. *Trams: a classic problem from Jeffreys (1939)*
Suppose you enter a town of unknown size whose trams you know are numbered consecutively from 1 to N. You first see tram number $y = 100$. How large might N be?

We first note that the sampling distribution is uniform between 1 and N, so that $p(y|N) = \frac{1}{N}$, $y = 1, 2, \ldots, N$. Therefore the likelihood function for N is $\propto 1/N$, $N \geq y$, so that y maximises the likelihood function and so is the maximum likelihood estimator. The maximum likelihood estimate is therefore 100, which does not appear very reasonable.

Suppose we take a Bayesian approach and consider the prior distributions on the positive integers explored earlier (Example 5.2.2) — we will first examine the consequences using WinBUGS and then algebraically. We first consider a prior that is uniform on the integers up to an arbitrary upper bound M, say 5000. Y is assumed drawn from a categorical distribution: the following code shows how to set a uniform prior for N over the integers 1 to 5000 (as in Example 5.2.2) and how to use the `step` function to create a uniform sampling distribution between 1 and N.

```
Y <- 100
########################
Y            ~ dcat(p[])
# sampling distribution is uniform over first N integers
# use step function to change p[j] to 0 for j>N
for (j in 1:M) {
  p[j]      <- step(N - j + 0.01)/N
}
N            ~ dcat(p.unif[])
for (j in 1:M) {
  p.unif[j] <- 1/M
}
```

```
node mean   sd     MC error 2.5%  median 97.5%  start sample
N    1274.0 1295.0 10.86    109.0 722.0  4579.0 1001  10000
```

The posterior mean is 1274 and the median is 722, reflecting a highly skewed distribution. But is this a sensible conclusion? For an improper uniform prior over the whole of the integers, the posterior distribution is

$$p(N|y) \propto p(y|N)p(N) \propto 1/N, \ N \geq y.$$

This series diverges and so this produces an improper posterior distribution. Although our bounded prior is proper and so our posterior distribution is formally proper, this "almost improper" character is likely to lead to extreme sensitivity to prior assumptions. For example, a second run with $M = 15{,}000$ results in a

posterior mean of 3041 and median 1258. In fact we could show algebraically that the posterior mean increases as $M/\log(M)$; thus we can make it as big as we want by increasing M (proof as exercise).

We now consider Jeffreys' suggestion of a prior $p(N) \propto 1/N$, which is improper but can be constructed as follows if an upper bound, say 5000, is set.

```
N                ~ dcat(p.jeffreys[])
for (j in 1:5000) {
  reciprocal[j] <- 1/j
  p.jeffreys[j] <- reciprocal[j]/sum.recip
}
sum.recip       <- sum(reciprocal[])
```

The results show a posterior mean of 409 and median 197, which seems more reasonable — Jeffreys approximated the probability that there are more than 200 trams as $1/2$.

```
node mean  sd    MC error 2.5%  median 97.5%  start sample
N    408.7 600.4 4.99     102.0 197.0  2372.0 1001  10000
```

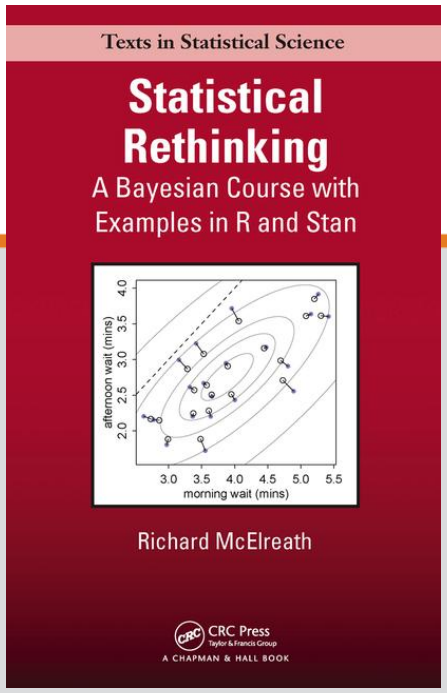
Suppose we now change the arbitrary upper bound to $M = 15{,}000$. Then the posterior mean becomes 520 and median 200. The median, but not the mean, is therefore robust to the prior. We could show that the conclusion about the median is robust to the arbitrary choice of upper bound M by proving that as M goes to infinity the posterior median tends to a fixed quantity (proof as exercise).

Finally, if a sensitivity analysis shows that the prior assumptions make a difference, then this finding should be welcomed. It means that the Bayesian approach has been worthwhile taking, and you will have to think properly about the prior and justify it. It will generally mean that, at a minimum, a weakly informative prior will need to be adopted.

CHAPTER

2

MARKOV CHAIN MONTE CARLO



This chapter is excerpted from *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*

by Richard McElreath.

Learn more

8 Markov Chain Monte Carlo

For most of Western history, chance has been a villain. In classic Roman civilization, chance was personified by Fortuna, goddess of cruel fate, with her spinning wheel of luck. Opposed to her sat Minerva, goddess of wisdom and understanding. Only the desperate would pray to Fortuna, while everyone implored Minerva for aid. Certainly science was the domain of Minerva, a realm with no useful role for Fortuna to play.

But by the beginning of the 20th century, the opposition between Fortuna and Minerva had changed to a collaboration. Scientists, servants of Minerva, began publishing books of random numbers, instruments of chance to be used for learning about the world. Now, chance and wisdom share a cooperative relationship, and few of us are any longer bewildered by the notion that an understanding of chance could help us acquire wisdom. Everything from weather forecasting to finance to evolutionary biology is dominated by the study of stochastic processes.[116]

This chapter introduces one of the more marvelous examples of how Fortuna and Minerva cooperate: the estimation of posterior probability distributions using a stochastic process known as **Markov chain Monte Carlo** (MCMC) estimation. Unlike in every earlier chapter in this book, here we'll produce samples from the joint posterior of a model without maximizing anything. Instead of having to lean on quadratic and other approximations of the shape of the posterior, now we'll be able to sample directly from the posterior without assuming a Gaussian, or any other, shape for it.

The cost of this power is that it may take much longer for our estimation to complete, and usually more work is required to specify the model as well. But the benefit is escaping the awkwardness of assuming multivariate normality. Equally important is the ability to directly estimate models, such as the generalized linear and multilevel models of later chapters. Such models routinely produce non-Gaussian posterior distributions, and sometimes they cannot be estimated at all with the techniques of earlier chapters.

The good news is that tools for building and inspecting MCMC estimates are getting better all the time. In this chapter you'll meet a convenient way to convert the `map` formulas you've used so far into Markov chains. The engine that makes this possible is **Stan** (free and online at: mc-stan.org). Stan's creators describe it as "a probabilistic programming language implementing statistical inference." You won't be working directly in Stan to begin with—the `rethinking` package provides tools that hide it from you for now. But as you move on to more advanced techniques, you'll be able to generate Stan versions of the models you already understand. Then you can tinker with them and witness the power of a fully armed and operational Stan.

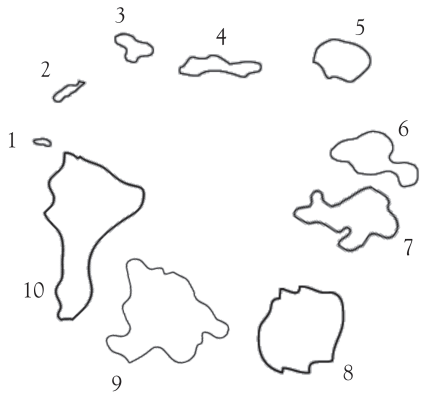


FIGURE 8.1. Good King Markov's island kingdom. Each of the 10 islands has a population proportional to its number, 1 through 10. The King's goal is to visit each island, in the long run, in proportion to its population size. This can be accomplished by the *Metropolis algorithm*.

Rethinking: Stan was a man. The Stan programming language is not an abbreviation or acronym. Rather, it is named after Stanislaw Ulam (1909–1984). Ulam is credited as one of the inventors of Markov chain Monte Carlo. Together with Ed Teller, Ulam applied it to designing fusion bombs. But he and others soon applied the general Monte Carlo method to diverse problems of less monstrous nature. Ulam made important contributions in pure mathematics, chaos theory, and molecular and theoretical biology, as well.

8.1. Good King Markov and His island kingdom

For the moment, forget about posterior densities and MCMC. Consider instead the tale of Good King Markov.[117] King Markov was a benevolent autocrat of an island kingdom, a circular archipelago, with 10 islands. Each island was neighbored by two others, and the entire archipelago formed a ring. The islands were of different sizes, and so had different sized populations living on them. The second island was about twice as populous as the first, the third about three times as populous as the first, and so on, up to the largest island, which was 10 times as populous as the smallest. The good king's island kingdom is displayed in FIGURE 8.1, with the islands numbered by their relative population sizes.

The Good King was an autocrat, but he did have a number of obligations to His people. Among these obligations, King Markov agreed to visit each island in His kingdom from time to time. Since the people love their king, each island would prefer that he visit them more often. And so everyone agreed that the king should visit each island in proportion to its population size, visiting the largest island 10 times as often as the smallest, for example.

The Good King Markov, however, wasn't one for schedules or bookkeeping, and so he wanted a way to fulfill his obligation without planning his travels months ahead of time. Also, since the archipelago was a ring, the King insisted that he only move among adjacent islands, to minimize time spent on the water—like many citizens of his kingdom, the king believes there are sea monsters in the middle of the archipelago.

The king's advisor, a Mr Metropolis, engineered a clever solution to these demands. We'll call this solution the *Metropolis algorithm*. Here's how it works.

(1) Wherever the King is, each week he decides between staying put for another week or moving to one of the two adjacent islands. To decide his next move, he flips a coin.

(2) If the coin turns up heads, the King considers moving to the adjacent island clockwise around the archipelago. If the coin turns up tails, he considers instead moving counterclockwise. Call the island the coin nominates the *proposal* island.
(3) Now, to see whether or not he moves to the proposal island, King Markov counts out a number of seashells equal to the relative population size of the proposal island. So for example, if the proposal island is number 9, then he counts out 9 seashells. Then he also counts out a number of stones equal to the relative population of the current island. So for example, if the current island is number 10, then King Markov ends up holding 10 stones, in addition to the 9 seashells.
(4) When there are more seashells than stones, King Markov always moves to the proposal island. But if there are fewer shells than stones, he discards a number of stones equal to the number of shells. So for example, if there are 4 shells and 6 stones, he ends up with 4 shells and $6 - 4 = 2$ stones. Then he places the shells and the remaining stones in a bag. He reaches in and randomly pulls out one object. If it is a shell, he moves to the proposal island. Otherwise, he stays put another week. As a result, the probability that he moves is equal to the number of shells divided by the original number of stones.

This procedure may seem baroque and, honestly, a bit crazy. But it does work. The king will appear to move around the islands randomly, sometimes staying on one island for weeks, other times bouncing around without apparent pattern. But in the long run, this procedure guarantees that the king will be found on each island in proportion to its population size.

You can prove this to yourself, by simulating King Markov's journey. Here's a short piece of code to do this, storing the history of the king's island positions in the vector `positions`:

R code 8.1

```
num_weeks <- 1e5
positions <- rep(0,num_weeks)
current <- 10
for ( i in 1:num_weeks ) {
    # record current position
    positions[i] <- current

    # flip coin to generate proposal
    proposal <- current + sample( c(-1,1) , size=1 )
    # now make sure he loops around the archipelago
    if ( proposal < 1 ) proposal <- 10
    if ( proposal > 10 ) proposal <- 1

    # move?
    prob_move <- proposal/current
    current <- ifelse( runif(1) < prob_move , proposal , current )
}
```

I've added comments to this code, to help you decipher it. The first three lines just define the number of weeks to simulate, an empty history vector, and a starting island position (the biggest island, number 10). Then the `for` loop steps through the weeks. Each week, it records the king's current position. Then it simulates a coin flip to nominate a proposal island. The only trick here lies in making sure that a proposal of "11" loops around to island 1 and a

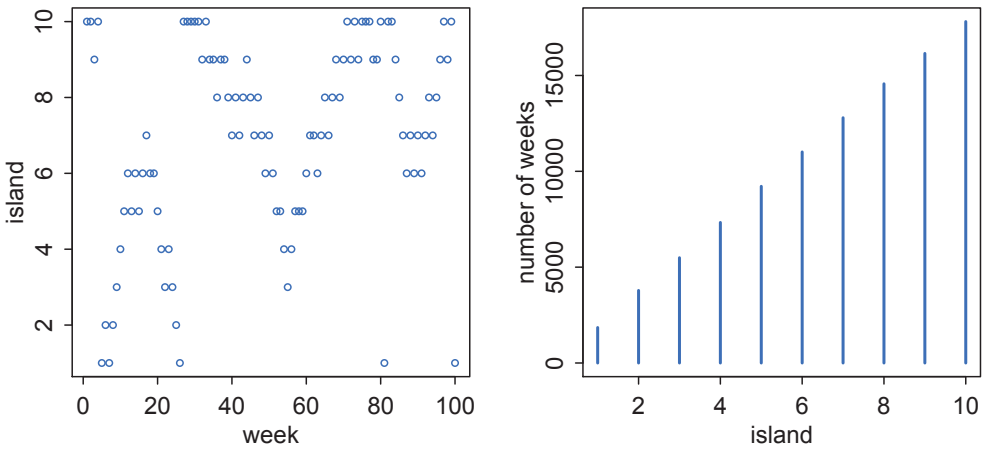


FIGURE 8.2. Results of the king following the Metropolis algorithm. The left-hand plot shows the king's position (vertical axis) across weeks (horizontal axis). In any particular week, it's nearly impossible to say where the king will be. The right-hand plot shows the long-run behavior of the algorithm, as the time spent on each island turns out to be proportional to its population size.

proposal of "0" loops around to island 10. Finally, a random number between zero and one is generated (`runif(1)`), and the king moves, if this random number is less than the ratio of the proposal island's population to the current island's population (`proposal/current`).

You can see the results of this simulation in FIGURE 8.2. The left-hand plot shows the king's location across the first 100 weeks of his simulated travels. As you move from the left to the right in this plot, the points show the king's location through time. The king travels among islands, or sometimes stays in place for a few weeks. This plot demonstrates the seemingly pointless path the Metropolis algorithm sends the king on. The right-hand plot shows that the path is far from pointless, however. The horizontal axis is now islands (and their relative populations), while the vertical is the number of weeks the king is found on each. After the entire 100,000 weeks (almost 2000 years) of the simulation, you can see that the proportion of time spent on each island converges to be almost exactly proportional to the relative populations of the islands.

The algorithm will still work in this way, even if we allow the king to be equally likely to propose a move to any island from any island, not just among neighbors. As long as King Markov still uses the ratio of the proposal island's population to the current island's population as his probability of moving, in the long run, he will spend the right amount of time on each island. The algorithm would also work for any size archipelago, even if the king didn't know how many islands were in it. All he needs to know at any point in time is the population of the current island and the population of the proposal island. Then, without any forward planning or backwards record keeping, King Markov can satisfy his royal obligation to visit his people proportionally.

8.2. Markov chain Monte Carlo

The precise algorithm King Markov used is a special case of the general **Metropolis algorithm** from the real world.[118] And this algorithm is an example of Markov chain Monte Carlo. In real applications, the goal is of course not to help an autocrat schedule his journeys, but instead to draw samples from an unknown and usually complex target distribution, like a posterior probability distribution.

- The "islands" in our objective are parameter values, and they need not be discrete, but can instead take on a continuous range of values as usual.
- The "population sizes" in our objective are the posterior probabilities at each parameter value.
- The "weeks" in our objective are samples taken from the joint posterior of the parameters in the model.

Provided the way we choose our proposed parameter values at each step is symmetric—so that there is an equal chance of proposing from A to B and from B to A—then the Metropolis algorithm will eventually give us a collection of samples from the joint posterior. We can then use these samples just like all the samples you've already used in this book.

The Metropolis algorithm is the grandparent of several different strategies for getting samples from unknown posterior distributions. In the remainder of this section, I briefly explain the concepts behind two of the most important in contemporary Bayesian inference: Gibbs sampling and Hamiltonian (aka Hybrid, aka HMC) Monte Carlo. Both are pretty common in applied Bayesian statistics. This book will use HMC, but a casual understanding of both algorithms is helpful for appreciating the advantages of each. After becoming acquainted with them, we'll turn in the second half of the chapter to using Hamiltonian Monte Carlo to do some new things with linear regression models.

8.2.1. Gibbs sampling. The Metropolis algorithm works whenever the probability of proposing a jump to B from A is equal to the probability of proposing A from B, when the proposal distribution is symmetric. There is a more general method, known as Metropolis-Hastings,[119] that allows asymmetric proposals. This would mean, in the context of King Markov's fable, that the King's coin were biased to lead him clockwise on average.

Why would we want an algorithm that allows asymmetric proposals? One reason is that it makes it easier to handle parameters, like standard deviations, that have boundaries at zero. A better reason, however, is that it allows us to generate savvy proposals that explore the posterior distribution more efficiently. By "more efficiently," I mean that we can acquire an equally good image of the posterior distribution in fewer steps.

The most common way to generate savvy proposals is a technique known as **Gibbs sampling**.[120] Gibbs sampling is a variant of the Metropolis-Hastings algorithm that uses clever proposals and is therefore more efficient. By "efficient," I mean that you can get a good estimate of the posterior from Gibbs sampling with many fewer samples than a comparable Metropolis approach. The improvement arises from *adaptive proposals* in which the distribution of proposed parameter values adjusts itself intelligently, depending upon the parameter values at the moment.

How Gibbs sampling computes these adaptive proposals depends upon using particular combinations of prior distributions and likelihoods known as *conjugate pairs*. Conjugate pairs have analytical solutions for the posterior distribution of an individual parameter. And these solutions are what allow Gibbs sampling to make smart jumps around the joint posterior distribution of all parameters.

In practice, Gibbs sampling can be very efficient, and it's the basis of popular Bayesian model fitting software like `BUGS` (Bayesian inference Using Gibbs Sampling) and `JAGS` (Just Another Gibbs Sampler). In these programs, you compose your statistical model using definitions very similar to what you've been doing so far in this book. The software automates the rest, to the best of its ability.

But there are some severe limitations to Gibbs sampling. First, maybe you don't want to use conjugate priors. Some conjugate priors seem silly, and choosing a prior so that the model fits efficiently isn't really a strong argument from a scientific perspective. Second, as models become more complex and contain hundreds or thousands or tens of thousands of parameters, Gibbs sampling becomes shockingly inefficient. In those cases, there are other algorithms.

8.2.2. Hamiltonian Monte Carlo.

> It appears to be a quite general principle that, whenever there is a randomized way of doing something, then there is a nonrandomized way that delivers better performance but requires more thought. —E. T. Jaynes[121]

The Metropolis algorithm and Gibbs sampling are both highly random procedures. They try out new parameter values and see how good they are, compared to the current values. But Gibbs sampling gains efficiency by reducing this randomness and exploiting knowledge of the target distribution. This seems to fit Jaynes' suggestion, quoted above, that when there is a random way of accomplishing some calculation, there is probably a less random way that is better. This less random way may require a lot more thought, however.

Hamiltonian Monte Carlo (or Hybrid Monte Carlo, HMC) pushes Jaynes' principle further. HMC is much more computationally costly than are Metropolis or Gibbs sampling. But its proposals are typically much more efficient. As a result, it doesn't need as many samples to describe the posterior distribution. And as models become more complex—thousands or tens of thousands of parameters—HMC can really outshine other algorithms.

We're going to be using HMC on and off for the remainder of this book. You won't have to implement it yourself. But understanding some of the concept behind it will help you grasp how it outperforms Metropolis and Gibbs sampling and also why it is not a universal solution to all MCMC problems.

Suppose King Markov's cousin Monty is King on the mainland. Monty's kingdom is not a discrete set of islands. Instead, it is a continuous territory stretched out along a narrow valley. But the King has a similar obligation: to visit his citizens in proportion to their local density. Like Markov, Monty doesn't wish to bother with schedules and calculations. So likewise he's not going to take a full census and solve for some optimal travel schedule.

Also like Markov, Monty has a highly educated and mathematically gifted advisor. His name is Hamilton. Hamilton realized that a much more efficient way to visit the citizens in the continuous Kingdom is to travel back and forth along its length. In order to spend more time in densely settled areas, they should slow the royal vehicle down when houses grow more dense. Likewise, they should speed up when houses grow more sparse. This strategy requires knowing how quickly population density is changing, at their current location. But it doesn't require remembering where they've been or knowing the population distribution anyplace else. And a major benefit of this strategy compared to that of Metropolis is that the King makes a full sweep of the kingdom before revisiting anyone.

This story is analogous to how Hamiltonian Monte Carlo works. In statistical applications, the royal vehicle is the current vector of parameter values. Let's consider the single

parameter case, just to keep things simple. In that case, the log-posterior is like a bowl, with the MAP at its nadir. Then the job is to sweep across the surface of the bowl, adjusting speed in proportion to how high up we are.

HMC really does run a physics simulation, pretending the vector of parameters gives the position of a little frictionless particle. The log-posterior provides a surface for this particle to glide across. When the log-posterior is very flat, because there isn't much information in the likelihood and the priors are rather flat, then the particle can glide for a long time before the slope (gradient) makes it turn around. When instead the log-posterior is very steep, because either the likelihood or the priors are very concentrated, then the particle doesn't get far before turning around.

All of this sounds, and is, very complex. But what is gained from all of this complexity is very efficient sampling of complex models. In cases where ordinary Metropolis or Gibbs sampling wander slowly through parameter space, Hamiltonian Monte Carlo remains efficient. This is especially true when working with multilevel models with hundreds or thousands of parameters. So HMC is becoming a popular conditioning engine.

As always, there are some limitations. HMC requires continuous parameters. It can't glide through a discrete parameter. In practice, this means that certain advanced techniques, like the imputation of discrete missing data, are not possible with HMC alone. And there are types of models that remain difficult for any MCMC strategy. HMC isn't a magic formula.

In practice, a big limitation of HMC is that it needs to be tuned to a particular model and its data—the frictionless particle does need mass, so it can acquire momentum, and the choice of mass can have big effects on efficiency. There are also a number of other parameters that define the HMC algorithm, but no the statistical model, that can change how efficiently the Markov chain samples. Tuning all of those parameters by hand is a pain. That's where an engine like Stan (mc-stan.org) comes in. Stan automates much of that tuning.[122] In the next section, you'll see how to use Stan to fit the models from earlier chapters. As the book continues, you'll encounter models that cannot be fit without some MCMC approach, so HMC and Stan will grow increasingly important.

Rethinking: The MCMC horizon. While the ideas behind Markov chain Monte Carlo are not new, widespread use dates only to the last decade of the 20th century.[123] New variants of and improvements to MCMC algorithms arise all the time. We might anticipate that interesting advances are coming, and that the current crop of tools—Gibbs sampling and HMC for example—will look rather pedestrian in another 20 years. At least we can hope.

8.3. Easy HMC: `map2stan`

The `rethinking` package provides a convenient interface, `map2stan`, to compile lists of formulas, like the lists you've been using so far to construct `map` estimates, into Stan HMC code. A little more housekeeping is needed to use `map2stan`: You need to preprocess any variable transformations, and you need to construct a clean data frame with only the variables you will use. But otherwise installing Stan on your computer is the hardest part. And once you get comfortable with interpreting samples produced in this way, you go peek inside and see exactly how the model formulas you already understand correspond to the code that drives the Markov chain.

To see how it's done, let's revisit the terrain ruggedness example from Chapter 7. This code will load the data and reduce it down to cases (nations) that have the outcome variable of interest:

R code 8.2

```
library(rethinking)
data(rugged)
d <- rugged
d$log_gdp <- log(d$rgdppc_2000)
dd <- d[ complete.cases(d$rgdppc_2000) , ]
```

So you remember the old way, we're going to repeat the procedure for fitting the interaction model. This model aims to predict log-GDP with terrain ruggedness, continent, and the interaction of the two. Here's the way to do it with `map`, just like before.

R code 8.3

```
m8.1 <- map(
    alist(
        log_gdp ~ dnorm( mu , sigma ) ,
        mu <- a + bR*rugged + bA*cont_africa + bAR*rugged*cont_africa ,
        a ~ dnorm(0,100),
        bR ~ dnorm(0,10),
        bA ~ dnorm(0,10),
        bAR ~ dnorm(0,10),
        sigma ~ dunif(0,10)
    ) ,
    data=dd )
precis(m8.1)
```

```
       Mean StdDev  5.5% 94.5%
a      9.22   0.14  9.00  9.44
bR    -0.20   0.08 -0.32 -0.08
bA    -1.95   0.22 -2.31 -1.59
bAR    0.39   0.13  0.19  0.60
sigma  0.93   0.05  0.85  1.01
```

Just as you saw in the previous chapter.

8.3.1. Preparation. But now we'll also fit this model using Hamiltonian Monte Carlo. This means there will be no more quadratic approximation—if the posterior distribution is non-Gaussian, then we'll get whatever non-Gaussian shape it has. You can use exactly the same formula list as before, but you need to do two additional things.

(1) Preprocess all variable transformations. If the outcome is transformed somehow, like by taking the logarithm, then do this before fitting the model by constructing a new variable in the data frame. Likewise, if any predictor variables are transformed, including squaring and cubing and such to build polynomial models, then compute these transformed values before fitting the model.

(2) Once you've got all the variables ready, make a new trimmed down data frame that contains only the variables you will actually use to fit the model. Technically, you don't have to do this. But doing so avoids common problems. For example, if any of the unused variables have missing values, `NA`, then Stan will refuse to work.

Here's how to do both for the terrain ruggedness model.

R code 8.4

```
dd.trim <- dd[ , c("log_gdp","rugged","cont_africa") ]
str(dd.trim)
```

```
'data.frame': 170 obs. of  3 variables:
 $ log_gdp    : num  7.49 8.22 9.93 9.41 7.79 ...
 $ rugged     : num  0.858 3.427 0.769 0.775 2.688 ...
 $ cont_africa: int  1 0 0 0 0 0 0 0 1 ...
```

The data frame `dd.trim` contains only the three variables we're using.

8.3.2. Estimation. Now provided you have the `rstan` package installed (mc-stan.org), you can get samples from the posterior distribution with this code:

R code 8.5

```
m8.1stan <- map2stan(
    alist(
        log_gdp ~ dnorm( mu , sigma ) ,
        mu <- a + bR*rugged + bA*cont_africa + bAR*rugged*cont_africa ,
        a ~ dnorm(0,100),
        bR ~ dnorm(0,10),
        bA ~ dnorm(0,10),
        bAR ~ dnorm(0,10),
        sigma ~ dcauchy(0,2)
    ) ,
    data=dd.trim )
```

There is one change to note here, but to explain later. The uniform prior on `sigma` has been changed to a half-CAUCHY prior. The Cauchy distribution is a useful thick-tailed probability distribution related to the Student t distribution. There's an Overthinking box about it later in the chapter (page 260). You can think of it as a weakly regularizing prior for standard deviations.[124] We'll use it again later in the chapter. And you'll have many chances to get used to it, as the book continues.

But note that it is not necessary to use a half-Cauchy. The uniform prior will still work, and a simple exponential prior is also appropriate. In this example, as in many, there is so much data that the prior hardly matters. There is a practice problem at the end of this chapter to guide you in comparing these priors.

After messages about translating, compiling, and sampling (see the Overthinking box later in this section for some explanations of these messages), `map2stan` returns an object that contains a bunch of summary information, as well as samples from the posterior distribution of all parameters. You can compare estimates:

R code 8.6

```
precis(m8.1stan)
```

```
       Mean StdDev lower 0.89 upper 0.89 n_eff Rhat
a      9.24   0.14       9.03       9.47   291    1
bR    -0.21   0.08      -0.32      -0.07   306    1
bA    -1.97   0.23      -2.31      -1.58   351    1
bAR    0.40   0.13       0.20       0.63   350    1
sigma  0.95   0.05       0.86       1.03   566    1
```

These estimates are very similar to the quadratic approximation. But note a few new things. First, the interval boundaries in the table just above are highest posterior density intervals (HPDI, page 56), not ordinary percentile intervals (PI). Second, there are two new columns, `n_eff` and `Rhat`. These columns provide MCMC diagnostic criteria, to help you tell how well estimation worked. We'll discuss them in detail later in the chapter. For now, it's enough to know that `n_eff` is a crude estimate of the number of independent samples you managed to get. `Rhat` is a complicated estimate of the convergence of the Markov chains to the target distribution. It should approach 1.00 from above, when all is well.

8.3.3. Sampling again, in parallel. The example so far is a very easy problem for MCMC. So even the default 1000 samples is enough for accurate inference. But often the default won't be enough. There will be specific advice in Section 8.4 (page 255).

For now, it's worth noting that once you have compiled your Stan model with `map2stan`, you can draw more samples from it anytime, running as many independent Markov chains as you like. And you can easily parallelize those chains, as well. To run four independent Markov chains for the model above, and to distribute them across separate processors in your computer, just pass the previous fit back to `map2stan`:

R code 8.7
```
m8.1stan_4chains <- map2stan( m8.1stan , chains=4 , cores=4 )
precis(m8.1stan_4chains)
```

```
       Mean StdDev lower 0.89 upper 0.89 n_eff Rhat
a      9.23   0.14       9.01       9.45  1029    1
bR    -0.20   0.08      -0.33      -0.08  1057    1
bA    -1.94   0.22      -2.29      -1.58  1154    1
bAR    0.39   0.13       0.17       0.59  1144    1
sigma  0.95   0.05       0.87       1.03  1960    1
```

The `resample` function will also recompute DIC and WAIC for you, using only the new samples. You can also add the `cores` argument to any original `map2stan` call. It will automatically parallelize the chains.

8.3.4. Visualization. By plotting the samples, you can get a direct appreciation for how Gaussian (quadratic) the actual posterior density has turned out to be. To pull out the samples, use:

R code 8.8
```
post <- extract.samples( m8.1stan )
str(post)
```

```
List of 5
 $ a    : num [1:1000(1d)] 9.3 9.34 9.21 9.43 9.28 ...
 $ br   : num [1:1000(1d)] -0.133 -0.214 -0.215 -0.229 -0.209 ...
 $ bA   : num [1:1000(1d)] -1.91 -2.25 -2.26 -1.98 -1.99 ...
 $ brA  : num [1:1000(1d)] 0.133 0.367 0.533 0.254 0.468 ...
 $ sigma: num [1:1000(1d)] 0.988 0.949 0.904 0.976 0.934 ...
```

Note that `post` here is a `list`, not a `data.frame`. This fact will be very useful to use later on, when we encounter multilevel models. For now, if it doesn't behave like you expect it to, you can coerce it to a data frame with `post<-as.data.frame(post)`. There are only 1000 samples for each parameter, because that's the default. In this case, it's enough. There's a lot

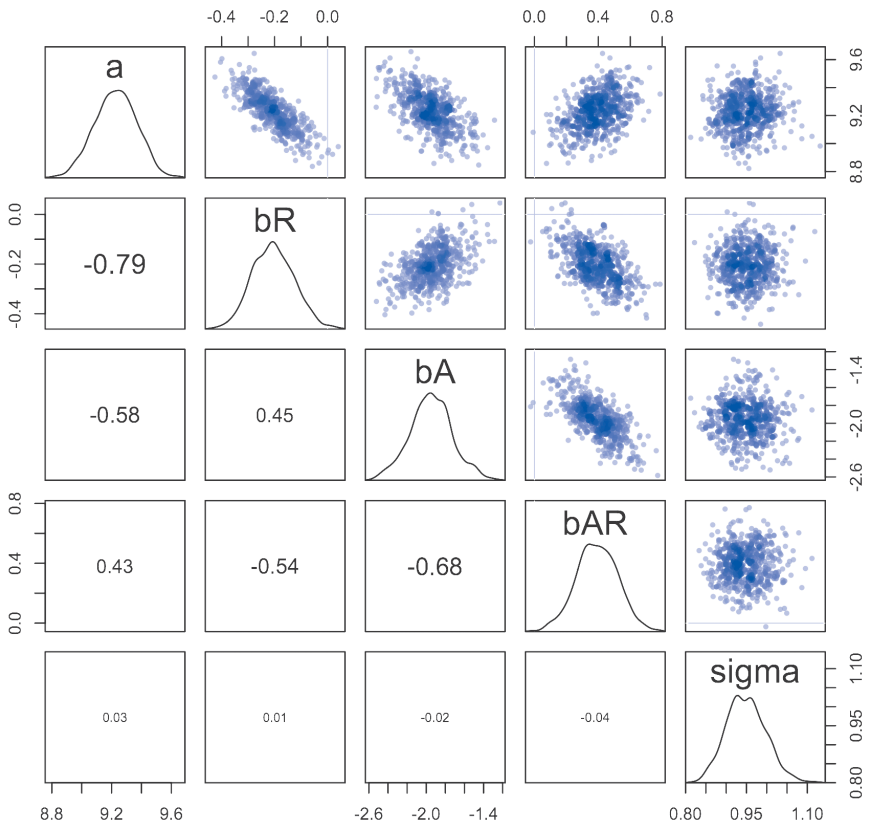


FIGURE 8.3. Pairs plot of the samples produced by Stan. The diagonal shows a density estimate for each parameter. Below the diagonal, correlations between parameters are shown.

more said about number of samples and such in the next major section of this chapter. Put those worries aside for the moment.

To plot all these samples at once, provided there aren't too many parameters, you can use the standard plotting function `pairs`:

```
pairs(post)
```

R code 8.9

Or use `pairs` directly on the fit model, so that R knows to display parameter names and parameter correlations:

```
pairs(m8.1stan)
```

R code 8.10

FIGURE 8.3 shows the resulting plot. This is a pairs plot, so it's still a matrix of bivariate scatter plots. But now along the diagonal the smoothed histogram of each parameter is shown, along with its name. And in the lower triangle of the matrix, below the diagonal, the correlation

between each pair of parameters is shown, with stronger correlations indicated by relative size.

For this model and these data, the resulting posterior distribution is quite nearly multivariate Gaussian. The density for `sigma` is certainly skewed in the expected direction. But otherwise the quadratic approximation does almost as well as Hamiltonian Monte Carlo. This is a very simple kind of model structure of course, with Gaussian priors, so an approximately quadratic posterior should be no surprise. Later, we'll see some more exotic posterior distributions.

Overthinking: Stan messages. When you fit a model using `map2stan`, R will first translate your model formula into a Stan language model. Then it sends that model to Stan. The messages you see in your R console are status updates from Stan. Stan first again translates the model, this time into C++ code. That code is then sent to a C++ compiler, to build an executable file is a specialized sampling engine for your model. Then Stan feeds the data and starting values to that executable file, and if all goes well, sampling begins. You will see Stan count through the iterations. During sampling, you might occasionally see a scary looking warning something like this:

```
Informational Message: The current Metropolis proposal is about to be rejected
because of the following issue: Error in function stan::prob::multi_normal_log
(N4stan5agrad3varE):Covariance matrix is not positive definite. Covariance
matrix(0,0) is 0:0. If this warning occurs sporadically, such as for highly
constrained variable types like covariance matrices, then the sampler is
fine, but if this warning occurs often then your model may be either severely
ill-conditioned or misspecified.
```

Severely ill-conditioned or misspecified? That certainly sounds bad. But rarely does this message indicate a serious problem. As long as it happens only a handful of times, and especially if it only happens during warmup, then odds are very good the chain is fine. You should still always check the chain for problems, of course. Just don't panic when you see this message. Keep calm and sample on.

8.3.5. Using the samples. Once you have samples in an object like `post`, you work with them just as you've already learned to do. If you have the samples from the posterior and you know the model, you can do anything: simulate predictions, compute differences between parameters, and calculate DIC and WAIC.

By default `map2stan` computes DIC and WAIC for you. You can extract them with `DIC(m8.1stan)` and `WAIC(m8.1stan)`. DIC and WAIC are also reported in the default `show` output for a `map2stan` model fit.

R code 8.11

```
show(m8.1stan)
```

```
map2stan model fit
1000 samples from 1 chain

Formula:
log_gdp ~ dnorm(mu, sigma)
mu <- a + bR * rugged + bA * cont_africa + bAR * rugged * cont_africa
a ~ dnorm(0, 100)
bR ~ dnorm(0, 10)
bA ~ dnorm(0, 10)
bAR ~ dnorm(0, 10)
sigma ~ dcauchy(0, 2)
```

```
Log-likelihood at expected values: -229.43
Deviance: 458.85
DIC: 468.73
Effective number of parameters (pD): 4.94

WAIC (SE): 469.3 (14.8)
pWAIC: 5.14
```

This report just reiterates the formulas used to define the Markov chain and then reports information criteria.

For computing predictions, the functions `postcheck`, `link`, and `sim` work on `map2stan` models just as they do on `map` models. For model comparison, `compare` and `ensemble` also work the same way. Regardless of how you fit and get samples from a model, once you have those samples, the logic is always the same: Process the samples to address questions about the relative plausibility of different parameter values and implied predictions.

8.3.6. Checking the chain. Provided the Markov chain is defined correctly—and it is here—then it is guaranteed to converge in the long run to the answer we want, the posterior distribution. But the machine does sometimes malfunction. In the next major section, we'll dwell on causes of and solutions to malfunction.

For now, let's meet the most broadly useful tool for diagnosing malfunction, a **TRACE PLOT**. A trace plot merely plots the samples in sequential order, joined by a line. It's King Markov's path through the islands, in the metaphor at the start of the chapter. Looking at the trace plot of each parameter can help to diagnose many common problems. And once you come to recognize a healthy, functioning Markov chain, quick checks of trace plots provide a lot of peace of mind. A trace plot isn't the last thing analysts do to inspect MCMC output. But it's nearly always the first.

In the terrain ruggedness example, the trace plot shows a very healthy chain. View it with:

```
plot(m8.1stan)
```

R code 8.12

The result is shown in FIGURE 8.4. Each plot in this figure is similar to what you'd get if you just used, for example, `plot(post$a,type="l")`, but with some extra information and labeling to help out. You can think of the zig-zagging trace of each parameter as the path the chain took through each dimension of parameter space.

The gray region in each plot, the first 1000 samples, marks the *adaptation* samples. During adaptation, the Markov chain is learning to more efficiently sample from the posterior distribution. So these samples are not necessarily reliable to use for inference. They are automatically discarded by `extract.samples`, which returns only the samples shown in the white regions of FIGURE 8.4.

Now, how is this chain a healthy one? Typically we look for two things in these trace plots: stationarity and good mixing. Stationarity refers to the path staying within the posterior distribution. Notice that these traces, for example, all stick around a very stable central tendency, the center of gravity of each dimension of the posterior. Another way to think of this is that the mean value of the chain is quite stable from beginning to end.

A well-mixing chain means that each successive sample within each parameter is not highly correlated with the sample before it. Visually, you can see this by the rapid zig-zag

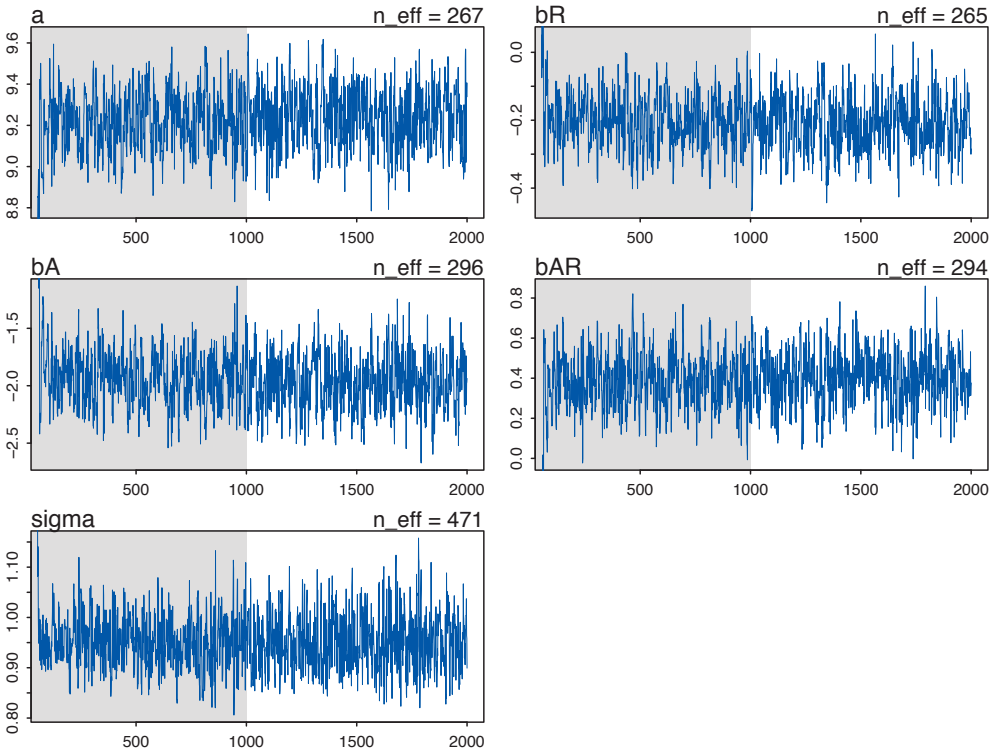


FIGURE 8.4. Trace plot of the Markov chain from the ruggedness model, `m8.1stan`. This is a clean, healthy Markov chain, both stationary and well-mixing. The gray region is warmup, during which the Markov chain was adapting to improve sampling efficiency. The white region contains the samples used for inference.

motion of each path, as the trace traverses the posterior distribution without getting mired anyplace.

To really understand these points, though, you'll have to see some trace plots for unhealthy chains. That's the project of the next section.

Overthinking: Raw Stan model code. All `map2stan` does is translate a list of formulas into Stan's modeling language. Then Stan does the rest. Learning how to write Stan code is not necessary for most of the models in this book. But other models do require some direct interaction with Stan, because it is capable of much more than `map2stan` allows you to express. And even for simple models, you'll gain additional comprehension and control, if you peek into the machine. You can always access the raw Stan code that `map2stan` produces by using the function `stancode`. For example, `stancode(m8.1stan)` prints out the Stan code for the ruggedness model. Before you're familiar with Stan's language, it'll look long and weird. So let's focus on just the most important part, the "model block":

```
model{
    vector[N] mu;
    sigma ~ cauchy( 0 , 2 );
    bAR ~ normal( 0 , 10 );
    bA ~ normal( 0 , 10 );
    bR ~ normal( 0 , 10 );
```

```
    a ~ normal( 0 , 100 );
    for ( i in 1:N ) {
        mu[i] <- a + bR * rugged[i] + bA * cont_africa[i] +
                 bAR * rugged[i] * cont_africa[i];
    }
    log_gdp ~ normal( mu , sigma );
}
```

This is Stan code, not R code. It is essentially the formula list you provided to `map2stan`, but in reverse order. The first line, `vector[N] mu;` names a symbol to hold the linear model, the kind of explicit housekeeping many computer languages require but that R does not. The rest of the code, however, just reiterates your formulas, beginning with priors and then computing the value of `mu` for each observation (nation). Finally, the last line comprises the likelihood. Given this kind of model definition, Stan will define and sample from the HMC chain or chains you ask for.

8.4. Care and feeding of your Markov chain

Markov chain Monte Carlo is a highly technical and usually automated procedure. Most people who use it don't really understand what it is doing. That's okay, up to a point. Science requires division of labor, and if every one of us had to write our own Markov chains from scratch, a lot less research would get done in the aggregate.

But as with many technical and powerful procedures, it's natural to feel uneasy about MCMC and maybe even a little superstitious. Something magical is happening inside the computer, and unless we make the right sacrifices and say the right words, the magic might become a curse. The good news is that HMC, unlike Gibbs sampling and ordinary Metropolis, makes it easy to tell when the magic goes wrong.

8.4.1. How many samples do you need? You can control the number of samples from the chain by using the `iter` and `warmup` parameters. The defaults are 2000 for `iter` and `warmup` is set to `iter/2`, which gives you 1000 warmup samples and 1000 real samples to use for inference. But these defaults are just meant to get you started, to make sure the chain gets started okay. Then you can decide on other values for `iter` and `warmup`.

So how many samples do we need for accurate inference about the posterior distribution? It depends. First, what really matters is the *effective* number of samples, not the raw number. The effective number of samples is an estimate of the number of independent samples from the posterior distribution. Markov chains are typically *autocorrelated*, so that sequential samples are not entirely independent. Stan chains tend to be less autocorrelated than those produced by other engines, but there is always some autocorrelation. Stan provides an estimate of effective number of samples as `n_eff`, and you'll see examples a bit later in the chapter.

Second, what do you want to know? If all you want are posterior means, it doesn't take many samples at all to get very good estimates. Even a couple hundred samples will do. But if you care about the exact shape in the extreme tails of the posterior, the 99th percentile or so, then you'll need many many more. So there is no universally useful number of samples to aim for. In most typical regression applications, you can get a very good estimate of the posterior mean with as few as 200 effective samples. And if the posterior is approximately Gaussian, then all you need in addition is a good estimate of the variance, which can be had with one order of magnitude more, in most cases. For highly skewed posteriors, you'll have to think more about which region of the distribution interests you.

The warmup setting is more subtle. On the one hand, you want to have the shortest warmup period necessary, so you can get on with real sampling. But on the other hand, more warmup can mean more efficient sampling. With Stan models, typically you can devote as much as half of your total samples, the `iter` value, to warmup and come out very well. But for simple models like those you've fit so far, much less warmup is really needed. Models can vary a lot in the shape of their posterior distributions, so again there is no universally best answer. But if you are having trouble, you might try increasing the warmup. If not, you might try reducing it. There's a practice problem at the end of the chapter that guides you in experimenting with the amount of warmup.

Rethinking: Warmup is not burn-in. Other MCMC algorithms and software often discuss **BURN-IN**. With a sampling strategy like ordinary Metropolis, it is conventional and useful to trim off the front of the chain, the "burn-in" phase. This is done because it is unlikely that the chain has reached stationarity within the first few samples. Trimming off the front of the chain hopefully removes any influence of which starting value you chose for a parameter.[125]

But Stan's sampling algorithms use a different approach. What Stan does during warmup is quite different from what it does after warmup. The warmup samples are used to adapt sampling, and so are not actually part of the target posterior distribution at all, no matter how long warmup continues. They are not burning in, but rather more like cycling the motor to heat things up and get ready for sampling. When real sampling begins, the samples will be immediately from the target distribution, assuming adaptation was successful. Still, you can usually tell if adaptation was successful because the warmup samples will come to look very much like the real samples. But that isn't always the case. For bad chains, the warmup will often look pretty good, but then actual sampling will demonstrate severe problems. You'll see examples a bit later in the chapter.

8.4.2. How many chains do you need? It is very common to run more than one Markov chain, when estimating a single model. To do this with `map2stan` or `stan` itself, the `chains` argument specifies the number of independent Markov chains to sample from. And the optional `cores` argument lets you distribute the chains across different processors, so they can run simultaneously, rather than sequentially. All of the non-warmup samples from each chain will be automatically combined in the resulting inferences.

So the question naturally arises: How many chains do we need? There are three answers to this question. First, when debugging a model, use a single chain. Then when deciding whether the chains are valid, you need more than one chain. Third, when you begin the final run that you'll make inferences from, you only really need one chain. But using more than one chain is fine, as well. It just doesn't matter, once you're sure it's working. I'll briefly explain these answers.

The first time you try to sample from a chain, you might not be sure whether the chain is working right. So of course you will check the trace plot. Having more than one chain during these checks helps to make sure that the Markov chains are all converging to the same distribution. Sometimes, individual chains look like they've settled down to a stable distribution, but if you run the chain again, it might settle down to a different distribution. When you run multiple Markov chains, and see that all of them end up in the same region of parameter space, it provides a check that the machine is working correctly. Using 3 or 4 chains is conventional, and quite often more than enough to reassure us that the sampling is working properly.

But once you've verified that the sampling is working well, and you have a good idea of how many warmup samples you need, it's perfectly safe to just run one long chain. For

example, suppose we learn that we need 1000 warmup samples and about 9000 real samples in total. Should we run one chain, with `warmup=1000` and `iter=10000`, or rather 3 chains, with `warmup=1000` and `iter=4000`? It doesn't really matter, in terms of inference.

But it might matter in efficiency, because the 3 chains cost you an extra 2000 samples of warmup that just get thrown away. And since warmup is typically the slowest part of the chain, these extra 2000 samples cost a disproportionate amount of your computer's time. On the other hand, if you run the chains on different computers or processor cores within a single computer, then you might prefer 3 chains, because you can spread the load and finish the whole job faster.

There are exotic situations in which all of the advice above must be modified. But for typical regression models, you can live by the motto *four short chains to check, one long chain for inference*. Things may still go wrong—you'll see some examples in the next sections, so you know what to look for. And once you know what to look for, you can fix any problems before running a long final Markov chain.

One of the perks of using HMC and Stan is that when sampling isn't working right, it's usually very obvious. As you'll see in the sections to follow, bad chains tend to have conspicuous behavior. Other methods of MCMC sampling, like Gibbs sampling and ordinary Metropolis, aren't so easy to diagnose.

Rethinking: Convergence diagnostics. The default diagnostic output from Stan includes two metrics, `n_eff` and `Rhat`. The first is a measure of the effective number of samples. The second is the Gelman-Rubin convergence diagnostic, $\hat{R}$.[126] When `n_eff` is much lower than the actual number of iterations (minus warmup) of your chains, it means the chains are inefficient, but possibly still okay. When `Rhat` is above 1.00, it usually indicates that the chain has not yet converged, and probably you shouldn't trust the samples. If you draw more iterations, it could be fine, or it could never converge. See the Stan user manual for more details. It's important however not to rely too much on these diagnostics. Like all heuristics, there are cases in which they provide poor advice. For example, `Rhat` can reach 1.00 even for an invalid chain. So view it perhaps as a signal of danger, but never of safety. For conventional models, these metrics typically work well.

8.4.3. Taming a wild chain. One common problem with some models is that there are broad, flat regions of the posterior density. This happens most often, as you might guess, when one uses flat priors. The problem this can generate is a wild, wandering Markov chain that erratically samples extremely positive and extremely negative parameter values.

Let's look at a simple example. The code below tries to estimate the mean and standard deviation of the two Gaussian observations −1 and 1. But it uses totally flat priors.

R code 8.13

```
y <- c(-1,1)
m8.2 <- map2stan(
    alist(
        y ~ dnorm( mu , sigma ) ,
        mu <- alpha
    ) ,
    data=list(y=y) , start=list(alpha=0,sigma=1) ,
    chains=2 , iter=4000 , warmup=1000 )
```

Now let's look at the `precis` output:

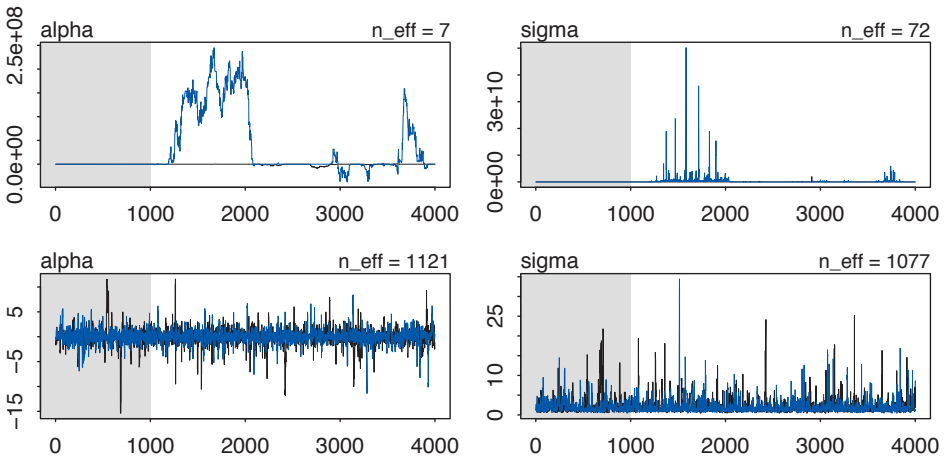


FIGURE 8.5. Diagnosing and healing a sick Markov chain. Top row: Trace plot from two independent chains defined by model `m8.2`. These chains are not stationary and should not be used for inference. Bottom row: Adding weakly informative priors (see `m8.3`) clears up the condition right away. These chains are fine to use for inference.

R code 8.14

```
precis(m8.2)
```

```
           Mean      StdDev    lower 0.89 upper 0.89 n_eff Rhat
alpha  21583691    54448550 -19611287.92  129922812     7 1.36
sigma 139399593 1147514738        29.06  185868167    72 1.02
```

Whoa! Those estimates can't be right. The mean of −1 and 1 is zero, so we're hoping to get a mean value for `alpha` around zero. Instead we get crazy values and implausibly wide confidence intervals. Inference for `sigma` is no better. You can also see that the diagnostic criteria indicate unreliable estimates. The number of effective samples, `n_eff`, is very small. And `Rhat` should approach 1.00 in a healthy set of chains. Even a value of 1.01 is suspicious. An `Rhat` of 1.10 indicates a catastrophe.

Take a look at the trace plot for this fit, `plot(m8.2)`. It's shown in the top row of FIGURE 8.5. The reason for the weird estimates is that the Markov chains seem to drift around and spike occasionally to extreme values. This is not a stationary pair of chains, and they do not provide useful samples.

It's easy to tame this particular chain by using weakly informative priors. The reason the model above drifts wildly in both dimensions is that there is very little data, just two observations, and flat priors. The flat priors say that every possible value of the parameter is equally plausible, a priori. For parameters that can take a potentially infinite number of values, like `alpha`, this means the Markov chain needs to occasionally sample some pretty extreme and implausible values, like negative 30 million. These extreme drifts overwhelm the chain. If the likelihood were stronger, then the chain would be fine, because it would stick closer to zero.

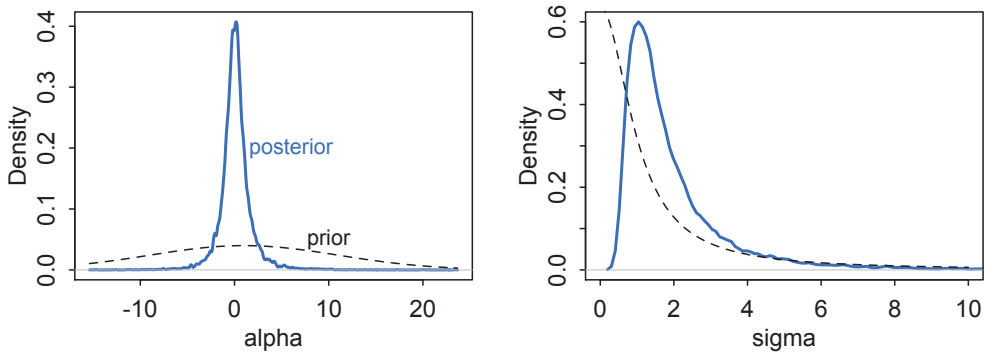


FIGURE 8.6. Prior (dashed) and posterior (blue) for the model with weakly informative priors, `m8.3`. Even with only two observations, the likelihood easily overcomes these priors. Yet the model cannot be successfully estimated without them.

But it doesn't take much information in the prior to stop this foolishness, even without a stronger likelihood. Let's use this model:

$$\begin{aligned} y_i &\sim \text{Normal}(\mu, \sigma) \\ \mu &= \alpha \\ \alpha &\sim \text{Normal}(1, 10) \\ \sigma &\sim \text{HalfCauchy}(0, 1) \end{aligned}$$

I've just added weakly informative priors for α and σ. We'll plot these priors in a moment, so you will be able to see just how weak they are. But let's re-estimate first:

R code 8.15

```
m8.3 <- map2stan(
    alist(
        y ~ dnorm( mu , sigma ) ,
        mu <- alpha ,
        alpha ~ dnorm( 1 , 10 ) ,
        sigma ~ dcauchy( 0 , 1 )
    ) ,
    data=list(y=y) , start=list(alpha=0,sigma=1) ,
    chains=2 , iter=4000 , warmup=1000 )
precis(m8.3)
```

```
       Mean StdDev lower 0.89 upper 0.89 n_eff Rhat
alpha -0.01   1.60      -1.98       2.37  1121    1
sigma  1.98   1.91       0.47       3.45  1077    1
```

That's much better. Take a look at the bottom row in FIGURE 8.5. This trace plot looks healthy. Both chains are stationary around the same values, and mixing is good. No more wild detours off to negative 20 million.

To appreciate what has happened, take a look at the priors (dashed) and posteriors (blue) in FIGURE 8.6. Both the Gaussian prior for α and the Cauchy prior for σ contain very gradual

downhill slopes. They are so gradual, that even with only two observations, as in this example, the likelihood almost completely overcomes them. The mean of the prior for α is 1, but the mean of the posterior is zero, just as the likelihood says it should be. The prior for σ is maximized at zero. But the posterior has its median around 1.4. The standard deviation of the data is 1.4.

These weakly informative priors have helped by providing a very gentle nudge towards reasonable values of the parameters. Now values like 30 million are no longer equally plausible as small values like 1 or 2. Lots of problematic chains want subtle priors like these, designed to tune estimation by assuming a tiny bit of prior information about each parameter. And even though the priors end up getting washed out right away—two observations were enough here—they still have a big effect on inference, by allowing us to get an answer. That answer is also a good answer.

Overthinking: Cauchy distribution. The models in this chapter, and in many chapters to follow, use half-Cauchy priors for standard deviations. The **Cauchy** (ko-shee) distribution gives the distribution of the ratio of two random Gaussian draws. Its parameters are a *location* x_0 and a *scale* γ. The location says where the center is, and the scale defines how stretched out the distribution is. Its probability density is:

$$p(x|x_0,\gamma) = \left(\pi\gamma\left[1+\left(\frac{x-x_0}{\gamma}\right)^2\right]\right)^{-1}$$

Note however that the Cauchy has no defined mean nor variance, so the location and scale are not its mean and, say, standard deviation. The reason the Cauchy has no mean and variance is that it is a very thick-tailed distribution. At any moment in a Cauchy sampling process, it is possible to draw an extreme value that overwhelms all of the previous draws. The consequence of this fact is that the sequence never converges to a stable mean and variance. It just keeps moving. You can prove this to yourself with a little simulation. The code below samples 10,000 values from a Cauchy distribution. Then it computes and plots the running mean at each sample. Run this simulation a few times to see how the trace of the mean is highly unpredictable.

R code 8.16

```
y <- rcauchy(1e4,0,5)
mu <- sapply( 1:length(y) , function(i) sum(y[1:i])/i )
plot(mu,type="l")
```

The Cauchy distributions in the model definitions are implicitly half-Cauchy, a Cauchy defined over the positive reals only. This is because they are applied to a parameter, usually σ, that is strictly positive. Stan figures out that you meant for it to be half-Cauchy. If you are curious how it knows this, check the raw Stan code with `stancode` and look for the `<lower=0>` constraint in the definition of the parameter `sigma`.

8.4.4. Non-identifiable parameters. Back in Chapter 5, you met the problem of highly correlated predictors and the non-identifiable parameters they can create. Here you'll see what such parameters look like inside of a Markov chain. You'll also see how you can identify them, in principle, by using a little prior information. Most importantly, the badly behaving chains produced in this example will exhibit characteristic bad behavior, so when you see the same pattern in your own models, you'll have a hunch about the cause.

To construct a non-identifiable model, we first simulate 100 observations from a Gaussian distribution with mean zero and standard deviation 1.

```
y <- rnorm( 100 , mean=0 , sd=1 )
```

R code 8.17

By simulating the data, we know the right answer. Then we fit this model:

$$y_i \sim \text{Normal}(\mu, \sigma)$$
$$\mu = \alpha_1 + \alpha_2$$
$$\sigma \sim \text{HalfCauchy}(0, 1)$$

The linear model contains two parameters, α_1 and α_2, which cannot be identified. Only their sum can be identified, and it should be about zero, after estimation.

Let's run the Markov chain and see what happens. This chain is going to take much longer than the previous ones. But it should still finish after a few minutes.

```
m8.4 <- map2stan(
    alist(
        y ~ dnorm( mu , sigma ) ,
        mu <- a1 + a2 ,
        sigma ~ dcauchy( 0 , 1 )
    ) ,
    data=list(y=y) , start=list(a1=0,a2=0,sigma=1) ,
    chains=2 , iter=4000 , warmup=1000 )
precis(m8.4)
```

R code 8.18

```
          Mean  StdDev lower 0.89 upper 0.89 n_eff Rhat
a1    -1194.76 1344.19   -2928.62    1053.52     1 2.83
a2     1194.81 1344.19   -1054.86    2927.39     1 2.83
sigma     0.92    0.07       0.81       1.02    17 1.13
```

Those estimates look suspicious, and the `n_eff` and `Rhat` values are terrible. The means for `a1` and `a2` are almost exactly the same distance from zero, but on opposite sides. And the standard deviations of the chains are massive. This is of course a result of the fact that we cannot simultaneously estimate `a1` and `a2`, but only their sum.

Looking at the trace plot reveals more. The left column in Figure 8.7 shows two Markov chains from the model above. These chains do not look like they are stationary, nor do they seem to be mixing very well. Indeed, when you see a pattern like this, it is reason to worry. Don't use these samples.

Again, weak priors can rescue us. Now the model fitting code is:

```
m8.5 <- map2stan(
    alist(
        y ~ dnorm( mu , sigma ) ,
        mu <- a1 + a2 ,
        a1 ~ dnorm( 0 , 10 ) ,
        a2 ~ dnorm( 0 , 10 ) ,
        sigma ~ dcauchy( 0 , 1 )
    ) ,
    data=list(y=y) , start=list(a1=0,a2=0,sigma=1) ,
    chains=2 , iter=4000 , warmup=1000 )
precis(m8.5)
```

R code 8.19

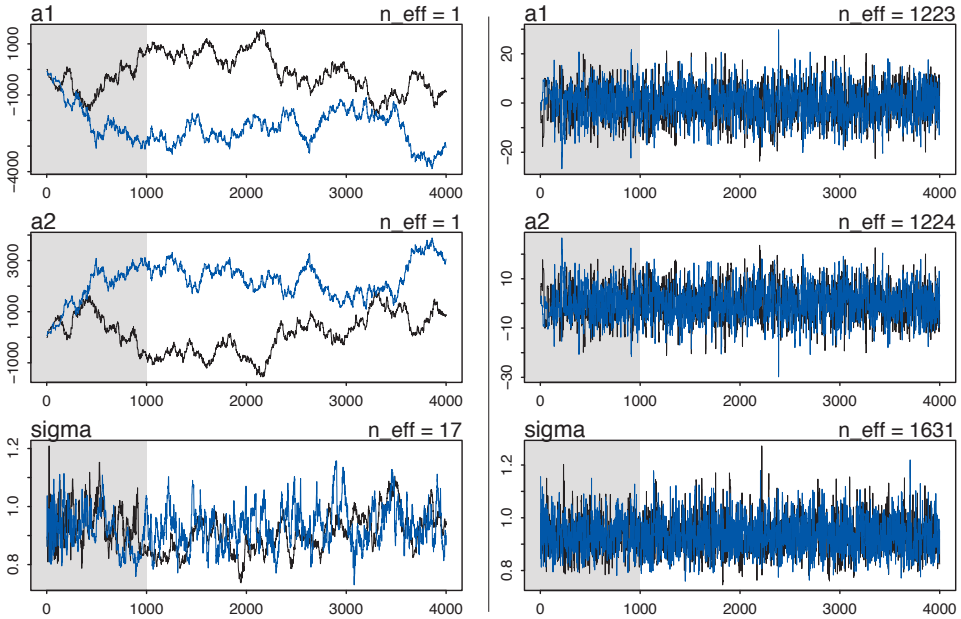


FIGURE 8.7. Left column: A chain with wandering parameters, `a1` and `a2`, generated by `m8.4`. Right column: Same model but now with weakly informative priors, `m8.5`.

```
       Mean StdDev lower 0.89 upper 0.89 n_eff Rhat
a1    -0.23   6.97      -11.25       10.95  1223    1
a2     0.28   6.97      -10.85       11.36  1224    1
sigma  0.93   0.07        0.82        1.04  1631    1
```

The estimates for `a1` and `a2` are better identified now. And take a look at the right column traces in FIGURE 8.7. Notice also that the model sampled a lot faster. With flat priors, `m8.4`, sampling may take 8 times as long as it does for `m8.5`. Often, a model that is very slow to sample is under-identified. This is an aspect of something Bayesian statistician Andrew Gelman calls the **FOLK THEOREM OF STATISTICAL COMPUTING**: When you are having trouble fitting a model, it often indicates a bad model.

In the end, adding some weakly informative priors saves this model. You might think you'd never accidentally try to fit an unidentified model. But you'd be wrong. Even if you don't make obvious mistakes, complex models can easily become unidentified or nearly so. With many predictors, and especially with interactions, correlations among parameters can be large. Just a little prior information telling the model "none of these parameters can be 30 million" often helps, and it has no effect on estimates. A flat prior really is flat, all the way to infinity. Unless you believe infinity is a reasonable estimate, don't use a flat prior.

Additionally, adding weak priors can speed up sampling, because the Markov chain won't feel that it has to run out to extreme values that you, but not your model, already know are highly implausible.

8.5. Summary

This chapter has been an informal introduction to Markov chain Monte Carlo (MCMC) estimation. The goal has been to introduce the purpose and approach MCMC algorithms. The major algorithms introduced were the Metropolis, Gibbs sampling, and Hamiltonian Monte Carlo algorithms. Each has its advantages and disadvantages. A function in the `rethinking` package, `map2stan`, was introduced that uses the Stan (mc-stan.org) Hamiltonian Monte Carlo engine to fit models as they are defined in this book. General advice about diagnosing poor MCMC fits was introduced by the use of a couple of pathological examples.

8.6. Practice

Easy.

8E1. Which of the following is a requirement of the simple Metropolis algorithm?

(1) The parameters must be discrete.
(2) The likelihood function must be Gaussian.
(3) The proposal distribution must be symmetric.

8E2. Gibbs sampling is more efficient than the Metropolis algorithm. How does it achieve this extra efficiency? Are there any limitations to the Gibbs sampling strategy?

8E3. Which sort of parameters can Hamiltonian Monte Carlo not handle? Can you explain why?

8E4. Explain the difference between the effective number of samples, `n_eff` as calculated by Stan, and the actual number of samples.

8E5. Which value should `Rhat` approach, when a chain is sampling the posterior distribution correctly?

8E6. Sketch a good trace plot for a Markov chain, one that is effectively sampling from the posterior distribution. What is good about its shape? Then sketch a trace plot for a malfunctioning Markov chain. What about its shape indicates malfunction?

Medium.

8M1. Re-estimate the terrain ruggedness model from the chapter, but now using a uniform prior and an exponential prior for the standard deviation, `sigma`. The uniform prior should be `dunif(0,10)` and the exponential should be `dexp(1)`. Do the different priors have any detectible influence on the posterior distribution?

8M2. The Cauchy and exponential priors from the terrain ruggedness model are very weak. They can be made more informative by reducing their scale. Compare the `dcauchy` and `dexp` priors for progressively smaller values of the scaling parameter. As these priors become stronger, how does each influence the posterior distribution?

8M3. Re-estimate one of the Stan models from the chapter, but at different numbers of `warmup` iterations. Be sure to use the same number of sampling iterations in each case. Compare the `n_eff` values. How much warmup is enough?

Hard.

8H1. Run the model below and then inspect the posterior distribution and explain what it is accomplishing.

R code 8.20

```
mp <- map2stan(
    alist(
        a ~ dnorm(0,1),
        b ~ dcauchy(0,1)
    ),
    data=list(y=1),
    start=list(a=0,b=0),
    iter=1e4, warmup=100 , WAIC=FALSE )
```

Compare the samples for the parameters `a` and `b`. Can you explain the different trace plots, using what you know about the Cauchy distribution?

8H2. Recall the divorce rate example from Chapter 5. Repeat that analysis, using `map2stan` this time, fitting models `m5.1`, `m5.2`, and `m5.3`. Use `compare` to compare the models on the basis of WAIC. Explain the results.

8H3. Sometimes changing a prior for one parameter has unanticipated effects on other parameters. This is because when a parameter is highly correlated with another parameter in the posterior, the prior influences both parameters. Here's an example to work and think through.

Go back to the leg length example in Chapter 5. Here is the code again, which simulates height and leg lengths for 100 imagined individuals:

R code 8.21

```
N <- 100                          # number of individuals
height <- rnorm(N,10,2)           # sim total height of each
leg_prop <- runif(N,0.4,0.5)      # leg as proportion of height
leg_left <- leg_prop*height +     # sim left leg as proportion + error
    rnorm( N , 0 , 0.02 )
leg_right <- leg_prop*height +    # sim right leg as proportion + error
    rnorm( N , 0 , 0.02 )
                                  # combine into data frame
d <- data.frame(height,leg_left,leg_right)
```

And below is the model you fit before, resulting in a highly correlated posterior for the two beta parameters. This time, fit the model using `map2stan`:

R code 8.22

```
m5.8s <- map2stan(
    alist(
        height ~ dnorm( mu , sigma ) ,
        mu <- a + bl*leg_left + br*leg_right ,
        a ~ dnorm( 10 , 100 ) ,
        bl ~ dnorm( 2 , 10 ) ,
        br ~ dnorm( 2 , 10 ) ,
        sigma ~ dcauchy( 0 , 1 )
    ) ,
    data=d, chains=4,
    start=list(a=10,bl=0,br=0,sigma=1) )
```

Compare the posterior distribution produced by the code above to the posterior distribution produced when you change the prior for `br` so that it is strictly positive:

R code 8.23

```
m5.8s2 <- map2stan(
    alist(
        height ~ dnorm( mu , sigma ) ,
        mu <- a + bl*leg_left + br*leg_right ,
        a ~ dnorm( 10 , 100 ) ,
        bl ~ dnorm( 2 , 10 ) ,
        br ~ dnorm( 2 , 10 ) & T[0,] ,
        sigma ~ dcauchy( 0 , 1 )
    ) ,
    data=d, chains=4,
    start=list(a=10,bl=0,br=0,sigma=1) )
```

Note that `T[0,]` on the right-hand side of the prior for `br`. What the `T[0,]` does is *truncate* the normal distribution so that it has positive probability only above zero. In other words, that prior ensures that the posterior distribution for `br` will have no probability mass below zero.

Compare the two posterior distributions for `m5.8s` and `m5.8s2`. What has changed in the posterior distribution of both beta parameters? Can you explain the change induced by the change in prior?

8H4. For the two models fit in the previous problem, use DIC or WAIC to compare the effective numbers of parameters for each model. Which model has more effective parameters? Why?

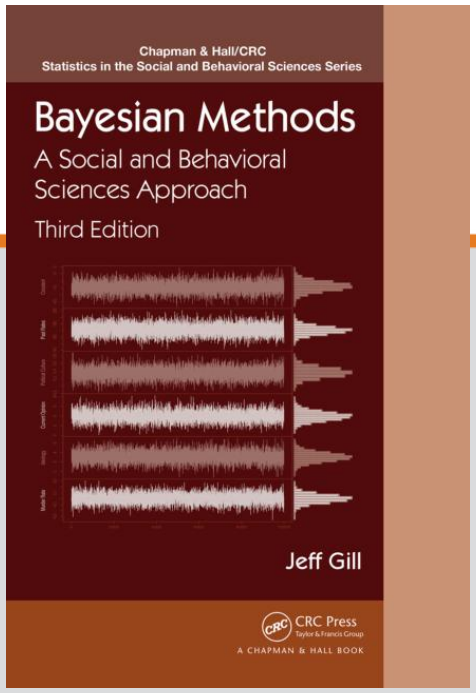
8H5. Modify the Metropolis algorithm code from the chapter to handle the case that the island populations have a different distribution than the island labels. This means the island's number will not be the same as its population.

8H6. Modify the Metropolis algorithm code from the chapter to write your own simple MCMC estimator for globe tossing data and model from Chapter 2.

CHAPTER

3

SPECIFYING BAYESIAN MODELS



This chapter is excerpted from

Bayesian Methods: A Social and Behavioral Sciences Approach, Third Edition

by Jeff Gill.

Learn more

Chapter 2

Specifying Bayesian Models

2.1 Purpose

This chapter changes the discussion from the basic workings of Bayes' Law in a probability context to a focus on the use of Bayes' Law for realistic statistical models. Consequently, the first order of business is to go from our previous vague definition of data, $\mathbf{D}$, to a rectangular $n \times k$ matrix of data, $\mathbf{X}$. In this chapter we also make the move from the unspecific $p()$ for posterior distributions to the more clear $\pi()$ notation in order to distinguish them from priors, likelihoods, and other functions. Also, from now on we use the vector form of theta, $\boldsymbol{\theta}$, since nearly all interesting social science models are multidimensional.

In the immediately forthcoming material, we cover the core idea of Bayesian statistics: updating prior distributions by conditioning on data through the likelihood function. We will also look at repeating this updating process as new information becomes available. There is an additional historical discussion placing this modeling approach into context.

2.2 Likelihood Theory and Estimation

In order to make inferences about unknown model parameters in generalized linear models, Bayesian probability models, or any other parametric specification, we would like to have a description of parameter values that are more or less probable given the observed data and the parametric form of the model. In other words, some values of an unknown parameter are certainly more likely to have generated the data than others, and if there is one value that is more likely than all others, we would typically prefer to report that one.

For instance, suppose we wanted to know the probability of getting a heads with a possibly unfair coin. Flipping it ten times, we observe 5 heads. It seems logical to infer that $p = 0.5$ is more likely than $p = 0.4$, or $p = 0.6$, or any other value for that matter. In this case, $p = 0.5$ is the value that maximizes the likelihood function given the observed series of flips. Maximizing a likelihood function with regard to coefficient values is without question the most frequently used estimation technique in applied statistics.

Stipulate now that we are interested in analyzing a model for a k-dimensional unknown $\boldsymbol{\theta}$ vector, $k-1$ explanatory variables, a constant, and n data points. Asymptotic theory assures us that for sufficiently large samples the likelihood surface is unimodal in k dimensions for the commonly used forms (Lehmann 1999). Denote this likelihood function as $L(\boldsymbol{\theta}|\mathbf{X})$ even though it is constructed as the joint distribution of the iid outcomes: $p(\mathbf{X}|\boldsymbol{\theta}) = f(x_1|\boldsymbol{\theta})f(x_2|\boldsymbol{\theta})\cdots f(x_n|\boldsymbol{\theta})$.

The likelihood function differs from the inverse probability, $p(\boldsymbol{\theta}|\mathbf{X})$, in that it is necessarily a *relative* function since it is not a normalized probability measure bounded by zero and one. From a frequentist standpoint, the probabilistic uncertainty is a characteristic of the random variable $\mathbf{X}$, not the unknown but fixed $\boldsymbol{\theta}$. Barnett (1973, p.131) clarifies this distinction: "Probability remains attached to X, not θ; it simply reflects inferentially on θ." Thus maximum likelihood estimation substitutes the unbounded notion of likelihood for the bounded definition of probability (Casella and Berger 2002, p.316; Fisher 1922, p.327; King 1989, p.23). This is an important theoretical distinction, but of little significance in applied practice. If we regard $p(\mathbf{X}|\boldsymbol{\theta})$ as a function of $\boldsymbol{\theta}$ for some given observed data $\mathbf{X}$, then $L(\boldsymbol{\theta}|\mathbf{X}) = \prod_{i=1}^{n} p(\mathbf{X}|\boldsymbol{\theta})$ (DeGroot 1986, p.339).

Typically it is mathematically more convenient to work with the natural log of the likelihood function. This does not change any of the resulting parameter estimates because the likelihood function and the log likelihood function have identical modal points for commonly used forms. Using a PDF for a single parameter of interest, the basic *log* likelihood function is very simple:

$$\ell(\boldsymbol{\theta}|\mathbf{X}) = \log(L(\boldsymbol{\theta}|\mathbf{X})), \tag{2.1}$$

where we use $\ell(\boldsymbol{\theta}|\mathbf{X})$ as shorthand to distinguish the log likelihood function from the likelihood function, $L(\boldsymbol{\theta}|\mathbf{X})$.

The score function is the first derivative of the log likelihood function with respect to the parameters of interest:

$$\dot{\ell}(\boldsymbol{\theta}|\mathbf{X}) = \frac{\partial}{\partial \boldsymbol{\theta}}\ell(\boldsymbol{\theta}|\mathbf{X}). \tag{2.2}$$

Setting $\dot{\ell}(\boldsymbol{\theta}|\mathbf{X})$ equal to zero and solving gives the maximum likelihood estimate, $\hat{\boldsymbol{\theta}}$. This is now the "most likely" value of $\boldsymbol{\theta}$ from the parameter space Θ treating the observed data as given: $\hat{\boldsymbol{\theta}}$ maximizes the likelihood function at the observed values. The *Likelihood Principle* (Birnbaum 1962) states that once the data are observed, and therefore treated as given, all of the available evidence for estimating $\boldsymbol{\theta}$ is contained in the (log) likelihood function, $\ell(\boldsymbol{\theta}|\mathbf{X})$. This is a handy data reduction tool because it tells us exactly what treatment of the data is important to us and allows us to ignore an infinite number of alternates (Poirer 1988, p.127). The key difference between the classic likelihood approach and the Bayesian inference is that more information is used in the analysis and more information is provided through descriptions of the posterior beyond modal summaries. Thus the likelihood principle only has relevance here for part of the Bayesian model.

The maximum likelihood doctrine states that an admissible $\boldsymbol{\theta}$ that maximizes the likelihood function probability (discrete case) or density (continuous case), relative to alternative

values of $\boldsymbol{\theta}$, provides the $\boldsymbol{\theta}$ that is "most likely" to have generated the observed data, $\mathbf{X}$, given the assumed parametric form. Restated, if $\hat{\boldsymbol{\theta}}$ is the maximum likelihood estimator for the unknown parameter vector, then it is necessarily true that $L(\hat{\boldsymbol{\theta}}|\mathbf{X}) \geq L(\boldsymbol{\theta}|\mathbf{X}) \; \forall \; \boldsymbol{\theta} \in \boldsymbol{\Theta}$, where $\boldsymbol{\Theta}$ is the admissible set of $\boldsymbol{\theta}$. Admissible here means values of $\boldsymbol{\theta}$ are taken from the valid parameter space ($\boldsymbol{\Theta}$): values of $\boldsymbol{\theta}$ that are unreasonable according to the form of the sampling distribution of $\boldsymbol{\theta}$ are not considered (integrated over).

Setting the score function from the joint PDF or PMF equal to zero and rearranging gives the likelihood equation:

$$\sum t(X_i) = n \frac{\partial}{\partial \boldsymbol{\theta}} E[\mathbf{X}] \tag{2.3}$$

where $\sum t(X_i)$ is the remaining function of the data, depending on the form of the probability density function (PDF) or probability mass function (PMF), and $E[\mathbf{X}]$ is the expectation over the kernel of the density function for $\mathbf{X}$. The kernel of a PDF or PMF is the component of the parametric expression that directly depends on the form of the random variable, i.e., what is left when normalizing constants are omitted. We can often work with kernels of distributions for convenience and recover all probabilistic information at the last stage of analysis by renormalizing (ensuring summation or integration to one). The kernel is the component of the distribution that assigns *relative* probabilities to levels of the random variable (see Gill 2000, Chapter 2). For example the kernel of a gamma distribution is just the part $x^{\alpha-1}\exp[-x\beta]$, without the normalizing constant $\beta^{\alpha}/\Gamma(\alpha)$.

The underlying theory here is remarkably strong. Solving (2.3) for the unknown coefficient produces an estimator that is unique (due to a unimodal posterior distribution), consistent (converges in probability to the population value), and asymptotically unbiased, but not necessarily unbiased in finite sample situations. On the latter point, the maximum likelihood estimate for the variance of a normal model, $\hat{\sigma}^2 = \frac{1}{n}\sum(x_i - \bar{x})^2$ is biased by $n/(n-1)$. This difference is rarely of significance and clearly the bias disappears in the limit, but it does illustrate that unbiasedness of the maximum likelihood estimate is guaranteed only in asymptotic circumstances. It is also asymptotically efficient (the variance of the estimator achieves the lowest possible value as the sample size becomes adequately large: the Cramér-Rao lower bound, see Shao 2005). This result combined with the central limit theorem gives the asymptotic normal form for the estimator: $\sqrt{n}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \xrightarrow{\mathcal{P}} \mathcal{N}(\mathbf{0}, \Sigma_{\boldsymbol{\theta}})$. This means that as the sample size gets large, the difference between the estimated value of $\boldsymbol{\theta}$ and the true value of $\boldsymbol{\theta}$ gets progressively close to zero, with a variance governed by $\frac{1}{\sqrt{n}}\Sigma_{\boldsymbol{\theta}}$, where $\Sigma_{\boldsymbol{\theta}}$ is the $k \times k$ variance-covariance matrix for $\boldsymbol{\theta}$. Furthermore, $\sum t(x_i)$ is a sufficient statistic for $\boldsymbol{\theta}$, meaning that all of the relevant information about $\boldsymbol{\theta}$ in the data is contained in $\sum t(x_i)$. For example, the normal log likelihood expressed as a joint exponential family form is $\ell(\boldsymbol{\theta}|\mathbf{X}) = \left(\mu \sum x_i - \frac{n\mu^2}{2}\right)/\sigma^2 - \frac{1}{2\sigma^2}\sum x_i^2 - \frac{n}{2}\log(2\pi\sigma^2)$. So $t(\mathbf{X}) = \sum X_i$, $\frac{d}{d\mu}\frac{n\mu^2}{2} = n\mu$, and equating gives the maximum likelihood estimate of μ to be the sample average that we know from basic texts: $\frac{1}{n}\sum x_i$. Bayesian inference builds upon this strong foundation by combining likelihood information, as just described, with prior information in a way describes all unknown quantities distributionally.

2.3 The Basic Bayesian Framework

Our real interest lies in obtaining the distribution of the unknown k-dimensional $\boldsymbol{\theta}$ coefficient vector, given an observed $\mathbf{X}$ matrix of data values: $p(\boldsymbol{\theta}|\mathbf{X})$. If we choose to here, we can still determine the "most likely" values of the $\boldsymbol{\theta}$ vector using the k-dimensional posterior mode or mean, but it is better to more fully describe the shape of the posterior distribution, given by Bayes' Law:

$$p(\boldsymbol{\theta}|\mathbf{X}) = p(\mathbf{X}|\boldsymbol{\theta})\frac{p(\boldsymbol{\theta})}{p(\mathbf{X})} \tag{2.4}$$

where $p(\mathbf{X}|\boldsymbol{\theta})$ is the joint probability function for data (the *probability of the sample* for a fixed $\boldsymbol{\theta}$) under the assumption that the data are independent and identically distributed according to $p(X_i|\boldsymbol{\theta})\ \forall\ i = 1, \ldots, n$, and $p(\boldsymbol{\theta})$, $p(\mathbf{X})$ are the corresponding unconditional probabilities. This is mechanically correct but it does not fully represent Bayesian thinking or notation about the inference process.

2.3.1 Developing the Bayesian Inference Engine

From the Bayesian perspective, there are only two fundamental types of quantities: known and unknown. The goal is to use the known quantities along with a specified parametric expression to make inferential statements about the unknown quantities. The definition of such unknown quantities is very general; they can be any missing data or unknown parameters. When quantities are observed, they are considered fixed and conditioned upon. Suppose we fully observe the data $\mathbf{X}$. This is now a fixed and given quantity in the inferential process. The first implication is that $p(\mathbf{X}|\boldsymbol{\theta})$ in (2.4) does not make notational sense since the known quantity is conditional on the unknown quantity. Instead label this quantity as $L(\boldsymbol{\theta}|\mathbf{X})$ and treat it as a likelihood function. It is a likelihood function of course, but note that the justification is inherently Bayesian (i.e., probabilistic). Also, since the $\mathbf{X}$ are treated as fixed, $p(\mathbf{X})$ is not especially useful here. However, this quantity performs an important role in model comparison as we shall see in Chapter 7.

The prior distribution, $p(\boldsymbol{\theta})$, must be specified, but need not be highly influential. This is simply a distributional statement about the unknown parameter vector $\boldsymbol{\theta}$, *before* observing or conditioning on the data. Much controversy has developed about the nature of prior distributions and we will look at alternative forms in detail in Chapter 4. It is *essential* to supply a prior distribution in Bayesian models and well over 100 years of futile searching for a way to avoid doing so have clearly demonstrated this. Currently an approach called *objective Bayes* (O-Bayes) seeks to mathematically minimize the effect of prior specifications.

Start with the form of Bayes' Law defined with conditional probability, giving the posterior of interest:

$$\pi(\boldsymbol{\theta}|\mathbf{X}) = \frac{p(\boldsymbol{\theta})L(\boldsymbol{\theta}|\mathbf{X})}{p(\mathbf{X})}, \tag{2.5}$$

which is an update of (2.4) that gives the desired probability statement on the left-hand side now using the $\pi()$ notation as a reminder. This states that the distribution of the unknown parameter conditioned on the observed data is equal to the product of the prior distribution assigned to the parameter and the likelihood function, divided by the unconditional probability of the data. The form of (2.5) can also be expressed as:

$$\pi(\boldsymbol{\theta}|\mathbf{X}) = \frac{p(\boldsymbol{\theta})L(\boldsymbol{\theta}|\mathbf{X})}{\int_{\boldsymbol{\Theta}} p(\boldsymbol{\theta})L(\boldsymbol{\theta}|\mathbf{X})d\boldsymbol{\theta}}, \tag{2.6}$$

where $\int_{\Theta} p(\boldsymbol{\theta})L(\boldsymbol{\theta}|\mathbf{X})d\boldsymbol{\theta}$ is an expression for $p(\mathbf{X})$ explicitly integrating the numerator over the support of $\boldsymbol{\theta}$. This term has several names in the literature: the *normalizing constant*, the *normalizing factor*, the *marginal likelihood*, and the *prior predictive distribution*, although it is actually the marginal distribution of the data, and it ensures that $\pi(\theta|\mathbf{X})$ integrates to one as required by the definition of a probability function. A more compact and succinct form of (2.6) is developed by dropping the denominator and using proportional notation since $p(\mathbf{X})$ does not depend on θ and therefore provides no relative inferential information about more or less likely values of θ:

$$\pi(\theta|\mathbf{X}) \propto p(\theta)L(\theta|\mathbf{X}), \tag{2.7}$$

meaning that the unnormalized *posterior* (sampling) distribution of the parameter of interest is proportional to the prior distribution times the likelihood function:

$$\text{Posterior Probability} \propto \text{Prior Probability} \times \text{Likelihood Function}.$$

It is typically (but not always, see later chapters) easy to renormalize the posterior distribution as the last stage of the analysis to return to (2.6).

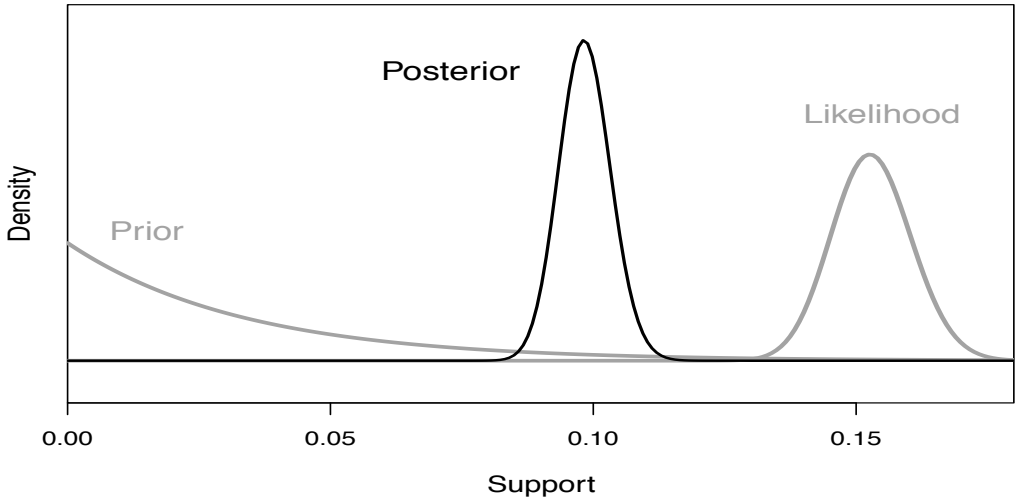


FIGURE 2.1: Posterior $\propto$ Prior $\times$ Likelihood

As an illustration, suppose we have data that are iid exponentially distributed $f(X|\theta) = \theta e^{-\theta X}$, $X, \theta \in (0, \infty)$, and an exponential prior distribution for the unknown parameter $p(\theta) = \beta e^{-\theta\beta}$, $\beta \in (0, \infty)$, where $\beta = 30$ here as an arbitrary modeling choice. These data are actually taken from Example 2.3.2.1 on page 44 below but the specific data context is not yet important here. The exponential assumption for the data means that the likelihood function is a gamma distribution with parameters $n + 1$ and $\sum X_i$: $L(\theta) \propto \theta^n e^{-\theta \sum X_i}$. Multiplying the likelihood and the prior produces a another gamma distribution with new parameters $n+1$ and $\sum X_i + \beta$ (Exercise 2 in this chapter). This is illustrated in Figure 2.1 where we see that the prior distribution on the left pulls the likelihood function towards it in the creation of the posterior distribution. This is called *shrinkage* in the Bayesian literature and it means that the posterior mean "shrinks" towards the prior mean. Figure 2.1 is purposefully over-dramatic in showing this effect, but movement such as this is a characteristic of all Bayesian models: the posterior distribution is always a compromise between the prior distribution and the likelihood function. The question is how influential is the prior distribution in this calculation.

We can also state Bayes' Law in odds form as done in (1.6) on page 11. Suppose we have two competing models expressed by $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, which are considered to exhaust the possible states of nature. This latter assumption may be unrealistic, but it is often the case that a researcher will consider only two alternatives at a time. If we now observe the data, $\mathbf{X}$, then Bayes' Law in odds form is:

$$\begin{aligned}\frac{\pi(\boldsymbol{\theta}_1|\mathbf{X})}{\pi(\boldsymbol{\theta}_2|\mathbf{X})} &= \frac{\frac{p(\boldsymbol{\theta}_1)}{p(\mathbf{X})} L(\boldsymbol{\theta}_1|\mathbf{X})}{\frac{p(\boldsymbol{\theta}_2)}{p(\mathbf{X})} L(\boldsymbol{\theta}_2|\mathbf{X})} \\ &= \frac{p(\boldsymbol{\theta}_1)}{p(\boldsymbol{\theta}_2)} \frac{L(\boldsymbol{\theta}_1|\mathbf{X})}{L(\boldsymbol{\theta}_2|\mathbf{X})}\end{aligned}$$

$$\text{Posterior Odds} = \text{Prior Odds} \times \text{Likelihood Ratio}.$$

This likelihood ratio will later be generalized in Chapter 7 as the Bayes Factor. Furthermore, if we assume equal prior probabilities, the posterior odds is simply equal to the likelihood ratio. Since likelihood ratio testing is a very popular tool in non-Bayesian model comparison, this is a nice linkage: under basic circumstances Bayesian posterior odds comparison is equivalent to simple likelihood ratio testing.

2.3.2 Summarizing Posterior Distributions with Intervals

In Chapter 1, we noted the value of describing posterior distributions with simple quantiles, and calculated analytical posterior moments: $E[\boldsymbol{\theta}|\mathbf{X}]$, and $\text{Var}[\boldsymbol{\theta}|\mathbf{X}]$. However, such summaries may miss distributional features and should be complemented with additional interval-based measures.

The first descriptive improvement here is found by moving from confidence intervals to Bayesian credible intervals. Recall that confidence intervals are intimately tied with

frequentist (not likelihoodist!) theory since a $100(1-\alpha)\%$ confidence interval covers the *true* underlying parameter value across $1-\alpha$ proportion of the replications in the experiment, *on average*. So confidence is a property of frequentist replication from a large number of repeated iid samples and underlying parameters that are fixed immemorial. In fact, the confidence interval may be considered *the most* frequentist summary possible since it does not have an interpretation without multiple replications of the exact same experiment. One major problem with the confidence interval lies in its interpretation. Most consumers of statistics *want* confidence intervals to be probabilistic statements about some region of the parameter space, but careful writers discourage this by explaining the actual nature of confidence intervals: "a 95% confidence interval covers the true value of the parameter in nineteen out of twenty trials on average." In most social science settings with observational data it is not practical to repeat some experiment nineteen more times with an assumed iid data-generating source.

2.3.2.1 Bayesian Credible Intervals

The Bayesian analogue to the confidence interval is the credible interval and more generally the credible set, which does not have to be contiguous. Most of the time in practice it is calculated in *exactly the same way* as the confidence interval for unimodal symmetric forms. For instance calculating a 95% credible interval under the Gaussian normal assumption means marching out 1.96 standard errors from the mean in either direction, just like the analogous confidence interval is created. However, for asymmetric distributions this algorithm would produce a credible interval with unequal tails and incorrect coverage.

The difference between confidence intervals and Bayesian credible intervals is in the interpretation of what the interval means. A $100(1-\alpha)\%$ credible interval gives the region of the parameter space where the probability of covering $\boldsymbol{\theta}$ is equal to $1-\alpha$ (it may actually be a little more than $1-\alpha$ for discrete parameter spaces in order to guarantee at least this level of coverage). In contrast, applying this new definition to the *confidence interval* means that the probability of coverage is either zero or one, since it either covers the true $\boldsymbol{\theta}$ or it doesn't.

Formally, an equal tail credible set for the posterior distribution is defined as follows. Define C as a subset of the parameter space, $\boldsymbol{\Theta}$, such that a $100(1-\alpha)\%$ credible interval meets the condition:

$$1-\alpha = \int_C \pi(\boldsymbol{\theta}|\mathbf{X})d\theta \tag{2.8}$$

(this is summation instead of an integration for discrete parameter spaces, but we will discuss mostly continuous parameter spaces here). It is important to note that credible intervals are not unique. That is, we can easily define C in different ways to cover varying parts of $\boldsymbol{\Theta}$ and still meet the probabilistic condition in (2.8). It is not necessary that we center these intervals at a mean or mode. Important differences arise in asymmetric and multimodal distributions, and the convention is to create *equal tail intervals*: no matter what the shape of the posterior distribution. This means that the $100(1-\alpha)\%$ credible

interval is created such that $\alpha/2$ of the density is put in both the left and right tails outside of the designated credible interval.

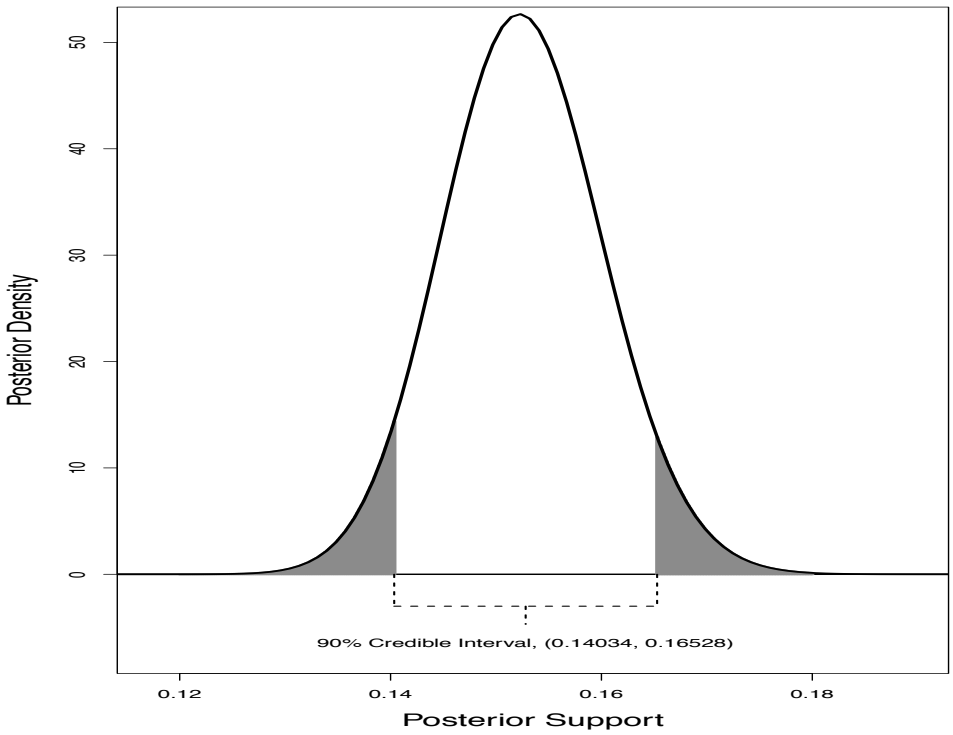


FIGURE 2.2: CI for State Duration Time to Adaptation, 1998-2005

■ **Example 2.1: Credible Interval, Fifty U.S. States Time to Adoption for Health Bills.** Boehmke (2009) counts bills passed in the fifty states between 1998 and 2005 that contain policy implications for the increasing obesity rates in the U.S. These include limits on sugary drinks at schools, requiring insurers to cover particular medical procedures, as well as limitations on lawsuits from consumer groups on the fast food industry. We define duration data, $\mathbf{X}$, to be the time in years through this period for a bill to be passed. Assume that $\mathbf{X}$ is exponentially distributed $p(X|\theta) = \theta e^{-\theta X}$ defined over $[0, \infty)$, where interest is in the posterior distribution of the unknown parameter θ. Specify the prior distribution as $p(\theta) = 1/\theta$, for $\theta \in [0{:}\infty)$. The resulting posterior is given by:

$$\pi(\theta|\mathbf{X}) \propto p(\theta)L(\theta|\mathbf{X}) = \left(\frac{1}{\theta}\right)\theta^n \exp\left[-\theta\sum_{i=1}^{n} x_i\right] = \theta^{n-1}\exp\left[-\theta\sum_{i=1}^{n} x_i\right] \qquad (2.9)$$

(note that we are using the proportionality shortcut from (2.7)). If we stare at this for a few moments we can see that $\theta|\mathbf{X} \sim \mathcal{G}(\theta|n, \sum x_i)$ with the "rate" specification for the second parameter. Putting the constants back in front to recover the full form

of this gamma posterior distribution produces:

$$\pi(\theta|\mathbf{X}) = \frac{(\sum x_i)^n}{\Gamma(n)}\theta^{n-1}\exp\left[-\theta\sum x_i\right]$$

(details in Appendix B). As stated, once we produce the posterior distribution, we know everything about the distribution of θ and can convey to our readers any summary we would like.

TABLE 2.1: State Duration Time to Adaptation, 1998-2005

State	N	Mean Duration	State	N	Mean Duration	State	N	Mean Duration
AL	2	7.500	LA	14	5.571	OK	12	6.583
AK	12	6.667	ME	2	5.500	OH	0	NaN
AZ	12	6.250	MD	11	6.455	OR	1	8.000
AR	6	6.167	MA	7	7.143	PA	12	7.083
CA	46	6.000	MI	4	7.000	RI	7	7.000
CO	11	6.636	MN	2	7.000	SC	6	6.333
CT	2	7.000	MS	7	7.143	SD	1	7.000
DE	4	7.000	MO	18	5.556	TN	17	7.235
FL	11	6.364	MT	2	7.000	TX	16	6.250
GA	7	5.857	NE	5	7.400	UT	3	7.667
HI	8	6.375	NV	4	8.000	VT	8	6.625
ID	6	6.000	NH	1	5.000	VA	15	6.533
IL	4	6.750	NJ	6	7.333	WA	12	6.083
IN	31	7.065	NM	6	6.500	WV	2	7.500
IA	3	5.000	NY	9	6.556	WI	4	7.750
KS	4	8.000	NC	8	7.250	WY	1	8.000
KY	4	7.500	ND	9	6.111			

The complete data are given in Table 2.1 for annualized periods, as well as in the `R` package `BaM`. Note the "`NaN`" value for the Ohio mean duration given by `R` since there is nothing to average. We will leave this case out of the subsequent analysis since the time to adoption is infinity, or more realistically, censored from us. The state averages from the third column of the table are weighted by N in the second column to reflect the number of such events: $\mathbf{X}_i N_i$. Since the sufficient statistic in the posterior distribution is a sum, there is no loss of information from not having the full original data from the authors (sums of means times n equal the total sum). The end-points of the equal tail credible interval are created by solving for the limits (L and H) in the two integrals:

$$\frac{\alpha}{2} = \int_0^L \pi(\boldsymbol{\theta}|\mathbf{X})d\theta \qquad \frac{\alpha}{2} = \int_H^\infty \pi(\boldsymbol{\theta}|\mathbf{X})d\theta \tag{2.10}$$

or, more simply, we could use basic `R` functions to manipulate the `state.df` dataframe containing the data in the table above:

```
state.df <- state.df[-35,] # REMOVES OHIO
qgamma(0.05,shape=sum(state.df$N),rate=sum(state.df$N*state.df$dur))
```

```
[1] 0.14034
qgamma(0.95,shape=sum(state.df$N),rate=sum(state.df$N*state.df$dur))
[1] 0.16528
```

for a 90% credible interval. These points and a plot of the posterior distribution are given in Figure 2.2. The slight asymmetry of this gamma distribution means that the left tail region needs to reach higher (moving the boundary to the right) in order to equal the right tail in total posterior density. Therefore the density values (y-axis) at endpoints differ: 14.384 versus 12.898. To contrast with these results, the maximum likelihood value is $\hat{\theta} = 0.018$, (the inverse of the weighted mean of the data, Casella and Berger [2002]), whereas the posterior mean $E_\pi[\theta|\mathbf{X}] = 0.153$, showing that the prior $p(\theta) = 1/\theta$ has some influence.

2.3.2.2 Bayesian Highest Posterior Density Intervals

Credible intervals are common and useful, but a theoretically more defensible interval can be produced by incorporating some additional flexibility. When looking at posterior distributions, we really care where the highest density exists on the support of the posterior density, regardless of whether it is contiguous or not. So the big idea behind highest posterior density (HPD) regions is that no region outside of the interval will have higher posterior density than any region inside the HPD region. Hence for multimodal distributions the HPD region may actually be a set of individually non-contiguous intervals. See Hyndman (1996) for interesting forms as well as a general introduction to HPD regions. The use of the word "intervals" is common instead of "regions," but HPD regions possess an automatic ability to be non-contiguous so the latter is more correct. For symmetric unimodal forms the HPD interval will be contiguous and identical to an equal tail credible interval.

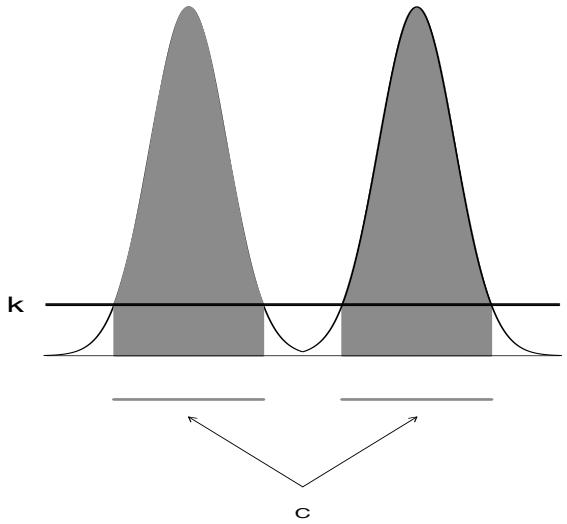


FIGURE 2.3: Bimodal Distribution Highest Posterior Density Interval

More specifically, a $100(1-\alpha)\%$ highest posterior density region is the subset of the support of the posterior distribution for some parameter, θ, that meets the criteria:

$$C = \{\theta : \pi(\theta|\mathbf{x}) \geq k\},$$

where k is the largest number such that:

$$1-\alpha = \int_{\theta:\pi(\theta|\mathbf{x})>k} \pi(\theta|\mathbf{x})d\theta \tag{2.11}$$

(Casella and Berger 2002, p.448). This is the $1-\alpha$ proportion subset of the sample space of θ where the posterior density of θ is maximized. So k is a horizontal line that slices across the density producing inside HPD areas and outside HPD areas. This will be a regular interval if the posterior distribution is unimodal, and it may be a discontiguous region if the posterior distribution is multimodal. This is shown in Figure 2.3 where it is clear that a bimodal distribution having a deep trough in the middle produces a non-contiguous HPD region emphasizing the undesirability of incorporating the middle region. Multimodal forms appear in Bayesian mixture models, and an example appears in Chapter 3 starting on page 87.

Chen and Shao (1999) provide another way to conceptualize the HPD region. For a unimodal posterior form, given by $\pi(\theta|\mathbf{X})$, our objective is to find the values $[\theta_L, \theta_U]$ that define a $(1-\alpha)$ HPD region. It turns out that the answer to this question is given by also using cumulative (Π) differences:

$$\min_{\theta_L<\theta_U} \left[\underbrace{|\pi(\theta_U)-\pi(\theta_L)|}_{\text{difference in "height"}} + \underbrace{|\Pi(\theta_U)-\Pi(\theta_L)-(1-\alpha)|}_{\text{difference in "width"}} \right]. \tag{2.12}$$

So the first difference lines up k across the two HPD region endpoints and the second difference gives the coverage probability. In many circumstances this minimization gives zero, but for posteriors with flat regions it would need to be modified with some additional criteria to provide a unique interval such as picking the one with smallest θ_L value.

■ **Example 2.2: HPD Region, Fifty U.S. States Time to Adoption for Health Bills.** Returning to the example from time to adopt health laws, Figure 2.4, shows the HPD region for this posterior along with the determining line at $k = 16.873$. Notice in comparing the HPD region to the credible interval for this model, that the HPD region has equal height at the end-points at $k = 13.602$, and that the endpoints of the interval differ slightly from the corresponding equal-tail credible interval. The HPD region is constructed in a very simple way by starting at the posterior mode, then incrementing a horizontal line down vertically until the separation between the higher density and lower density regions reflects the desired coverage. So for each value of k, the level on the y-axis, we separately sum the area inside and outside the coverage area, regardless of contiguity. The **Computational Addendum** at the end of this chapter provides the `R` code used here for a posterior gamma distributed form. This process was quite easy to implement in `R` since we know the exact form of the gamma distribution from the model. Later when estimating marginal posterior forms with Bayesian stochastic simulation (MCMC), we will see that there are similarly easy ways to make this calculation even when we do not have an exact parametric description of

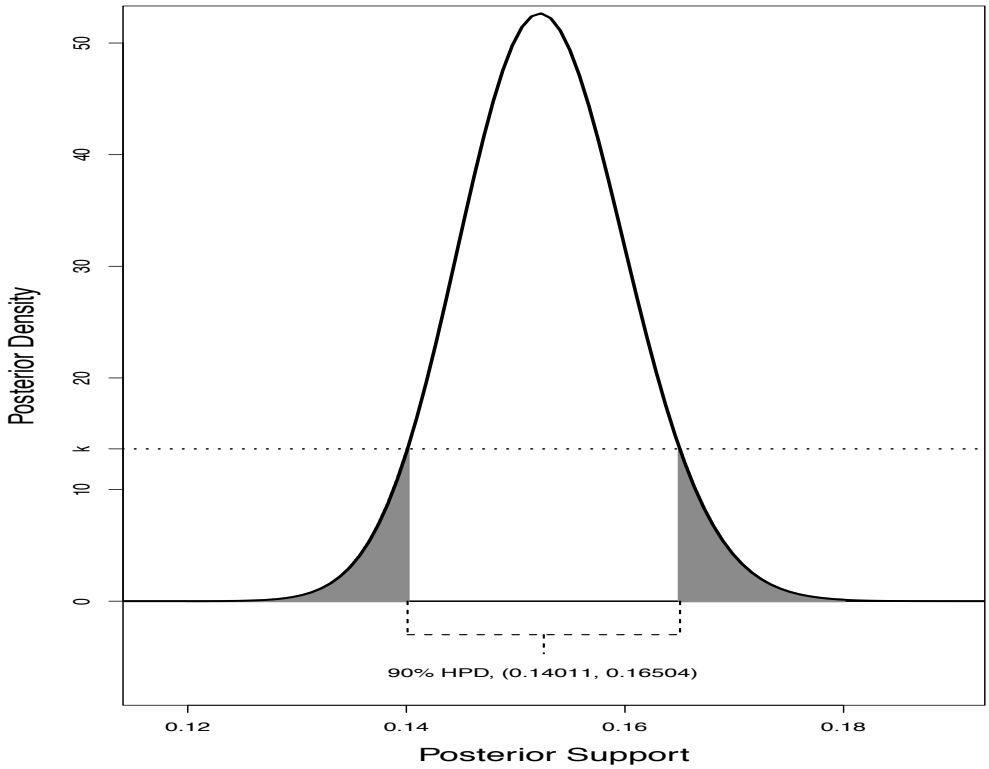


FIGURE 2.4: HPD Region, State Duration Time to Adaptation, 1998-2005

the posterior distribution. There is also the `HPDinterval` function in the `CODA` library for this, but it does not illustrate the underlying theory as directly as the exposition here.

2.3.3 Quantile Posterior Summaries

Often interval results are given basic quantile summaries without explicitly labeling these as credible intervals or HPD regions. Of course quantile summaries for unimodal forms are credible interval definitions so this is an analogous procedure. However, for highly multimodal forms basic quantile summaries can be misleading in the same way that boxplots hide such characteristics. Frequently summary tables for regression models contain quantile summaries since they are efficient with printed space and reveal characteristics of the distribution of interest.

■ **Example 2.3: Quantiles, Fifty U.S. States Time to Adoption for Health Bills.** Using the data in Example 2.3.2.1 (starting on page 44), we can also calculate quantiles in addition to the credible intervals done before. We know that the

time to adopt health laws example has a unimodal posterior, so simple quantiles are applicable. Consider the simple R commands:

```
q.vals <- c(0.025,0.05,0.25,0.5,0.75,0.95,0.975)
rbind( quant.vals, "quantiles"=
   qgamma(quant.vals,shape=sum(state.df$N),
          rate=sum(state.df$N*state.df$dur)) )
              [,1]    [,2]    [,3]    [,4]    [,5]    [,6]    [,7]
quant.vals 0.0250 0.05000 0.25000 0.50000 0.75000 0.95000 0.97500
quantiles  0.1381 0.14034 0.14742 0.15247 0.15764 0.16528 0.16781
```

using the `state.df` dataframe described above. This gives quantiles for common interval definitions ($\alpha = 0.05$ and $\alpha = 0.10$), the interquartile range (IQR), and the median). Note that the values $(0.14034, 0.16528)$ were given before with the 90% credible interval.

2.3.4 Beta-Binomial Model

This model illustrates the development of a posterior distribution for an interesting implication and shows how the properties of this posterior distribution can be described in conventional terms. Let $X_1, X_2, \ldots, X_n$ be independent random variables, all produced by the same probability mass function (iid): $\mathcal{BR}(p)$, and place a $\mathcal{BE}(A, B)$ prior distribution on the unknown population probability, p (see Appendix B for details on these forms). The goal is to get a posterior distribution for p, and this process is greatly simplified by noting that the sum of n Bernoulli(p) random variables is distributed binomial(n, p). Define a new variable: $Y = \sum_{i=1}^{n} X_i$. The joint distribution of Y and p is the product of the conditional distribution of Y and the marginal (prior) distribution of p:

$$\begin{aligned} f(y,p) &= f(y|p)f(p) \\ &= \left[\binom{n}{y} p^y (1-p)^{n-y}\right] \quad \times \quad \left[\frac{\Gamma(A+B)}{\Gamma(A)\Gamma(B)} p^{A-1}(1-p)^{B-1}\right] \\ &= \frac{\Gamma(n+1)\Gamma(A+B)}{\Gamma(y+1)\Gamma(n-y+1)\Gamma(A)\Gamma(B)} p^{y+A-1}(1-p)^{n-y+B-1}. \end{aligned} \tag{2.13}$$

The marginal distribution of Y is easy to calculate by integrating (2.13) with respect to p using a standard trick:

$$\begin{aligned} f(y) &= \int_0^1 \frac{\Gamma(n+1)\Gamma(A+B)}{\Gamma(y+1)\Gamma(n-y+1)\Gamma(A)\Gamma(B)} p^{y+A-1}(1-p)^{n-y+B-1} dp \\ &= \int_0^1 \frac{\Gamma(n+1)\Gamma(A+B)}{\Gamma(y+1)\Gamma(n-y+1)\Gamma(A)\Gamma(B)} \frac{\Gamma(y+A)\Gamma(n-y+B)}{\Gamma(n+A+B)} \\ &\qquad \times \frac{\Gamma(n+A+B)}{\Gamma(y+A)\Gamma(n-y+B)} p^{y+A-1}(1-p)^{n-y+B-1} dp \end{aligned}$$

$$= \frac{\Gamma(n+1)\Gamma(A+B)}{\Gamma(y+1)\Gamma(n-y+1)\Gamma(A)\Gamma(B)} \frac{\Gamma(y+A)\Gamma(n-y+B)}{\Gamma(n+A+B)}$$

$$\times \underbrace{\int_0^1 \frac{\Gamma(n+A+B)}{\Gamma(y+A)\Gamma(n-y+B)} p^{y+A-1}(1-p)^{n-y+B-1} dp}_{\text{equal to one}}$$

$$= \frac{\Gamma(n+1)\Gamma(A+B)}{\Gamma(y+1)\Gamma(n-y+1)\Gamma(A)\Gamma(B)} \frac{\Gamma(y+A)\Gamma(n-y+B)}{\Gamma(n+A+B)}. \tag{2.14}$$

The trick here is rearranging the terms such that a complete beta distribution with differing parameters is integrated across the support of the unknown random variable. The probability density function given in the last line of (2.14) is called (not surprisingly) the beta-binomial. Obtaining the posterior distribution of p is now a simple application of the definition of conditional probability:

$$f(p|y) = \frac{f(y,p)}{f(y)}$$

$$= \left(\frac{\Gamma(n+1)\Gamma(A+B)}{\Gamma(y+1)\Gamma(n-y+1)\Gamma(A)\Gamma(B)} p^{y+A-1}(1-p)^{n-y+B-1} \right) \Bigg/$$

$$\left(\frac{\Gamma(n+1)\Gamma(A+B)}{\Gamma(y+1)\Gamma(n-y+1)\Gamma(A)\Gamma(B)} \frac{\Gamma(y+A)\Gamma(n-y+B)}{\Gamma(n+A+B)} \right)$$

$$= \frac{\Gamma(n+A+B)}{\Gamma(y+A)\Gamma(n-y+B)} p^{(y+A)-1}(1-p)^{(n-y+B)-1}. \tag{2.15}$$

This can easily be seen as a new beta distribution with parameters $A' = y + A$ and $B' = n - y + B$. While it would be more typical of a Bayesian to describe this posterior distribution with quantiles, credible sets or highest posterior density intervals, we can get a point estimate of p here by taking the mean of this beta distribution:

$$\hat{p} = \frac{y+A}{A+B+n}. \tag{2.16}$$

Rearrange (2.16) algebraically to produce:

$$\hat{p} = \left[\frac{n}{A+B+n} \right] \left(\frac{y}{n} \right) + \left[\frac{A+B}{A+B+n} \right] \left(\frac{A}{A+B} \right), \tag{2.17}$$

which is the weighted combination of the sample mean from the binomial and the mean of the prior beta distribution where the weights are determined by the beta parameters, A and B, along with the sample size n. Holding A and B constant at reasonable values and increasing the sample size places more weight on the y/n term since the weight for the beta mean, $\frac{A}{A+B}$, has n only in the denominator and the weights necessarily add to one. This turns out to be theoretically more interesting than it would first appear and it highlights an important and desirable property of Bayesian data analysis: as the sample size increases, the likelihood function, $f(y|p)$, is iteratively updated to incorporate this new information

and eventually subsumes the choice of prior, $f(p)$, because of sample size. Conversely, when the sample size is very small it makes sense to rely upon reliable prior information if it exists.

This hierarchical parameterization of the binomial with a *random effects component* (the name commonly used in non-Bayesian settings) is often done when there is evidence of overdispersion in the data: the variance exceeds that of the binomial: $np(1-p)$ (see Agresti 2002, p.151, Lehmann and Casella 1998, p.230, Carlin and Louis 2001, p.44, McCullagh and Nelder 1989, p.140). When the posterior has the same distributional family as the prior, as in this case, we say that the prior and the likelihood distributions are *conjugate*. This is an attractive property since it not only assures that there is a closed form for the prior, it means that it is also easy to calculate. Conjugate priors are discussed in detail in Chapter 4, Section 4.3, and Appendix B lists conjugate relationships, if they exist for commonly used distributions.

■ **Example 2.4: A Cultural Consensus Model in Anthropology.** Romney (1999) looks at the level of consensus among 24 Guatemalan women on whether or not 27 diseases known to all respondents are contagious. The premise is that a high level of consensus about something as important as the spread of diseases indicates to what extent knowledge is a component of culture in this setting. The survey data for polio are given by a vector, x, containing:

1	1	1	1	0	1	1	0	1	0	1	1	1	0	1	1	1	1	1	1	0	0	0	1

where 1 indicates that the respondent believes polio to be noncontagious and 0 indicates that the respondent believes polio to be contagious (Romney ranks it 13 out of 27 on a contagious scale). Here we apply the beta-binomial model with two different beta priors exactly in the manner discussed above. The first prior is a $\mathcal{BE}(p|15,2)$ prior that imposes a great deal of prior knowledge about the unknown p parameter. The second prior is a $\mathcal{BE}(p|1,1)$ prior that is actually a uniform prior over $[0:1]$ indicating a great deal of uncertainty about the location of p. Because the beta distribution is conjugate to the binomial the resulting posterior distributions are also beta and we obtain $\mathcal{BE}(\sum x_i + 15, n - \sum x_i + 2) = \mathcal{BE}(32,9)$, and $\mathcal{BE}(\sum x_i + 1, n - \sum x_i + 1) = \mathcal{BE}(18,8)$, respectively. Notice that there is almost no work to be done here (e.g. just plugging-in the specified values) since we have already worked out the analytical form of the posterior distribution. These posteriors along with the specified priors are shown in Figure 2.5 where the posterior is illustrated with the darker line and 95% HPD intervals are shown.

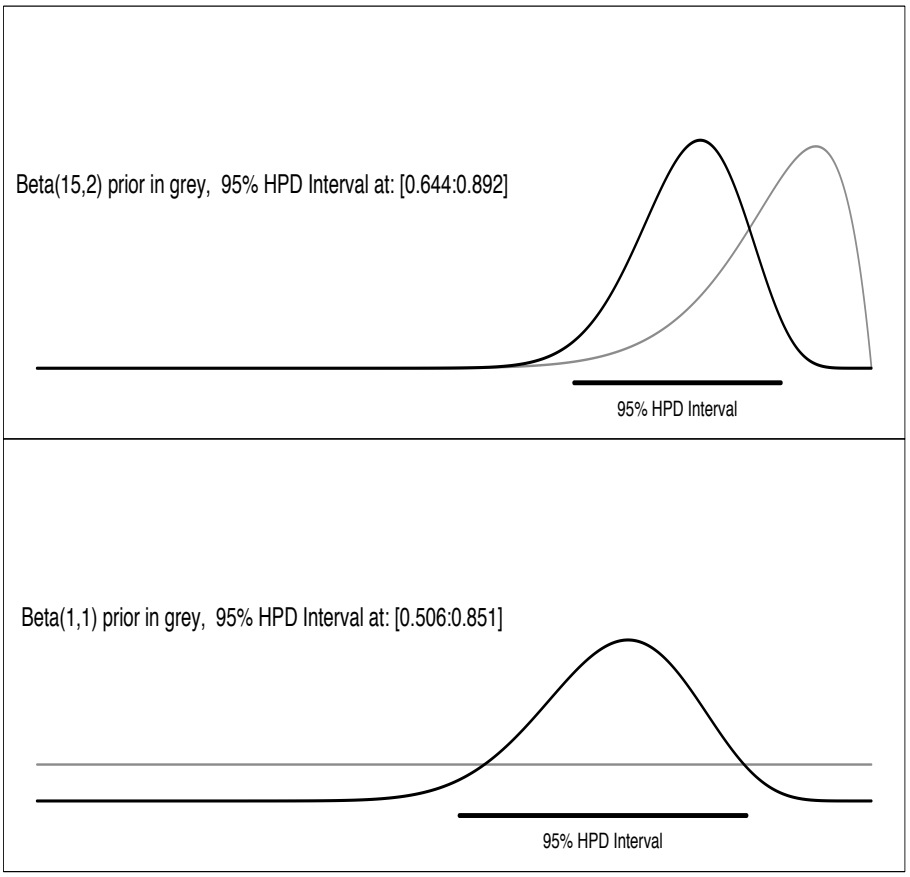


FIGURE 2.5: Prior and Posterior Distributions in the Beta-Binomial Model

It is clear from this figure that even though the prior specifications are very distinct, the resulting posteriors differ only modestly as evidenced by the highest posterior density regions indicated below the distributions. The uniform prior clearly pulls the posterior density to the left in contrast to the $\mathcal{BE}(15, 2)$, but it is apparent that the likelihood has a substantial effect *even though there are only 24 data points.* Substantively, we would have to say that this analysis only provides modest support for Romney's theory that believing specific diseases are contagious is a learned cultural response (although he looks at 26 additional diseases as well).

This relatively simple inference engine, as described, belies an immensely powerful procedure for developing and testing parametric models of inference. The central philosophical point is that the posterior distribution, which summarizes everything we know about the unknown parameter, is a weighted average of knowledge we possess before observing the data and the most likely value given the data. Furthermore, as the size of the data increases the likelihood becomes increasingly influential relative to the prior where in the limit the

prior is immaterial. This is powerful because it explicitly incorporates knowledge about the parameter that researchers possess before developing the empirical model and collecting the data.

2.4 Bayesian "Learning"

There is actually no restriction on what constitutes "prior information," provided it can be expressed in a distributional form, and as new data are observed a new posterior can be created by treating the old posterior as a prior and updating with the new data through the new likelihood function. This is a rigorous formulation for the way people think: we have opinions about the manner in which something works and this opinion is updated or altered as new behavior is observed. Suppose there exist three serial events, C, B, A, that are nonindependent and we wanted to update the joint probability distribution as each event occurs. The first is just $p(C)$, and no updating is needed. The second event occurs conditionally on the first and we get the joint probability distribution by serially updating: $p(B, C) = p(B|C)p(C)$. Now the third event occurs conditional on the first two, and the resulting joint distribution is a new update of the previous update: $p(C, B, A) = p(A|B, C)p(B|C)p(C)$. It is easy to see that this can continue as long as we want, and that as long as the last conditional is multiplied by the string of previous conditionals, we always obtain the complete joint. In this simple example, we could think of $p(C)$ as the first prior since it is not conditional on any other event.

To be a bit more statistically concrete about this point, start with a univariate prior distribution, $p(\theta)$, on an unknown variable θ. We observe the first set of iid data, $\mathbf{x}_1$, and calculate the posterior from the likelihood function along with the specified prior:

$$\pi_1(\theta|\mathbf{x}_1) \propto p(\theta)L(\theta|\mathbf{x}_1). \tag{2.18}$$

Subsequent to calculating this posterior, we observe a second set of iid data, $\mathbf{x}_2$, independent of the first set but from the same data-generating process. To *update* our posterior and therefore improve our state of knowledge, we simply treat the previous posterior as a prior and proceed to calculate using a likelihood function from the new data, and this is exactly the same result we would have obtained if all the data had arrived at once:

$$\begin{aligned} \pi_2(\theta|\mathbf{x}_1, \mathbf{x}_2) &\propto \pi_1(\theta|\mathbf{x}_1)L(\theta|\mathbf{x}_2) \\ &= p(\theta)L(\theta|\mathbf{x}_1)L(\theta|\mathbf{x}_2) \\ &= p(\theta)L(\theta|\mathbf{x}_1, \mathbf{x}_2). \end{aligned} \tag{2.19}$$

Needless to say this process can be repeated *ad infinitum* and the model will continue to update the posterior conclusions as new information arrives. This cycle of prior to posterior is actually a very principled way of conceptualizing the scientific process: we take what

knowledge we have in hand and update it with new information when such results become available.

■ **Example 2.5: Timing of Campaign Polls.** The updating characteristic of the Bayesian framework is ideal for analyzing time-series, either statically (after all the data have been collected) or dynamically (as it comes in). In many commercial and economic settings, data analysis is often performed continually as the daily, monthly, or yearly figures arrive. In modern elections for political office, campaigns will do multiple polls of the relevant electorate in order to update their strategies. Suppose a campaign observes the candidate support figures for period t, support_t, after the conclusion of the period. The question is: given the known information, including previous polling data prior to period t (D_t), what does this tell us about support for the next period? Formalized, this means that we have a posterior at the end of time period t that reflects the current understanding of the underlying nature of the support pattern for the candidate:

$$\pi(\text{support}_t|D_t). \tag{2.20}$$

Now before viewing the next period's support figures, we treat this exact same distribution as a new prior:

$$p(\text{support}_{t+1}) = \pi(\text{support}_t|D_t). \tag{2.21}$$

After period $t+1$, we have new support data and create a new posterior:

$$\pi(\text{support}_{t+1}|D_{t+1}) = p(\text{support}_{t+1})p(D_{t+1}). \tag{2.22}$$

Suppose that after period t, we were informed that a competing candidate suffered a public scandal. This is also new information (I_t) and would be incorporated in our prior for the next period's support:

$$p(\text{support}_{t+1}) = \pi(\text{support}_t|D_t, I_t). \tag{2.23}$$

What this means is that we should adjust our electoral expectations based on past support *and* this new information. The determination is still probabilistic because we cannot know for certain whether support will increase due to the competitors' misfortune: it could be that many of their supporters will move their intended vote to a third candidate, or they could be sufficiently loyal that the scandal is disregarded.

In a campaign environment, we might also want to predict support for future periods beyond a single period into the future. This is done by applying the sequential property of Bayes' Law:

$$p(\text{support}_{t+2}) = p(\text{support}_{t+1}|D_t)\pi(\text{support}_t|D_t). \tag{2.24}$$

This process can also be extended further into the future: $p(\text{support}_{t+k})$, albeit at the cost of progressively increasing uncertainty.

■ **Example 2.6: Example: A Bayesian Meta-Estimate of Deaths in Stalin's Gulags.** It is very difficult to obtain a reliable estimate of the number of people that perished in the Soviet Gulags (forced labor camps) during Stalin's era as dictator (1924-1953). While there will apparently never be a *definitive* answer (Solzhenitsyn 1997), Blyth (1995) uses Bayesian conditional inference to provide a meta-estimate based on the best guesses of multiple historical researchers. The basic idea is to take summary notions of the number of deaths by these experts and translate them into workable probability functions using Lindley's (1983) location-scale translation (and adjusting them by subjective assessments of possible prejudices). Building a likelihood function based on these estimates would generally be straightforward multiplicatively with independent guesses, except that the experts have seen and are influenced by each others' work. Blyth's solution to this nonindependence is to explicitly recognize the chronology, and to build the likelihood function by conditional updating.

Denote all widely available knowledge on the scale of Gulag deaths from demographics, journalistic descriptions, and published personal accounts as $\mathbf{X}$. The opinions of four experts and their associated estimates are considered by translating each best guess and level of uncertainty into a normal specification, or in the case where no uncertainty is given a uniform specification. For instance, in the case of one expert who estimates 10 to 20 million deaths, this is treated as a 95% credible interval. The normal distribution then is obtained by backing out the resulting coverage probability. The result is the following list of chronologically conditional statements:

Wiles, 1965:	$p(\theta_1\|\mathbf{X}) \sim \mathcal{U}(\theta)$
Kurganov, 1973:	$p(\theta_2\|\theta_1, \mathbf{X}) \sim \mathcal{U}(\theta)$
Conquest, 1978:	$p(\theta_3\|\theta_2, \theta_1, \mathbf{X}) \sim \mathcal{N}(18.2, 8.5)$
Medvedev, 1989:	$p(\theta_4\|\theta_3, \theta_2, \theta_1, \mathbf{X}) \sim \mathcal{N}(12, 9)$.

Therefore the likelihood function from these "data" is:

$$L(\theta|\theta_4, \theta_3, \theta_2, \theta_1, \mathbf{X}) = p(\theta_4|\theta_3, \theta_2, \theta_1, \mathbf{X})p(\theta_3|\theta_2, \theta_1, \mathbf{X})p(\theta_2|\theta_1, \mathbf{X})p(\theta_1|\mathbf{X}). \quad (2.25)$$

The "supra-Bayesian" posterior developed here is modeled as a normal weighted by the precisions with the assumption that the intermediate conditionals are normal, $\mathcal{N}(\mu_i, \sigma_i^2)$, and this posterior form is therefore given by $\mathcal{N}(\mu_\pi, \sigma_\pi^2)$ where $\sigma_\pi^2 = \left(\sum_{i=1}^4 \sigma_i^{-2}\right)^{-1}$ and $\mu_\pi = \sigma_\pi^2 \sum_{i=1}^4 (\mu_i/\sigma_i^2)$. Blyth assigns the relatively "diffuse" normal prior (i.e., widely spread out by specifying a large variance parameter) $\mathcal{N}(8, 12)$ and produces the posterior:

$$\pi(\theta|\theta_4, \theta_3, \theta_2, \theta_1, \mathbf{X}) \propto p(\theta)L(\theta|\theta_4, \theta_3, \theta_2, \theta_1, \mathbf{X}) \sim \mathcal{N}(13.2, 3.2).$$

This translates to a 95% credible interval of [9.7 : 16.7] million deaths, which is a compromise between the four experts and the author of the meta-analysis.

2.5 Comments on Prior Distributions

The most controversial aspect of Bayesian statistics is the necessary assignment of a prior distribution. The primary criticism here is that this is a subjective process or worse yet, that it is a tool that allows researchers to manipulate the probability calculations to obtain a desired result. In truth, there exist subjective aspects of *every* statistical model, including: the experimental design or observational setting that produces the data, the parametric form of the model, the specification of explanatory variables, the choice of hypotheses to be tested, the selected significance level, and the determination of an adequate sample size (Barnett 1973, p.160; Howson and Urbach 1993, p.12). Obviously we should add the choice of prior distribution to this list, but note that Bayesians spend considerably more time and energy defending a prior distribution than non-Bayesians do justifying other subjective decisions.

Prior distributions can be categorized as either proper or improper. *Proper priors* meet the Kolmogorov axioms, most specifically that they integrate or sum to a finite value. A non-normalized prior that integrates or sums to some positive value other than one can always be renormalized, and this distinction is immaterial with Bayes' Law expressed proportionally anyway. *Improper priors* are those that sum or integrate to infinity, and yet they are useful and play an important role in Bayesian inference.

Importantly, a standard maximum likelihood inferential model is identical to a Bayesian model in which the prior probability distribution is an appropriate (correctly bounded for the parameter at hand) uniform distribution function, and the two models are asymptotically identical for *any* proper prior distribution. Specifically, if $\hat{\theta}$ is the MLE and $\tilde{\theta}$ is the posterior mean from a Bayesian model using the *same* likelihood, but any proper prior (and most improper priors), then:

$$\sqrt{n}(\tilde{\theta} - \hat{\theta}) \underset{n\to\infty}{\longrightarrow} 0 \tag{2.26}$$

almost assuredly for reasonable starting values of θ (Chao 1970). This is not to say that prior distributions are actually *unimportant*, but rather that in the presence of overwhelming data size we should not care about whether the inferential model puts non-zero mass on the prior or not. More practically, there are plenty of instances where we cannot rely on data size alone to drive the quality of statistical inference. Is it unreasonable to study 25 European Union countries, 7 Central American countries, a set of small group experiments, presidential nominees, 15 CIS countries, classroom level education, or other comparable problems?

The strongest substantive argument for inclusion of priors is that there often exists scientific evidence at hand before the statistical model is developed and it would be foolish to ignore such previous knowledge (Tiao and Zellner 1964a; Press 1989, Section 2.7.1). Furthermore, a formal statement of the prior distribution is an overt, nonambiguous assertion within the model specification that the reader can accept or dismiss (Box and Tiao 1973, p.9; Gelman *et al.* 2003, p.14). Also, imprecise or vague knowledge often justifies a diffuse (very large variance) or even uniform (flat) prior if bounded (Jeffreys 1961, Chapter III; Zellner 1971, p.41ff), and certain probability models logically lead to particular forms of the prior for mathematical reasons (Good 1950; Press 1989).

An immediate payoff for applying this Bayesian framework is that it facilitates the explicit comparison of rival models about the system under study: H_1 and H_2 (even if these are not nested models). In a preview of Chapter 6, suppose Γ_1 and Γ_2 represent two competing hypotheses about the location of some unknown parameter, γ, which together form a partition of the sample space: $\Gamma = \Gamma_1 \cup \Gamma_2$. Initially prior probabilities are assigned to each of the two outcomes:

$$p_1 = p(\gamma \in \Gamma_1) \quad \text{and} \quad p_2 = p(\gamma \in \Gamma_2). \tag{2.27}$$

This allows us to calculate the posterior probabilities from the two alternative priors and the likelihood function:

$$\pi_1 = p(\gamma \in \Gamma_1 | D, H_1) \quad \text{and} \quad \pi_2 = p(\gamma \in \Gamma_2 | D, H_2). \tag{2.28}$$

The Bayes Factor combines the prior odds, p_1/p_2, and the posterior odds, π_1/π_2, as evidence for H_1 versus H_2 by calculating the ratio:

$$B = \frac{(\pi_1/\pi_2)}{(p_1/p_2)} \tag{2.29}$$

(Berger 1985; Kass and Raftery 1995; Lee 2004). Thus the Bayes Factor is the odds favoring H_1 versus H_2, given the observed data incorporating both prior and posterior information. As we will see in Chapter 7, this Bayes Factor model testing framework is even more flexible than this discussion implies.

2.6 Bayesian versus Non-Bayesian Approaches

There is a long history of antagonism between Bayesians and those adhering to strongly classical approaches: frequentist methods from Neyman and Pearson, and likelihood based methods from Fisher. However, this disagreement has greatly diminished over the last three decades. The core frequentist paradigm bases a sampling model on an imagined infinite series of replications of the same analysis where the reliability of the calculated

statistics is derived from their asymptotic properties. The likelihood approach is different in that only the currently observed sample is considered and statistics are produced from these data to estimate unknown population parameters by determining the value that is most likely, given that observed sample.

The likelihood theorem (Birnbaum 1962) states that all information that can be used for estimating some unknown parameter is contained in the likelihood function: a statement about the most likely value of the parameter given the observed data, and the Bayesian approach uses this same likelihood function (albeit with the addition of a prior distribution on the unknown parameters). Thus for likelihood inference, all information needed from the sample comes from the likelihood function. This does not mean that there is no additional information in the sample for other forms of inference. We know that every likelihood model is actually a Bayesian model with the appropriately bounded uniform prior and every Bayesian model is asymptotically equivalent to a corresponding likelihood model for any given prior. Therefore likelihoodists are simply Bayesians that do not know it or do not care to worry about the convenience of describing unknown quantities probabilistically. This means that the real differences are with classical frequentists.

There is a second distinction that causes more disagreement than it should. In classical inference, one assumes that the population parameters are fixed and unknown and therefore estimated with sample quantities. Conversely in Bayesian inference unknown inference parameters are treated as random quantities as a consequence of the application of Bayes' Law to invert the conditional probability statement. Actually this distinction is not very important in practice as the frequentist "sampling distribution" is exactly the same principle as the Bayesian posterior distribution, except that the imagined asymptote is unavailable. This explains why one hears social science researchers applying traditional inference procedures still using the word "posterior." Perhaps they have become frustrated with the difficulty in teaching the distinction between a *sample* distribution and a *sampling* distribution. Furthermore, Lewis (1994) points out the easily observed, but often forgotten fact that "Most applied statisticians have little interest in confrontation between rival philosophies but have a keen interest in pragmatic solutions to real problems ...." This is true of quantitative social scientists in particular.

A substantial amount of frequentist theory is built on the asymptotic normality of the sampling distribution of calculated statistics, and the associated calculation of such properties (Barndorff-Nielson and Cox 1989). Associated with this is the assumption of an unending stream of iid data. While Bayesian inference does not assume infinite replications to define sampling distributions, the posterior, being a compromise between the prior and the likelihood, will be affected by the same asymptotic properties as the amount of the data increases. Laplace was the first to note the near-normality of posterior distributions, as long ago as 1811! This property was later fully explored around the 1960s (Chao 1963, 1965), and Diaconis and Freedman (1986) subsequently gave mathematically rigorous conditions for the consistency of these Bayesian estimates, thus subjecting frequentist and Bayesian procedures to the same quality standard. 1970; Fabius 1964; Freedman

So it is important to understand where Bayesian inference stands relative to the Neyman-Pearson frequentist paradigm. We can now tabulate core differences between Bayesian and frequentist approaches. Most of these contrasts have been noted already in this chapter, and simply summarized here. This is done in the context of the following categories:

	Interpretation of Probability
Frequentist:	Observed result from infinite series of trials performed or imagined under identical conditions.
	Probabilistic quantity of interest is $p(\text{data}\|H_0)$.
Bayesian:	Probability is the researcher/observer "degree of belief" before or after the data are observed.
	Probabilistic quantity of interest is $p(\theta\|\text{data})$.
	What Is Fixed and Variable
Frequentist:	Data are an iid random sample from continuous stream.
	Parameters are fixed by nature.
Bayesian:	Data observed and so fixed by the sample generated.
	Parameters are unknown and described distributionally.
	How Results Are Summarized
Frequentist:	Point estimates and standard errors.
	95% confidence intervals indicating that 19/20 times the interval covers the true parameter value, on average.
Bayesian:	Descriptions of posteriors such as means and quantiles.
	Highest posterior density intervals indicating region of highest posterior probability.
	Inference Engine
Frequentist:	Deduction from $p(\text{data}\|H_0)$, by setting α in advance.
	Accept H_1 if $p(\text{data}\|H_0) < \alpha$.
	Accept H_0 if $p(\text{data}\|H_0) \geq \alpha$.
Bayesian:	Induction from $p(\theta\|data)$, starting with $p(\theta)$.
	$100(1-\alpha)\%$ of highest probability levels in $1-\alpha$ HPD region.
	Quality Checks
Frequentist:	Calculation of Type I and Type II errors.
	Sometimes: effect size and/or power.
	Usually: attention to small differences in p-values.
Bayesian:	Posterior predictive checks.
	Sensitivity of the posterior to forms of the prior.
	Bayes Factors, BIC, DIC (see Chapter 7).

In some ways, the seemingly wide gap between frequentist/likelihoodist and Bayesian thinking outlined above is an artificial and superficial divide. The maximum likelihood estimate is equal to the Bayesian posterior mode with the appropriate (correctly bounded) uniform prior, and they are asymptotically equal and normal given *any* proper prior (i.e., meeting the Kolmogorov axioms). Both approaches make extensive use of the central limit

theorem and normal theory in general. However, differences are seen particularly in small sample problems where the asymptotic equivalence is obviously not applicable. A common frequentist criticism of the Bayesian approach is that "subjective" priors have great impact on the posterior distribution for problems with small sample sizes. There is a developing literature on robust Bayesian analysis that seeks to mitigate this problem by developing estimators that are relatively insensitive to a wide range of prior distributions (Berger 1984).

■ **Example 2.7: The Timing of Polls.** Bernardo (1984) developed a precinct-level Bayesian hierarchical model of vote choice for the Spanish election of 1982 in which the Socialist party obtained control of the government for the first time since the Civil War. Bayesian hierarchical models recognize and organize differing levels of data and prior information (see Chapter 12 for more details). The author defines n_{ij} as the number of voters in the i^{th} precinct voting for the j^{th} party. The data from the m precincts surveyed ($\{n_{1,j}, n_{2,j}, \ldots, n_{m,j}\}$, $j = 1$ to 5 major political parties) are assumed to be from an underlying multinomial distribution with unknown parameters θ_{ij} representing the *probability* of a vote for the j^{th} party in the i^{th} precinct with the constraints that these values are nonnegative and sum to one. Bernardo specifies a uniform prior distribution on the θ_{ij} values and this leads naturally to a Dirichlet form (a multivariate generalization of the beta) of the posterior.

TABLE 2.2: HPD Regions: Predicted 1982 Vote Percentages for Five Major Parties

Valencia	Party				
	Socialist	Conservative	Center	Center-Left	Communist
Four weeks before election	[39.0:48.9]	[12.6:19.4]	[7.9:13.2]	[4.0: 7.8]	[5.1:9.2]
One week before election	[47.3:54.2]	[13.3:24.9]	[5.0:11.8]	[7.0:11.9]	[4.0:6.7]
First 100 votes from 20 polls	[49.0:57.7]	[23.5:31.2]	[2.3: 4.6]	[1.1: 2.9]	[4.0:6.2]
Total vote from 20 polls	[50.1:56.8]	[26.6:32.6]	[3.3: 4.6]	[1.8: 2.7]	[3.7:5.6]
Actual results	53.5	29.4	4.4	2.3	5.3

A substantively interesting aspect of the methodology is the scheduling of data collection and analysis. Data are collected in the province of Valencia at four points in time: by a survey four weeks before the election ($n = 1,000$), by a survey one week before the election ($n = 1,000$), using the first 100 valid votes from 20 representative polling stations, and all valid votes from these same polling stations after the polls are closed. Data collection was performed with the full cooperation of the Spanish government and the results were immediately provided to the national media.

Bernardo presents the Bayesian estimates of predicted vote proportion by party as 0.90 highest posterior density regions. These results are summarized in Table 2.2.

One interesting result from this analysis is that the 0.90 HPD regions shrink as the final tally nears and better polling data are received. This reflects the growing certainty about the estimates as data quality improves. In addition, the estimates from actual polling data are remarkably accurate for the two parties receiving the largest vote share. Note that the uniform prior does not constrain the final, highly nonuniform, posterior distribution.

The last example demonstrates that Bayesian data analysis is essentially free from the well-known problems with the null hypothesis significance test. Inferences are communicated to the reader without artificial decisions, p-values, and confused conditional probability statements. The Bayesian approach also interprets sample size increases in a more desirable manner: larger sample sizes reduce the importance of prior information rather than guarantee a low but meaningless p-value.

2.7 Exercises

2.1 Suppose that 25 out of 30 firms develop new marketing plans during the next year. Using the beta-binomial model from Section 2.3.4 starting on page 49, apply a $\mathcal{BE}(0.5, 0.5)$ (Jeffreys prior) and then specify a normal prior centered at zero and truncated to fit on $[0:1]$ as prior distributions and plot the respective posterior densities. What differences do you observe?

2.2 Derive the posterior distribution for a sample of size n of iid data distributed $f(X|\theta) = \theta e^{-\theta X}$, $X, \theta \in (0, \infty)$), with a prior for θ, $p(\theta) = \beta e^{-\theta\beta}$, $\beta \in (0, \infty)$. What is common to all three distributions?

2.3 Prove that the gamma distribution,

$$f(\mu|\alpha, \beta) = \frac{1}{\Gamma(\alpha)}\beta^{\alpha}\mu^{\alpha-1}e^{-\beta\mu}, \qquad \mu, \alpha, \beta > 0,$$

is the conjugate prior distribution for μ in a Poisson likelihood function,

$$f(\mathbf{y}|\mu) = \left(\prod_{i=1}^{n} y_i!\right)^{-1} \exp\left[\log(\mu)\sum_{i=1}^{n} y_i\right] \exp[-n\mu],$$

that is, calculate a form for the posterior distribution of μ and show that it is also gamma distributed.

2.4 One requirement for specifying prior distributions is that the support of the assigned

prior must match the allowable range of the parameter being modeled. What common distributions, without modification, can be used as priors on model variance components?

2.5 Use the gamma-Poisson conjugate specification developed in Exercise 2.3 to analyze the following count data on worker strikes in Argentina over the period 1984 to 1993, from McGuire (1996). Assign your best guess as to reasonable values for the two parameters of the gamma distribution: α and β. Produce the posterior distribution for μ and describe it with quantiles and graphs using empirically simulated values according to the following procedure:

- ▷ The posterior distribution for μ is gamma(δ_1, δ_2) according to some parameters δ_1 and δ_2 that you derived above, which of course depends on your choice of the gamma parameters.
- ▷ Generate a large number of values from this distribution in `R`, say 10,000 or so, using the command:
  ```
  posterior.sample <- rgamma(10000,d1,d2)
  ```
- ▷ Produce posterior quantiles, such as the interquartile range, according to:
  ```
  iqr.posterior <- c(sort(posterior.sample)[2500],
                     sort(posterior.sample)[7500])
  ```
  Note: the `IQR` function in `R` gives a single value for the difference, which is not as useful.
- ▷ Graph the posterior in different ways, such as with a smoother like lowess (a local-neighborhood smoother, see Cleveland [1979, 1981]):
  ```
  post.hist <- hist(posterior.sample,plot=F,breaks=100)
  plot(lowess(post.hist$mids,post.hist$intensities),
  type="l")
  ```

Economic Sector	Number of Strikes		
Public Administrators	496	Meat Packers	56
Teachers	421	Paper Industry Workers	55
Metalworkers	199	Sugar Industry Workers	50
Municipal Workers	186	Public Services	47
Private Hospital Workers	181	University Staff Employees	43
Bank Employees	133	Telephone Workers	39
Court Clerks	128	Textile Workers	37
Bus Drivers	113	State Petroleum Workers	32
Construction Workers	92	Food Industry Workers	28
Doctors	83	Post Office Workers	26
Nationalized Industries	77	Locomotive Drivers	25
Railway Workers	76	Light and Power Workers	21
Maritime Workers	57	TOTAL	2701

2.6 For $\theta \sim \text{binomial}(10, 0.5)$ construct an even tail credible interval that has *at least* 0.90 coverage. Is it possible to get exact coverage?

2.7 In his original essay (1763, p.376) Bayes offers the following question:

> *Given* the number of times in which an unknown event has happened and failed: *Required* the chance that the probability of its happening in a single trial lies somewhere between any two degrees of probability that can be named.

Provide an analytical expression for this quantity using an appropriate uniform prior (Bayes argued reluctantly for the use of the uniform as a "no information" prior: *Bayes postulate*).

2.8 Given a proper prior distribution, $p(\theta)$, and a likelihood function, $L(\theta|\mathbf{X})$, demonstrate that the only way that the prior distribution and the resulting posterior distribution, $\pi(\theta|\mathbf{X})$ can be identical is when the likelihood function does not contain θ.

2.9 Suppose we have two urns containing marbles; the first contains 6 red marbles and 4 green marbles, and the second contains 9 red marbles and 1 green marble. Now we take one marble from the first urn (without looking at it) and put it in the second urn. Subsequently, we take one marble from the second urn (again without looking at it) and put it in the first urn. Give the full probabilistic statement of the probability of now drawing a red marble from the first urn, and calculate its value.

2.10 In an experimental context Gill and Freeman (2013) ask participants to answer a wide range of background questions prior to eliciting prior distributions from watching video clips. One of these,

> "What proportion (percent) of undergraduate students at the University of Minnesota are women?"

generates the following response times in seconds to this question:

7	7	11	7	7	10	7	5	8	7	5	7	12	6	8	7
8	7	28	13	6	4	10	6	13	11	6	14	4	7	12	16
8	9	8	9	4	5	8	4	5	15	9	7	7	8	4	9
7	9	19	19	9	7	5	6	6	17	7	6	10	7	15	

Assume that the distribution of these times is $\mathcal{G}(4.5, 2)$ (shape and scale). Find and graph a 95% HPD region for an additional sample point drawn from the same population. Now suppose we do not know the distribution with certainly but impose a prior distribution that is $\mathcal{G}(3, 3)$. Find and graph the resulting 95% HPD regions for the posterior distribution.

2.11 This is the famous envelope problem. You and another contestant are each given one sealed envelope containing some quantity of money with equal probability of receiving either envelope. All you know at the moment is that one envelope contains twice the cash as the other. So if you open your envelope and observe \$10, then the other envelope contains either \$5 or \$20 with equal probability. You are now given the opportunity to trade with the other contestant. Should you? The expected value of the unseen envelope is $E[other] = 0.5(5) + 0.5(20) = 12.50$, meaning that you have a higher expected value by trading. Interestingly, so does the other player for analogous reasons. Now suppose you are offered the opportunity to trade again before you open the newly traded envelope. Should you? What is the expected value of doing so? Explain how this game leads to infinite cycling. There is a Bayesian solution. Define M as the known maximum value in either envelope, stipulate a probability distribution, and identify a suitable prior.

2.12 Radiocarbon dating of the famous *Shroud of Turin* cloth that some believe was used to wrap Jesus Christ's body (since it has front and rear impressions of a bearded male with whipping and crucifixion injuries) was done by the "Arizona Group" (Linick *et al.* 1986) using accelerator mass spectrometry. Their serial process in 1988 produced the following estimated ages in years and associated standard errors:

Iteration	Mean	SE
1	606	51
2	574	52
3	753	51
4	632	49
5	676	59
6	540	57
7	701	47
8	701	47

As done in Example 2.4 starting on page 55, treat these as consecutive updates on the posterior distribution and produce the set of posterior distributions under normally distributed assumptions for each. Stipulate a reasonable prior to begin the process.

2.13 If the posterior distribution of θ is $\mathcal{N}(1, 3)$, then calculate a 99% HPD region for θ.

2.14 Given a posterior distribution for θ that is $\mathcal{BE}(0.5, 0.5)$, calculate the 95% HPD region for θ.

2.15 Assume that the data $[1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1]$ are produced from iid Bernoulli trials. Produce a $1 - \alpha$ credible set for the unknown value of p using a uniform prior distribution.

2.16 Browne, Frendreis, and Gleiber (1986) tabulate complete cabinet duration (constitutional inter-election period) for eleven Western European countries from 1945 to 1980 for annualized periods:

Country	N	Average Duration
Italy	38	0.833
Finland	28	1.070
Belgium	27	1.234
Denmark	20	1.671
Norway	17	2.065
Iceland	15	2.080
Austria	15	2.114
West Germany	15	2.168
Sweden	15	2.274
Ireland	14	2.629
Netherlands	12	2.637

The country averages from the third column of the table are weighted by N in the second column to reflect the number of such events: $\mathbf{X}_i N_i$. Assume that the durations, $\mathbf{X}$, are exponentially distributed $p(X|\theta) = \theta e^{-\theta X}$ defined over $(0, \infty)$, and like Example 2.3.2.1 on page 44 specify the prior distribution of $p(\theta) = 1/\theta$, for $\theta \in (0{:}\infty)$. Calculate an equal tail credible interval and an HPD region for the resulting posterior distribution of θ. Plot the posterior density and indicate the location of these intervals.

2.17 The beta distribution, $f(x|\alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}, 0 < x < 1, \alpha > 0, \beta > 0$, is often used to model the probability parameter in a binomial setup. If you were very unsure about the prior distribution of p, what values would you assign to α and β to make it relatively "flat"?

2.18 An *improper prior distribution* is a function that does not sum or integrate to a finite constant. Show that it is possible to still get a proper posterior distribution through (2.6). A possible prior for μ in a Poisson likelihood function is $p(\mu) = 1/\mu$. Show that this is improper.

2.19 Laplace (1774, p.28) derives Bayes' Law for uniform priors. His claim is

> . . . je me propose de déterminer la probabilité des causes par les événements matière neuve à bien des égards et qui mérite d'autant plus d'être cultivée que c'est principalement sous ce point de vue que la science des hasards peut être utile dans la vie civile.

He starts with two events: E_1 and E_2 and n causes: $A_1, A_2, \ldots, A_n$. The assumptions are: *(1)* E_i are *conditionally independent* given A_i, and *(2)* A_i are equally

probable. Derive Laplace's inverse probability relation:

$$p(A_i|E) = \frac{p(E|A_i)}{\sum_j p(E|A_j)}.$$

2.20 Martins (2009) is concerned with Bayesian updating by interacting actors who pay attention to each others' choices. Actor i has a prior distribution $f_i(\theta)$, and $E[\theta] = x_i$. This actor's posterior for θ is affected by the average estimates of others x_j, giving $f_i(\theta|x_j)$. Show that the mixture likelihood:

$$f(x_j|\theta) = \omega\mathcal{N}(\theta, \sigma_j) + (1-\omega)\mathcal{U}(0,1),$$

(for mixture parameter ω), leads to the posterior:

$$f(\theta|x_j) \propto \omega \exp\left[-\frac{1}{2\sigma_i^2}((\theta - x_i)^2 + (x_j - \theta)^2)\right](1-\omega)\exp\left[-\frac{1}{2\sigma_i^2}(x_i - x_j)^2\right].$$

2.8 Computational Addendum: R for Basic Analysis

This code gives the analysis and graphing of the cultural anthropology example in Section 2.3.4 starting on page 51.

```
par(oma=c(1,1,1,1),mar=c(0,0,0,0),mfrow=c(2,1))
x <- c(1,1,1,1,0,1,1,0,1,0,1,1,1,0,1,1,1,1,1,1,0,0,0,1)
ruler <- seq(0,1,length=300)

A <- 15; B <- 2
beta.prior <- dbeta(ruler,A,B)
beta.posterior <- dbeta(ruler,sum(x)+A,length(x)-sum(x)+B)
plot(ruler,beta.prior, ylim=c(-0.7,9.5),
        xaxt="n", yaxt="n", xlab="", ylab="", pch=".")
lines(ruler,beta.posterior)
hpd.95 <- qbeta(c(0.025,0.975),sum(x)+A,length(x)-sum(x)+B)
segments(hpd.95[1],0,hpd.95[2],0,lwd=4)
text(mean(hpd.95),-0.4,"95% HPD Region",cex=0.6)
text(0.25,5,paste("Beta(",A,",",B,
    ") prior,  95% HPD Regionat: [",round(hpd.95[1],3),
    ":",round(hpd.95[2],3),"]",sep=""),cex=1.1)

A <- 1; B <- 1
beta.prior <- dbeta(ruler,A,B)
beta.posterior <- dbeta(ruler,sum(x)+A,length(x)-sum(x)+B)
plot(ruler,beta.prior, ylim=c(-0.7,9.5),
        xaxt="n", yaxt="n", xlab="", ylab="", pch=".")
lines(ruler,beta.posterior)
```

```
hpd.95 <- qbeta(c(0.025,0.975),sum(x)+A,length(x)-sum(x)+B)
segments(hpd.95[1],0,hpd.95[2],0,lwd=4)
text(mean(hpd.95),-0.4,"95% HPD Region",cex=0.6)
text(0.25,5,paste("Beta(",A,",",B,
    ") prior,  95% HPD Region at: [",round(hpd.95[1],3),
    ":",round(hpd.95[2],3),"]",sep=""),cex=1.1)
```

The following is the simple HPD region calculation used in Example 2.3.2.2.

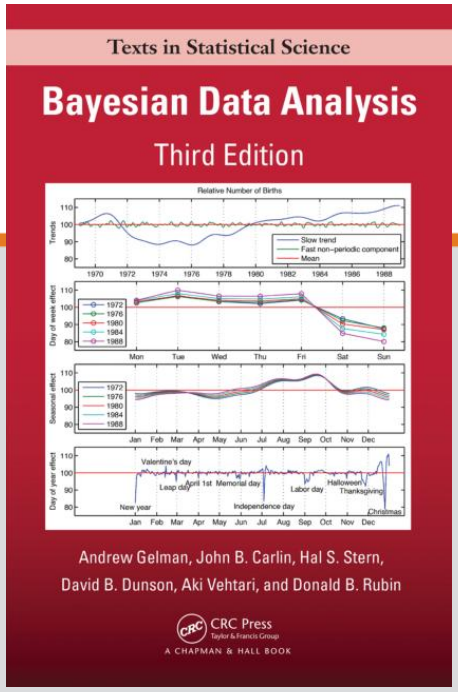
```
hpd.gamma <- function(g.shape,g.rate,target=0.90,steps=300,tol=0.01)  {
    if (steps %% 2 == 1) steps <- steps + 1
    g.mode  <- sum(state.df$N)/sum(state.df$N*state.df$dur)
    g.range <- seq(qgamma(0.001,g.shape,g.rate), qgamma(0.999,g.shape,
                   g.rate),length=steps)
    g.range <- c(g.range[1:(steps/2)],g.mode,g.range[(steps/2+1):steps])
    g.dens  <- dgamma(g.range,g.shape,g.rate)
    g.probs <- pgamma(g.range,g.shape,g.rate)
    for (i in 1:(steps/2)) {
        k.dir <- which(c(g.dens[(steps/2-i)],g.dens[(steps/2+i)]) ==
                       max(g.dens[(steps/2-i)],g.dens[(steps/2+i)]))
        k <- c(g.dens[(steps/2-i)],g.dens[(steps/2+i)])[k.dir]
        k.loc <- c((steps/2-i),(steps/2+i))[k.dir]
        if (k.dir == 2) k2.range <- c(1:(steps/2))
else k2.range <- c((steps/2 + 1):steps)
        k2.min <- which(abs(k-g.dens[k2.range])==min(abs(k-g.dens[k2.range])))
        if (k.dir == 1) k2.min <- k2.min + steps/2
        if (g.probs[k.loc] + (1-g.probs[k2.min]) < 1-target)  break
        bounds <- c(g.range[k.loc],g.range[k2.min])
    }
    return(list("cdf.vals"=c(g.probs[k.loc],g.probs[k2.min]),
                "bounds"=bounds,"k"=k))
}

state.hpd <- hpd.gamma(g.shape=sum(state.df$N),
                       g.rate=sum(state.df$N*state.df$dur))
```

CHAPTER

4

HIERARCHICAL LINEAR MODELS



This chapter is excerpted from

Bayesian Data Analysis, Third Edition

by Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, Donald B. Rubin.

Learn more

Chapter 15

Hierarchical linear models

Hierarchical regression models are useful as soon as there are predictors at different levels of variation. For example, in studying scholastic achievement we may have information about individual students (for example, family background), class-level information (characteristics of the teacher), and also information about the school (educational policy, type of neighborhood). Another situation in which hierarchical modeling arises naturally is in the analysis of data obtained by stratified or cluster sampling. A natural family of models is regression of y on indicator variables for strata or clusters, in addition to any measured predictors x. With cluster sampling, hierarchical modeling is in fact *necessary* in order to generalize to the unsampled clusters.

With predictors at multiple levels, the assumption of exchangeability of units or subjects at the lowest level breaks down. The simplest extension from classical regression is to introduce as predictors a set of indicator variables for each of the higher-level units in the data—that is, for the classes in the educational example or for the strata or clusters in the sampling example. But this will in general dramatically increase the number of parameters in the model, and sensible estimation of these is only possible through further modeling, in the form of a population distribution. The latter may itself take a simple exchangeable or independent and identically distributed form, but it may also be reasonable to consider a further regression model at this second level, to allow for the predictors defined at this level. In principle there is no limit to the number of levels of variation that can be handled in this way. Bayesian methods provide ready guidance on handling the estimation of unknown parameters, although computational complexities can be considerable, especially if one moves out of the realm of conjugate normal specifications. In this chapter we give a brief introduction to the broad topic of hierarchical linear models, emphasizing the general principles used in handling normal models.

In fact, we have already considered a hierarchical linear model in Chapter 5: the problem of estimating several normal means can be considered as a special case of linear regression. In the notation of Section 5.5, the data points are y_j, $j = 1, \ldots, J$, and the regression coefficients are the school parameters θ_j. In this example, therefore, $n = J$; the number of 'data points' equals the number of explanatory variables. The X matrix is just the $J \times J$ identity matrix, and the individual observations have known variances σ_j^2. Section 5.5 discussed the flaws of no pooling and complete pooling of the data, $y_1, \ldots, y_J$, to estimate the parameters, θ_j. In the regression context, no pooling corresponds to a noninformative uniform prior distribution on the regression coefficients, and complete pooling corresponds to the J coefficients having a common prior distribution with zero variance. The favored hierarchical model corresponds to a prior distribution of the form $\beta \sim \mathrm{N}(\mu, \tau^2 I)$.

In the next section, we present notation and computations for the simple varying-coefficients model, which constitutes the simplest version of the general hierarchical linear model (of which the eight schools example of Section 5.5 is in turn a simple case). We illustrate in Section 15.2 with the example of forecasting U.S. presidential elections, and then go on to the general form of the hierarchical linear model in Section 15.3. Throughout, we

assume a normal linear regression model for the likelihood, $y \sim \mathrm{N}(X\beta, \Sigma_y)$, as in Chapter 14, and we label the regression coefficients as β_j, $j = 1, \ldots, J$.

15.1 Regression coefficients exchangeable in batches

We begin by considering hierarchical regression models in which groups of the regression coefficients are exchangeable and are modeled with normal population distributions. Each such group is called a batch of *random effects* or *varying coefficients*.

Simple varying-coefficients model

In the simplest form of the random-effects or varying-coefficients model, all of the regression coefficients contained in the vector β are exchangeable, and their population distribution can be expressed as

$$\beta \sim \mathrm{N}(1\alpha, \sigma_\beta^2 I), \tag{15.1}$$

where α and σ_β are unknown scalar parameters, and 1 is the $J \times 1$ vector of ones, $1 = (1, \ldots, 1)^T$. We use this vector-matrix notation to allow for easy generalization to regression models for the coefficients β, as we discuss in Section 15.3. Model (15.1) is equivalent to the hierarchical model we applied to the educational testing example of Section 5.5, using $(\beta, \alpha, \sigma_\beta)$ in place of (θ, μ, τ). As in that example, this general model includes, as special cases, unrelated β_j's ($\sigma_\beta = \infty$) and all β_j's equal ($\sigma_\beta = 0$).

It can be reasonable to start with a prior density that is uniform on α, σ_β, as we used in the educational testing example. As discussed in Section 5.4, we cannot assign a uniform prior distribution to $\log \sigma_\beta$ (the standard 'noninformative' prior distribution for variance parameters), because this leads to an improper posterior distribution with all its mass in the neighborhood of $\sigma_\beta = 0$. Another relatively noninformative prior distribution that is often used for σ_β^2 is scaled inverse-χ^2 (see Appendix A) with the degrees of freedom set to a low number such as 2. In applications one should be careful to ensure that posterior inferences are not sensitive to these choices; if they are, then greater care needs to be taken in specifying prior distributions that are defensible on substantive grounds. If there is little replication in the data at the level of variation corresponding to a particular variance parameter, then that parameter is generally not well estimated by the data and inferences may be sensitive to prior assumptions.

Intraclass correlation

There is a straightforward connection between the varying-coefficients model just described and a within-group correlation. Suppose data $y_1, \ldots, y_n$ fall into J batches and have a multivariate normal distribution: $y \sim \mathrm{N}(\alpha\mathbf{1}, \Sigma_y)$, with $\mathrm{var}(y_i) = \eta^2$ for all i, and $\mathrm{cov}(y_{i_1}, y_{i_2}) = \rho\eta^2$ if i_1 and i_2 are in the same batch and 0 otherwise. (We use the notation $\mathbf{1}$ for the $n \times 1$ vector of 1's.) If $\rho \geq 0$, this is equivalent to the model $y \sim \mathrm{N}(X\beta, \sigma^2 I)$, where X is a $n \times J$ matrix of indicator variables with $X_{ij} = 1$ if unit i is in batch j and 0 otherwise, and β has the varying-coefficients population distribution (15.1). The equivalence of the models occurs when $\eta^2 = \sigma^2 + \sigma_\beta^2$ and $\rho = \sigma_\beta^2/(\sigma^2 + \sigma_\beta^2)$, as can be seen by deriving the marginal distribution of y, averaging over β. More generally, positive intraclass correlation in a linear regression can be subsumed into a varying-coefficients model by augmenting the regression with J indicator variables whose coefficients have the population distribution (15.1).

Mixed-effects model

An important variation on the simple varying-coefficients or random-effects model is the 'mixed-effects model,' in which the first J_1 components of β are assigned independent im-

proper prior distributions, and the remaining $J_2 = J - J_1$ components are exchangeable with common mean α and standard deviation σ_β. The first J_1 components, which are implicitly modeled as exchangeable with infinite prior variance, are sometimes called *fixed effects*.[1]

A simple example is the hierarchical normal model considered in Chapter 5; the varying-coefficients model with the school means normally distributed and a uniform prior density assumed for their mean α is equivalent to what is sometimes called a mixed-effects model with a single constant 'fixed effect' and a set of random effects with mean 0.

Several sets of varying coefficients

To generalize, allow the J components of β to be divided into K clusters of coefficients, with cluster k having population mean α_k and standard deviation $\sigma_{\beta k}$. A mixed-effects model is obtained by setting the variance to ∞ for one of the clusters of coefficients. We return to these models in discussing the analysis of variance in Section 15.7.

Exchangeability

The essential feature of varying-coefficient models is that exchangeability of the units of analysis is achieved by conditioning on indicator variables that represent groupings in the population. The varying coefficients allow each subgroup to have a different mean outcome level, and averaging over these parameters to a marginal distribution for y induces a correlation between outcomes observed on units in the same subgroup (just as in the simple intraclass correlation model described above).

15.2 Example: forecasting U.S. presidential elections

We illustrate hierarchical linear modeling with an example in which a hierarchical model is useful for obtaining realistic forecasts. Following standard practice, we begin by fitting a nonhierarchical linear regression with a noninformative prior distribution but find that the simple model does not provide an adequate fit. Accordingly we expand the model hierarchically, including varying coefficients to model variation at a second level in the data.

Political scientists in the U.S. have been interested in the idea that national elections are highly predictable, in the sense that one can accurately forecast election results using information publicly available several months before the election. In recent years, several different linear regression forecasts have been suggested for U.S. presidential elections. In this chapter, we present a hierarchical linear model that was estimated from the elections through 1988 and used to forecast the 1992 election.

Unit of analysis and outcome variable

The units of analysis are results in each state from each of the 11 presidential elections from 1948 through 1988. The outcome variable of the regression is the Democratic party candidate's share of the two-party vote for president in that state and year. For convenience and to avoid tangential issues, we discard the District of Columbia (in which the Democrats have received over 80% in every presidential election) and states with third-party victories from our model, leaving us with 511 units from the 11 elections considered.

[1]The terms 'fixed' and 'random' come from the non-Bayesian statistical tradition and are somewhat confusing in a Bayesian context where all unknown parameters are treated as 'random' or, equivalently, as having fixed but unknown values.

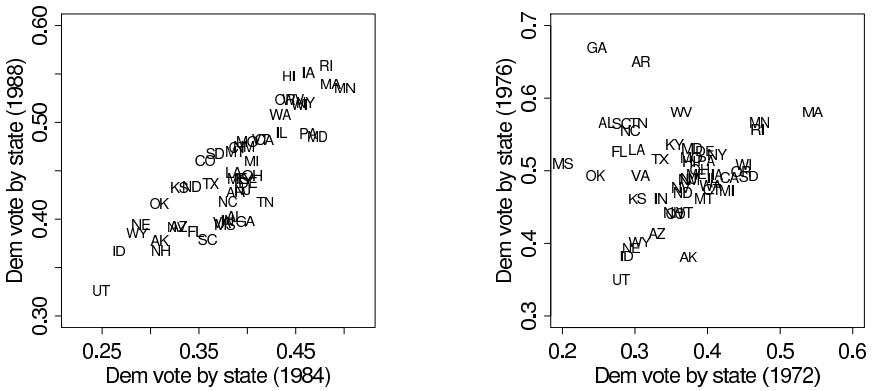


Figure 15.1 *(a) Democratic share of the two-party vote for president, for each state, in 1984 and 1988. (b) Democratic share of the two-party vote for president, for each state, in 1972 and 1976.*

Preliminary graphical analysis

Figure 15.1a suggests that the presidential vote may be strongly predictable from one election to the next. The fifty points on the figure represent the states of the U.S. (indicated by their two-letter abbreviations); the x and y coordinates of each point show the Democratic party's share of the vote in the presidential elections of 1984 and 1988, respectively. The points fall close to a straight line, indicating that a linear model predicting y from x is reasonable and relatively precise. The pattern is not always so strong, however; consider Figure 15.1b, which displays the votes by states in 1972 and 1976—the relation is not close to linear. Nevertheless, a careful look at the second graph reveals some patterns: the greatest outlying point, on the upper left, is Georgia ('GA'), the home state of Jimmy Carter, the Democratic candidate in 1976. The other outlying points, all on the upper left side of the 45° line, are other states in the South, Carter's home region. It appears that it may be possible to create a good linear fit by including other predictors in addition to the Democratic share of the vote in the previous election, such as indicator variables for the candidates' home states and home regions. (For political analysis, the United States is typically divided into four regions: Northeast, South, Midwest, and West, with each region containing ten or more states.)

Fitting a preliminary, nonhierarchical, regression model

Political trends such as partisan shifts can occur nationwide, at the level of regions of the country, or in individual states; to capture these three levels of variation, we include three kinds of explanatory variables in the regression. The nationwide variables—which are the same for every state in a given election year—include national measures of the popularity of the candidates, the popularity of the incumbent President (who may or may not be running for reelection), and measures of the condition of the economy in the past two years. Regional variables include home-region indicators for the candidates and various adjustments for past elections in which regional voting had been important. Statewide variables include the Democratic party's share of the state's vote in recent presidential elections, measures of the state's economy and political ideology, and home-state indicators. The explanatory variables used in the model are listed in Table 15.1. With 511 observations, a large number of state and regional variables can reasonably be fitted in a model of election outcome, assuming there are smooth patterns of dependence on these covariates across states and regions. Fewer relationships with national variables can be estimated, however, since for this purpose there are essentially only 11 data points—the national elections.

Description of variable	Sample quantiles		
	min	median	max
Nationwide variables:			
Support for Dem. candidate in Sept. poll	0.37	0.46	0.69
(Presidential approval in July poll) × Inc	−0.69	−0.47	0.74
(Presidential approval in July poll) × Presinc	−0.69	0	0.74
(2nd quarter GNP growth) × Inc	−0.024	−0.005	0.018
Statewide variables:			
Dem. share of state vote in last election	−0.23	−0.02	0.41
Dem. share of state vote two elections ago	−0.48	−0.02	0.41
Home states of presidential candidates	−1	0	1
Home states of vice-presidential candidates	−1	0	1
Democratic majority in the state legislature	−0.49	0.07	0.50
(State economic growth in past year) × Inc	−0.22	−0.00	0.26
Measure of state ideology	−0.78	−0.02	0.69
Ideological compatibility with candidates	−0.32	−0.05	0.32
Proportion Catholic in 1960 (compared to U.S. avg.)	−0.21	0	0.38
Regional/subregional variables:			
South	0	0	1
(South in 1964) × (−1)	−1	0	0
(Deep South in 1964) × (−1)	−1	0	0
New England in 1964	0	0	1
North Central in 1972	0	0	1
(West in 1976) × (−1)	−1	0	0

Table 15.1 *Variables used for forecasting U.S. presidential elections. Sample minima, medians, and maxima come from the 511 data points. All variables are signed so that an increase in a variable would be expected to increase the Democratic share of the vote in a state. 'Inc' is defined to be* +1 *or* −1 *depending on whether the incumbent President is a Democrat or a Republican. 'Presinc' equals Inc if the incumbent President is running for reelection and 0 otherwise. 'Dem. share of state vote' in last election and two elections ago are coded as deviations from the corresponding national votes, to allow for a better approximation to prior independence among the regression coefficients. 'Proportion Catholic' is the deviation from the average proportion in 1960, the only year in which a Catholic ran for President. See Gelman and King (1993) and Boscardin and Gelman (1996) for details on the other variables, including a discussion of the regional/subregional variables. When fitting the hierarchical model, we also included indicators for years and regions within years.*

For a first analysis, we fit a classical regression including all the variables in Table 15.1 to the data up to 1988, as described in Chapter 14. We could then draw simulations from the posterior distribution of the regression parameters and use each of these simulations, applied to the national, regional, and state explanatory variables for 1992, to create a random simulation of the vector of election outcomes for the fifty states in 1992. These simulated results could be used to estimate the probability that each candidate would win the national election and each state election, the expected number of states each candidate would win, and other predictive quantities.

Checking the preliminary regression model

The ordinary linear regression model ignores the year-by-year structure of the data, treating them as 511 independent observations, rather than 11 sets of roughly 50 *related* observations each. Substantively, the feature of these data that such a model misses is that partisan support across the states does not vary independently: if, for example, the Democratic candidate for President receives a higher-than-expected vote share in Massachusetts in a

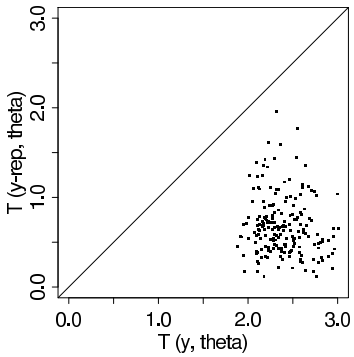


Figure 15.2 *Scatterplot showing the joint distribution of simulation draws of the realized test quantity, $T(y, \beta)$—the square root of the average of the 11 squared nationwide residuals—and its hypothetical replication, $T(y^{\text{rep}}, \beta)$, under the nonhierarchical model for the election forecasting example. The 200 simulated points are far below the 45° line, which means that the realized test quantity is much higher than predicted under the model.*

particular year, we would expect him also to perform better than expected in Utah in that year. In other words, because of the known grouping into years, the assumption of exchangeability among the 511 observations does not make sense, *even after controlling for the explanatory variables*.

An important use of the model is to forecast the nationwide outcome of the presidential election. One way of assessing the significance of possible failures in the model is to use the model-checking approach of Chapter 6. To check whether correlation of the observations from the same election has a substantial effect on nationwide forecasts, we create a test variable that reflects the average precision of the model in predicting the national result—the square root of the average of the squared nationwide realized residuals for the 11 general elections in the dataset. (Each nationwide realized residual is the average of $(y_i - X_i\beta)$ for the roughly 50 observations in that election year.) This test variable should be sensitive to positive correlations of outcomes in each year. We then compare the values of the test variable $T(y, \beta)$ from the posterior simulations of β to the hypothetical replicated values under the model, $T(y^{\text{rep}}, \beta)$. The results are displayed in Figure 15.2. As can be seen in the figure, the observed variation in national election results is larger than would be expected from the model. The practical consequence of the failure of the model is that its forecasts of national election results are falsely precise.

Extending to a varying-coefficients model

We can improve the regression model by adding an additional predictor for each year to serve as an indicator for nationwide partisan shifts unaccounted for by the other national variables; this adds 11 new components of β corresponding to the 11 election years in the data. The additional columns in X are indicator vectors of zeros and ones indicating which observations correspond to which year. After controlling for the national variables already in the model, we fit an exchangeable model for the election-year variables, which in the normal model implies a common mean and variance. Since a constant term is already in the model, we can set the mean of the population distribution of the year-level errors to zero (recall Section 15.1). By comparison, the classical regression model we fitted earlier is a special case of the current model in which the variance of the 11 election-year coefficients is fixed at zero.

In addition to year-to-year variability not captured by the model, there are also electoral

swings that follow the region of the country—Northeast, South, Midwest, and West. To capture regional variability, we include 44 region × year indicator variables (also with the mean of the population distributions set to zero) to cover all regions in all election years. Within each region, we model these indicator variables as exchangeable. Because the South tends to act as a special region of the U.S. politically, we give the 11 Southern regional variables their own common variance, and treat the remaining 33 regional variables as exchangeable with their own variance. In total, we have added 55 new β parameters and three new variance components to the model, and we have excluded the regional and subregional corrections in Table 15.1 associated with specific years. We can write the model for data in states s, regions $r(s)$, and years t,

$$\begin{aligned} y_{st} &\sim \mathrm{N}(X_{st}\beta + \gamma_{r(s)\,t} + \delta_t,\, \sigma^2) \\ \gamma_{rt} &\sim \begin{cases} \mathrm{N}(0, \tau^2_{\gamma 1}) & \text{for } r = 1, 2, 3 \quad \text{(non-South)} \\ \mathrm{N}(0, \tau^2_{\gamma 2}) & \text{for } r = 4 \qquad\quad \text{(South)} \end{cases} \\ \delta_t &\sim \mathrm{N}(0, \tau^2_\delta), \end{aligned} \tag{15.2}$$

with a uniform hyperprior distribution on $\beta, \sigma, \tau_{\gamma 1}, \tau_{\gamma 2}, \tau_\delta$. (We also fit with a uniform prior distribution on the hierarchical variances, rather than the standard deviations, and obtained essentially identical inferences.)

In the notation of Section 15.1, we would label β as the concatenated vector of all the varying coefficients, (β, γ, δ) in formulation (15.2). The augmented β has a prior with mean 0 and diagonal variance matrix $\Sigma_\beta = \mathrm{diag}(\infty, \ldots, \infty, \tau^2_{\gamma 1}, \ldots, \tau^2_{\gamma 1}, \tau^2_{\gamma 2}, \ldots, \tau^2_{\gamma 2}, \tau^2_\delta, \ldots, \tau^2_\delta)$. The first 20 elements of β, corresponding to the constant term and the predictors in Table 15.1, have noninformative priors—that is, $\sigma_{\beta j} = \infty$ for these elements. The next 33 values of $\sigma_{\beta j}$ are set to the parameter $\tau_{\gamma 1}$. The 11 elements corresponding to the Southern regional variables have $\sigma_{\beta j} = \tau_{\gamma 2}$. The final 11 elements correspond to the nationwide shifts and have prior standard deviation τ_δ.

Forecasting

Predictive inference is more subtle for a hierarchical model than a classical regression model, because of the possibility that new parameters (varying coefficients) β must be estimated for the predictive data. Consider the task of forecasting the outcome of the 1992 presidential election, given the 50×20 matrix of explanatory variables for the linear regression corresponding to the 50 states in 1992. To form the complete matrix of explanatory variables for 1992, $\tilde{X}$, we must include 55 columns of zeros, thereby setting the indicators for previous years to zero for estimating the results in 1992. Even then, we are not quite ready to make predictions. To simulate draws from the predictive distribution of 1992 state election results using the hierarchical model, we must include another year indicator and four new region × year indicator variables for 1992—this adds five new predictors. However, we have no information on the coefficients of these predictor variables; they are not even included in the vector β that we have estimated from the data up to 1988. Since we have no data on these five new components of β, we must simulate their values from their posterior (predictive) distribution; that is, the coefficient for the year indicator is drawn as $\mathrm{N}(0, \tau^2_\delta)$, the non-South region × year coefficients are drawn as $\mathrm{N}(0, \tau^2_{\gamma 1})$, and the South × year coefficient is drawn as $\mathrm{N}(0, \tau^2_{\gamma 2})$, using the values $\tau_\delta, \tau_{\gamma 1}, \tau_{\gamma 2}$ drawn from the posterior simulation.

Posterior inference

We fit the model using EM and the vector Gibbs sampler as described in Section 15.5, to obtain a set of draws from the posterior distribution of $(\beta, \sigma, \tau_\delta, \tau_{\gamma 1}, \tau_{\gamma 2})$. We ran ten

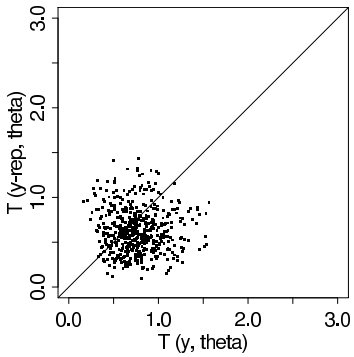


Figure 15.3 *Scatterplot showing the joint distribution of simulation draws of the realized test quantity, $T(y, \beta)$—the square root of the average of the 11 squared nationwide residuals—and its hypothetical replication, $T(y^{\text{rep}}, \beta)$, under the hierarchical model for the election forecasting example. The 200 simulated points are scattered evenly about the 45° line, which means that the model accurately fits this particular test quantity.*

parallel Gibbs sampler sequences; after 500 steps, the potential scale reductions, $\widehat{R}$, were below 1.1 for all parameters.

The coefficient estimates for the variables in Table 15.1 are similar to the results from the preliminary, nonhierarchical regression. The posterior medians of the coefficients all have the expected positive sign. The hierarchical standard deviations $\tau_\delta, \tau_{\gamma 1}, \tau_{\gamma 2}$ are not determined with great precision. This points out one advantage of the full Bayesian approach; if we had simply made point estimates of these variance components, we would have been ignoring a wide range of possible values for all the parameters.

When applied to data from 1992, the model yields state-by-state predictions that are summarized in Figure 6.1, with a forecasted 85% probability that the Democrats would win the national electoral vote total. The forecasts for individual states have predictive standard errors between 5% and 6%.

We tested the model in the same way as we tested the nonhierarchical model, by a posterior predictive check on the average of the squared nationwide residuals. The simulations from the hierarchical model, with their additional national and regional error terms, accurately fit the observed data, as shown in Figure 15.3.

Reasons for using a hierarchical model

In summary, there are three main advantages of the hierarchical model here:

- It allows the modeling of correlation within election years and regions.
- Including the year and region × year terms without a hierarchical model, or not including these terms at all, correspond to special cases of the hierarchical model with $\tau = \infty$ or 0, respectively. The more general model allows for a reasonable compromise between these extremes.
- Predictions will have additional components of variability for regions and year and should therefore be more reliable.

15.3 Interpreting a normal prior distribution as extra data

More general forms of the hierarchical linear model can be created, with further levels of parameters representing additional structure in the problem. For instance, building on the

brief example in the opening paragraph of this chapter, we might have a study of educational achievement in which class-level effects are not considered exchangeable but rather depend on features of the school district or state. In a similar vein, the election forecasting example might be extended to attempt some modeling of the year-by-year errors in terms of trends over time, although there is probably limited information on which to base such a model after conditioning on other observed variables. No serious conceptual or computational difficulties are added by extending the model to more levels.

A general formulation of a model with three levels of variation is:

$y\|X,\beta,\Sigma_y \sim \mathrm{N}(X\beta,\Sigma_y)$	'likelihood'	n data points y_i
$\beta\|X_\beta,\alpha,\Sigma_\beta \sim \mathrm{N}(X_\beta\alpha,\Sigma_\beta)$	'population distribution'	J parameters β_j
$\alpha\|\alpha_0,\Sigma_\alpha \sim \mathrm{N}(\alpha_0,\Sigma_\alpha)$	'hyperprior distribution'	K parameters α_k

Interpretation as a single linear regression

The conjugacy of prior distribution and regression likelihood (see Section 14.8) allows us to express the hierarchical model as a single normal regression model using a larger 'dataset' that includes as 'observations' the information added by the population and hyperprior distributions. Specifically, for the three-level model, we can extend (14.24) to write

$$y_*|X_*,\gamma,\Sigma_* \sim \mathrm{N}(X_*\gamma,\Sigma_*),$$

where γ is the vector (β,α) of length $J+K$, and y_*, X_*, and Σ_*^{-1} are defined by considering the likelihood, population, and hyperprior distributions as $n+J+K$ 'observations' informative about γ:

$$y_* = \begin{pmatrix} y \\ 0 \\ \alpha_0 \end{pmatrix}, \quad X_* = \begin{pmatrix} X & 0 \\ I_J & -X_\beta \\ 0 & I_K \end{pmatrix}, \quad \Sigma_*^{-1} = \begin{pmatrix} \Sigma_y^{-1} & 0 & 0 \\ 0 & \Sigma_\beta^{-1} & 0 \\ 0 & 0 & \Sigma_\alpha^{-1} \end{pmatrix}. \tag{15.3}$$

If any of the components of β or α have noninformative prior distributions, the corresponding rows in y_* and X_*, as well as the corresponding rows and columns in Σ_*^{-1}, can be eliminated, because they correspond to 'observations' with infinite variance. The resulting regression then has $n+J_*+K_*$ 'observations,' where J_* and K_* are the number of components of β and α, respectively, with informative prior distributions.

For example, in the election forecasting example, β has 75 components—20 predictors in the original regression (including the constant term but excluding the five regional variables in Table 15.1 associated with specific years), 11 year errors, and 44 region × year errors—but J_* is only 55 because only the year and region × year errors have informative prior distributions. (All three groups of varying coefficients have means fixed at zero, so in this example $K_*=0$.)

More than one way to set up a model

A hierarchical regression model can be set up in several equivalent ways. For example, we have already noted that the hierarchical model for the 8 schools could be written as $y_j \sim \mathrm{N}(\theta_j,\sigma_j^2)$ and $\theta_j \sim \mathrm{N}(\mu,\tau^2)$, or as $y_j \sim \mathrm{N}(\beta_0+\beta_j,\sigma_j^2)$ and $\beta_j \sim \mathrm{N}(0,\tau^2)$. The hierarchical model for the election forecasting example can be written, as described above, as a regression, $y \sim \mathrm{N}(X\beta,\sigma^2 I)$, with 70 predictors X and normal prior distributions on the coefficients β, or as a regression with 20 predictors and three independent error terms, corresponding to year, region × year, and state × year.

In the three-level formulation described at the beginning of this section, group-level predictors can be included in either the likelihood or the population distribution, and the constant term can be included in any of the three regression levels.

15.4 Varying intercepts and slopes

So far we have focused on hierarchical models for scalar parameters. What do we do when multiple parameters can vary by group? Then we need a multivariate prior distribution. Consider the following generic model of data in J groups and, within each group j, a likelihood, $p(y^{(j)}|\theta^{(j)})$, depending on a vector of parameters θ, which themselves are given a prior distribution, $p(\theta_j|\phi)$, given hyperparameters ϕ.

The model can also have parameters that do not vary by group, for example in the varying-intercept, varying-slope linear model:

$$\begin{aligned} y_{ij} &\sim \text{N}(\alpha_j + x_{ij}\beta_j, \sigma_y^2) \\ \begin{pmatrix} \alpha \\ \beta \end{pmatrix} &\sim \text{N}\left(\begin{pmatrix} \mu_\alpha \\ \mu_\beta \end{pmatrix}, \begin{pmatrix} \sigma_\alpha^2 & \rho\sigma_\alpha\sigma_\beta \\ \rho\sigma_\alpha\sigma_\beta & \sigma_\beta^2 \end{pmatrix} \right). \end{aligned} \tag{15.4}$$

We then assign a hyperprior distribution to the vector of hyperparameters, $(\mu_\alpha, \mu_\beta, \sigma_\alpha, \sigma_\beta, \rho)$, probably starting with a uniform (with the constraints that the scale parameters must be positive and the correlation parameter is between -1 and 1) and then adding more information as necessary (for example, if the number of groups is low).

When more than two coefficients vary by group, we can write (15.4) in vector-matrix form as,

$$\begin{aligned} y_{ij} &\sim \text{N}(X_j\beta^{(j)}, \sigma_y^2) \\ \beta^{(j)} &\sim \text{N}(\mu_\beta, \Sigma_\beta). \end{aligned} \tag{15.5}$$

The model could be further elaborated, for example by having the data variance itself vary by group, or by adding structure to the mean vector, or by having more than one level of grouping. In any case, the key is the prior distribution on Σ_β. (The data variance σ_y and the mean vector μ_β also need hyperprior distributions, but these parameters can typically be estimated pretty well from the data, so their exact prior specification is not so important.)

Inverse-Wishart model

Let K be the number of coefficients in the regression model, so that β is a $J \times K$ matrix, and the group-level variance matrix Σ_β is $K \times K$. The conditionally conjugate distribution for Σ_β in model (15.5) is the inverse-Wishart (see Section 3.6).

Scaled inverse-Wishart model

The trouble with the Inv-Wishart$_{K+1}(I)$ model is that it strongly constrains the variance parameters, the diagonal elements of the covariance matrix. What we want is a model that is noninformative on the correlations but allows a wider range of uncertainty on the variances. We can create such a model using a redundant parameterization, rewriting (15.5) as,

$$\begin{aligned} y_{ij} &\sim \text{N}(X_j(\mu_\beta + \xi \otimes \eta^{(j)}), \sigma_y^2) \\ \eta^{(j)} &\sim \text{N}(0, \Sigma_\eta). \end{aligned} \tag{15.6}$$

Here, ξ is a vector of length K, the symbol $\otimes$ represents component-wise multiplication, we have decomposed the coefficient vector from each group as $\beta^{(j)} = \mu_\beta + \xi \otimes \eta$, and the covariance matrix is

$$\Sigma_\beta = \text{Diag}(\xi)\Sigma_\eta\text{Diag}(\xi).$$

The advantage of splitting up the model in this way is that now we can assign separate prior distributions to ξ and Σ_η, inducing a richer structure of uncertainty modeling. It is, in fact, an example of the scaled inverse-Wishart model described at the end of Section 3.6.

As a noninformative model, we can set $\Sigma_\eta \sim \text{Inv-Wishart}_{K+1}(I)$, along with independent uniform prior distributions on ξ's. If necessary, one can add informative priors to ξ.

Predicting business school grades for different groups of students

We illustrate varying-intercept, varying-slope regression with an example of prediction in which the data are so sparse that a hierarchical model is *necessary* for inference about some of the estimands of interest.

It is common for schools of business management in the United States to use regression equations to predict the first-year grade point average of prospective students from their scores on the verbal and quantitative portions of the Graduate Management Admission Test (GMAT-V and GMAT-Q) as well as their undergraduate grade point average (UGPA). This equation is important because the predicted score derived from it may play a central role in the decision to admit the student. The coefficients of the regression equation are typically estimated from the data collected from the most recently completed first-year class.

A concern was raised with this regression model about possible biased predictions for identifiable subgroups of students, particularly black students. A study was performed based on data from 59 business schools over a two-year period, involving about 8500 students of whom approximately 4% were black. For each school, a separate regression was performed of first-year grades on four explanatory variables: a constant term, GMAT-V, GMAT-Q, and UGPA. By looking at the residuals for all schools and years, it was found that the regressions tended to *overpredict* the first-year grades of blacks.

At this point, it might seem natural to add another term to the regression model corresponding to an indicator variable that is 1 if a student is black and 0 otherwise. However, such a model was considered too restrictive; once blacks and non-blacks were treated separately in the model, it was desired to allow different regression models for the two groups. For each school, the expanded model then has eight explanatory variables: the four mentioned above, and then the same four variables multiplied by the indicator for black students. For student $i = 1, \dots, n_j$ in school $j = 1, \dots, 59$ we model the first-year grade point average y_{ij}, given the vector of eight covariates x_{ij}, as a linear regression with coefficient vector β_j and residual variance σ_j^2. Then the model for the entire vector of responses y is

$$p(y|\beta, \sigma^2) \sim \prod_{j=1}^{59} \prod_{i=1}^{n_j} \text{N}(y_{ij}|X_{ij}\beta_j, \sigma_j^2).$$

Geometrically, the model is equivalent to requiring two different regression planes: one for blacks and one for non-blacks. For each school, nine parameters must be estimated: $\beta_{1j}, \dots, \beta_{8j}, \sigma_j$. Algebraically, all eight terms of the regression are used to predict the scores of blacks but only the first four terms for non-blacks.

At this point, the procedure of estimating separate regressions for each school becomes impossible using standard least-squares methods, which are implicitly based on noninformative prior distributions. Blacks comprise only 4% of the students in the dataset, and many of the schools are all non-black or have so few blacks that the regression parameters cannot be estimated under classical regression (that is, based on a noninformative prior distribution on the nine parameters in each regression). Fortunately, it is possible to estimate all 8×59 regression parameters simultaneously using a hierarchical model. To use the most straightforward approach, the 59 vectors β_j are modeled as independent samples from a multivariate $\text{N}(\alpha, \Lambda_\beta)$ distribution, with unknown vector α and 8×8 matrix Λ_β.

The unknown parameters of the model are then β, α, Λ_β, and $\sigma_1, \dots, \sigma_{59}$. We first

assumed a uniform prior distribution on $\alpha, \Lambda_\beta, \log\sigma_1, \ldots, \log\sigma_{59}$. This noninformative approach is not ideal (at the least, one would want to embed the 59 σ_j parameters in a hierarchical model) but is a reasonable start.

A crude approximation to α and the parameters σ_j^2 was obtained by running a regression of the combined data vector y on the eight explanatory variables, pooling the data from all 59 schools. Using the crude estimates as a starting point, the posterior mode of $(\alpha, \Lambda_\beta, \sigma_1^2, \ldots, \sigma_{59}^2)$ was found using EM. Here, we describe the conclusions of the study, which were based on this modal approximation.

One conclusion from the analysis was that the multivariate hierarchical model is a substantial improvement over the standard model, because the predictions for both black and non-black students are relatively accurate. Moreover, the analysis revealed systematic differences between predictions for black and non-black students. In particular, conditioning the test scores at the mean scores for the black students, in about 85% of schools, non-blacks were predicted to have higher first-year grade-point averages, with over 60% of the differences being more than one posterior standard deviation above zero, and about 20% being more than two posterior standard deviations above zero. This sort of comparison, conditional on school and test scores, could not be reasonably estimated with a nonhierarchical model in this dataset, in which the number of black students per school was so low.

15.5 Computation: batching and transformation

There are several ways to use the Gibbs sampler to draw posterior simulations for hierarchical linear models. The different methods vary in their programming effort required, computational speed, and convergence rate, with different approaches being reasonable for different problems. In general we prefer Hamiltonian Monte Carlo to simple Gibbs; however, ideas of Gibbs sampling remain relevant, both for parameterizing the problem so that HMC runs more effectively, and because for large problems it can make sense to break up a hierarchical model into smaller pieces for more efficient computation.

We shall discuss computation for models with independent variances at each of the hierarchical levels; computation can be adapted to structured covariance matrices as described in Section 14.7.

Gibbs sampler, one batch at a time

Perhaps the simplest computational approach for hierarchical regressions is to perform a blockwise Gibbs sampler, updating each batch of regression coefficients given all the others, and then updating the variance parameters. Given the data and all other parameters in the model, inference for a batch of coefficients corresponds to a linear regression with fixed prior distribution. We can thus update the coefficients of the entire model in batches, performing at each step an augmented linear regression, as discussed in Section 14.8. In many cases (including the statewide, regional, and national error terms in the election forecasting example), the model is simple enough that the means and variances for the Gibbs updating do not actually require a regression calculation but instead can be performed using simple averages.

The variance parameters can also be updated one at a time. For simple hierarchical models with scalar variance parameters (typically one parameter per batch of coefficients, along with one or more data-level variance parameters), the Gibbs updating distributions are scaled inverse-χ^2. For more complicated models, the variance parameters can be updated using Metropolis jumps.

The Gibbs sampler for the entire model then proceeds by cycling through all the batches of parameters in the model (including the batch, if any, of coefficients with noninformative

or fixed prior distributions). One attractive feature of this algorithm is that it mimics the natural way one might try to combine information from the different levels: starting with a guess at the upper-level parameters, the lower-level regression is run, and then these simulated coefficients are used to better estimate the upper-level regression parameters, and so on.

All-at-once Gibbs sampler

As noted in Section 15.3, the different levels in a hierarchical regression context can be combined into a single regression model by appropriately augmenting the data, predictors, and variance matrix. The Gibbs sampler can then be applied, alternately updating the variance parameters (with independent inverse-χ^2 distributions given the data and regression coefficients), and the vector of regression coefficients, which can be updated at each step by running a weighted regression with weight matrix depending on the current values of the variance parameters.

In addition to its conceptual simplicity, this all-at-once Gibbs sampler has the advantages of working efficiently even if regression coefficients at different levels are correlated in their posterior distribution, as can commonly happen with hierarchical models (for example, the parameters θ_j and μ have positive posterior correlations in the 8-schools example).

The main computational disadvantage of all-at-once Gibbs sampling for this problem is that each step requires a regression on a potentially large augmented dataset. For example, the election forecasting model has $n = 511$ data points and $k = 20$ predictors, but the augmented regression has $n_* = 511 + 55$ observations and $k_* = 20 + 55$ predictors. The computer time required to perform a linear regression is proportional to nk^2, and thus each step of the all-at-once Gibbs sampler can be slow when the number of parameters is large. In practice we would implement such a computation using HMC rather than Gibbs.

Parameter expansion

Any of the algorithms mentioned above can be slow to converge when estimated hierarchical variance parameters are near zero. The problem is that, if the current draw of a hierarchical variance parameter is near 0, then in the next updating step, the corresponding batch of linear parameters γ_j will themselves be 'shrunk' to be close to their population mean. Then, in turn, the variance parameter will be estimated to be close to 0 because it is updated based on the relevant γ_j's. Ultimately, the stochastic nature of the Gibbs sampler allows it to escape out of this trap but this may require many iterations.

The parameter-expanded Gibbs sampler and EM algorithms (see Sections 12.1 and 13.4) can be used to solve this problem. For hierarchical linear models, the basic idea is to associate with each batch of regression coefficients a multiplicative factor, which has the role of breaking the dependence between the coefficients and their variance parameter.

Example. Election forecasting (continued)
We illustrate with the presidential election forecasting model (15.2), which in its expanded-parameter version can be written as,

$$y_{st} \sim \begin{cases} \text{N}(X_{st}\beta + \zeta_1^{\text{region}}\gamma_{r(s)\,t} + \zeta^{\text{year}}\delta_t,\, \sigma^2) & \text{if } r(s) = 1, 2, 3 \quad \text{(non-South)} \\ \text{N}(X_{st}\beta + \zeta_2^{\text{region}}\gamma_{r(s)\,t} + \zeta^{\text{year}}\delta_t,\, \sigma^2) & \text{if } r(s) = 4 \qquad\quad \text{(South)} \end{cases}$$

with the same prior distributions as before. The new parameters ζ_1^{region}, ζ_2^{region}, and ζ^{year} are assigned uniform prior distributions and are not identified in the posterior distribution. The products $\zeta_1^{\text{region}}\gamma_{rt}$ (for $r = 1, 2, 3$), $\zeta_2^{\text{region}}\gamma_{rt}$ (for $r = 4$), and $\zeta^{\text{year}}\delta_t$, correspond to the parameters γ_{rt} and δ_t, respectively, in the old model. Similarly, the

products $|\zeta_1^{\text{region}}|\tau_{\gamma_1}$, $|\zeta_2^{\text{region}}|\tau_{\gamma_2}$, and $|\zeta^{\text{year}}|\tau_\delta$ correspond to the variance components τ_{γ_1}, τ_{γ_2}, and τ_δ in the original model.

For the election forecasting model, the variance parameters are estimated precisely enough that the ordinary Gibbs sampler performs reasonably efficiently. However, we can use this problem to illustrate the computational steps of parameter expansion.

The Gibbs sampler for the parameter-expanded model alternately updates the regression coefficients (β, γ, δ), the variance parameters $(\sigma, \tau_{\gamma 1}, \tau_{\gamma 2}, \tau_\delta)$, and the additional parameters $(\zeta_1^{\text{region}}, \zeta_2^{\text{region}}, \zeta^{\text{year}})$. The regression coefficients can be updated using any of the Gibbs samplers described above—by batch, all at once, or one element at a time. The ζ parameters do not change the fact that β, γ, and δ can be estimated by linear regression. Similarly, the Gibbs sampler updates for the variances are still independent inverse-χ^2.

The final step of the Gibbs sampler is to update the multiplicative parameters ζ. This step turns out to be easy: given the data and the other parameters in the model, the information about the ζ's can be expressed simply as a linear regression of the 'data' $z_{st} = y_{st} - X_{st}\beta$ on the 'predictors' $\gamma_{r(s)\,t}$ (for states s with regions $r(s) = 1$, 2, or 3), $\gamma_{r(s)\,t}$ (for $r(s) = 4$), and δ_t, with variance matrix $\sigma^2 I$. The three parameters ζ are then easily updated with a linear regression.

When running the Gibbs sampler, we do not worry about the individual parameters $\zeta, \beta, \gamma, \delta$; instead, we save and monitor the convergence of the variance parameters σ and the parameters γ_{rt} and δ_t in the original parameterization (15.2). This is most easily done by just multiplying each of the parameters γ_{rt} and δ_t by the appropriate ζ parameter, and multiplying each of the variance components $\tau_{\gamma 1}, \tau_{\gamma 2}, \tau_\delta$, by the absolute value of its corresponding ζ.

More on the parameter-expanded Gibbs sampler for hierarchical models appears at the end of Section 15.7.

Transformations for HMC

A slightly different transformation can be useful when implementing Hamiltonian Monte Carlo. HMC can be slow to converge when certain parameters have very short-tailed or long-tailed distributions, and this can happen with the variance parameters or the log variance parameters in hierarchical models. A simple decoupling of the model can sometimes solve this problem.

Example. Eight schools model

We demonstrate with the hierarchical model for the educational testing experiments from Section 5.5:

$$\begin{aligned} y_j &\sim \text{N}(\theta_j, \sigma_j^2), \text{ for } j = 1, \ldots, J \\ \theta_j &\sim \text{N}(\mu, \tau^2), \text{ for } j = 1, \ldots, J, \end{aligned}$$

along with some prior distribution, $p(\mu, \tau)$. EM or Gibbs for this model can get stuck when the value of τ is near zero. In HMC, the joint posterior distribution of θ, μ, τ forms a 'whirlpool': no single step size works well for the whole distribution, and, again, the trajectories are unlikely to go into the region where τ is near zero and then unlikely to leave that vortex when they are there.

The following parameterization works much better:

$$\begin{aligned} y_j &\sim \text{N}(\mu + \tau\eta_j, \sigma_j^2), \text{ for } j = 1, \ldots, J \\ \eta_j &\sim \text{N}(0, 1), \text{ for } j = 1, \ldots, J, \end{aligned}$$

where $\theta_j = \mu + \tau\eta_j$ for each j.

15.6 Analysis of variance and the batching of coefficients

The largest gains in estimating regression coefficients often come from specifying structure in the model. For example, in the election forecasting problem, it is crucial that the national and regional indicator variables are clustered and modeled separately from the quantitative predictors. In general, when many predictor variables are used in a regression, they should be set up so they can be structured hierarchically, so the Bayesian analysis can do the most effective job at pooling the information about them.

Analysis of variance (Anova) represents a key idea in statistical modeling of complex data structures—the grouping of predictor variables and their coefficients into batches. In the traditional application of analysis of variance, each batch of coefficients and the associated row of the Anova table corresponds to a single experimental block or factor or to an interaction of two or more factors. In a hierarchical linear regression context, each row of the table corresponds to a set of regression coefficients, and we are potentially interested in the individual coefficients and also in the variance of the coefficients in each batch. We thus view the analysis of variance as a way of making sense of hierarchical regression models with many predictors or indicators that can be grouped into batches within which all the coefficients are exchangeable.

Notation and model

We shall work with the linear model, with the 'analysis of variance' corresponding to the batching of coefficients into 'sources of variation,' with each batch corresponding to one row of an Anova table. We use the notation $m = 1, \ldots, M$ for the rows of the table. Each row m represents a batch of J_m regression coefficients $\beta_j^{(m)},\ j = 1, \ldots, J_m$. We denote the m-th subvector of coefficients as $\beta^{(m)} = (\beta_1^{(m)}, \ldots, \beta_{J_m}^{(m)})$ and the corresponding classical least-squares estimate as $\hat{\beta}^{(m)}$. These estimates are subject to c_m linear constraints, yielding $(df)_m = J_m - c_m$ degrees of freedom. We label the $c_m \times J_m$ constraint matrix as $C^{(m)}$, so that $C^{(m)}\hat{\beta}^{(m)} = 0$ for all m, and we assume that $C^{(m)}$ is of rank c_m. For notational convenience, we label the grand mean as $\beta_1^{(0)}$, corresponding to the (invisible) zeroth row of the Anova table and estimated with no linear constraints.

The linear model is fitted to the data points $y_i,\ i = 1, \ldots, n$, and can be written as

$$y_i = \sum_{m=0}^{M} \beta_{j_i^m}^{(m)}, \tag{15.7}$$

where j_i^m indexes the appropriate coefficient j in batch m corresponding to data point i. Thus, each data point pulls one coefficient from each row in the Anova table. Equation (15.7) could also be expressed as a linear regression model with a design matrix composed entirely of 0's and 1's. The coefficients β_j^M of the last row of the table correspond to the residuals or error term of the model.

The analysis of variance can also be applied more generally to regression models (or to generalized linear models), in which case we can have any design matrix X, and (15.7) is replaced by

$$y_i = \sum_{m=0}^{M} \sum_{j=1}^{J_m} x_{ij}^{(m)} \beta_j^{(m)}. \tag{15.8}$$

The essence of analysis of variance is in the structuring of the coefficients into batches—hence the notation $\beta_j^{(m)}$—going beyond the usual linear model formulation that has a single indexing of coefficients β_j.

We shall use a hierarchical formulation in which each batch of regression coefficients is

modeled as a sample from a normal distribution with mean 0 and its own variance σ_m^2:

$$\beta_j^{(m)} \sim \text{N}(0, \sigma_m^2), \quad \text{for } j = 1, \ldots, J_m, \quad \text{for each batch } m = 1, \ldots, M.$$

Without loss of generality, we can center the distribution of each batch $\beta^{(m)}$ at 0—if it were appropriate for the mean to be elsewhere, we would just include the mean of the $x_j^{(m)}$'s as a separate predictor. As in classical Anova, we usually include interactions only if the corresponding main effects are in the model.

The conjugate hyperprior distributions for the variances are scaled inverse-χ^2 distributions:

$$\sigma_m^2 \sim \text{Inv-}\chi^2(\nu_m, \sigma_{0m}^2).$$

A natural noninformative prior density is uniform on σ_m, which corresponds to $\nu_m = -1$ and $\sigma_{0m} = 0$. For values of m in which J_m is large (that is, rows of the Anova table corresponding to many linear predictors), σ_m is essentially estimated from data. When J_m is small, the flat prior distribution implies that σ is allowed the possibility of taking on large values, which minimizes the amount of shrinkage in the coefficient estimates.

Computation

In this model, the posterior distribution for the parameters (β, σ) can be simulated using the Gibbs sampler, alternately updating the vector β given σ with linear regression, and updating the vector σ from the independent inverse-χ^2 conditional posterior distributions given β.

The only trouble with this Gibbs sampler is that it can get stuck with variance components σ_m near zero. A more efficient updating uses parameter expansion, as described at the end of Section 15.5. In the notation here, we reparameterize into vectors γ, ζ, and τ, which are defined as follows:

$$\begin{aligned} \beta_j^{(m)} &= \zeta_m \gamma_j^{(m)} \\ \sigma_m &= |\zeta_m| \tau_m. \end{aligned} \tag{15.9}$$

The model can be then expressed as

$$\begin{aligned} y &= X\zeta\gamma \\ \gamma_j^{(m)} &\sim \text{N}(0, \tau_m^2) \quad \text{for each } m \\ \tau_m^2 &\sim \text{Inv-}\chi^2(\nu_m, \sigma_{0m}^2). \end{aligned}$$

The auxiliary parameters ζ are given a uniform prior distribution, and then this reduces to the original model. The Gibbs sampler then proceeds by updating γ (using linear regression with n data points and $\sum_{m=0}^{M} J_m$ predictors), ζ (linear regression with n data points and M predictors), and τ (independent inverse-χ^2 distributions). The parameters in the original parameterization, β and σ, can then be recomputed from (15.9) and stored at each step.

Finite-population and superpopulation standard deviations

One measure of the importance of each row or 'source' in the Anova table is the standard deviation of its constrained regression coefficients, defined as

$$s_m = \sqrt{\frac{1}{(df)_m} \beta^{(m)\,T} \left[I - C^{(m)\,T} (C^{(m)} C^{(m)\,T})^{-1} C^{(m)} \right] \beta^{(m)}}. \tag{15.10}$$

We divide by $(df)_m = J_m - c_m$ rather than $J_m - 1$ because multiplying by $C^{(m)}$ induces c_m linear constraints. We model the underlying β coefficients as unconstrained.

For each batch of coefficients $\beta^{(m)}$, there are two natural variance parameters to estimate: the *superpopulation* standard deviation σ_m and the *finite-population* standard deviation s_m as defined in (15.10). The superpopulation standard deviation characterizes the uncertainty for predicting a new coefficient from batch m, whereas the finite-population standard deviation describes the variation in the existing J_m coefficients.

Variance estimation is often presented in terms of the superpopulation standard deviations σ_m, but in our Anova summaries, we focus on the finite-population quantities s_m, for reasons we shall discuss here. However, for computational reasons, the parameters σ_m are useful intermediate quantities to estimate. Our general procedure is to use computational methods such as described in Section 15.5 to draw joint posterior simulations of (β, σ) and then compute the finite-sample standard deviations s_m from β using (15.10).

To see the difference between the two variances, consider the extreme case in which $J_m = 2$ (with the usual constraint that $\beta_1^{(m)} + \beta_2^{(m)} = 0$) and a large amount of data are available in both groups. Then the two parameters $\beta_1^{(m)}$ and $\beta_2^{(m)}$ will be estimated accurately and so will $s_m^2 = \frac{1}{2}(\beta_1^{(m)} - \beta_2^{(m)})^2$. The superpopulation variance σ_m^2, on the other hand, is only being estimated by a measurement that is proportional to a χ^2 with 1 degree of freedom. We know much about the two parameters $\beta_1^{(m)}, \beta_2^{(m)}$ but can say little about others from their batch.

We believe that much of the statistical literature on fixed and random effects can be fruitfully reexpressed in terms of finite-population and superpopulation inferences. In some contexts (for example, collecting data on the 50 states of the U.S.), the finite population seems more meaningful; whereas in others (for example, subject-level effects in a psychological experiment), interest clearly lies in the superpopulation.

For example, suppose a factor has four degrees of freedom corresponding to five different medical treatments, and these are the only existing treatments and are thus considered 'fixed.' Suppose it is then discovered that these are part of a larger family of many possible treatments, and so it makes sense to model them as 'random.' In our framework, the inference about these five parameters $\beta_j^{(m)}$ and their finite-population and superpopulation standard deviations, s_m and σ_m, will not change with the news that they can actually be viewed as a random sample from a distribution of possible treatment effects. But the superpopulation variance now has an important new role in characterizing this distribution. The difference between fixed and random effects is thus not a difference in inference or computation but in the ways that these inferences will be used.

Example. Five-way factorial structure for data on Web connect times

We illustrate the analysis of variance with an example of a linear model fitted for exploratory purposes to a highly structured dataset. Data were collected by an internet infrastructure provider on connect times for messages processed by two different companies. Messages were sent every hour for 25 consecutive hours, from each of 45 locations to 4 different destinations, and the study was repeated one week later. It was desired to quickly summarize these data to learn about the importance of different sources of variation in connect times.

Figure 15.4 shows the Bayesian Anova display for an analysis of logarithms of connect times on the five factors: destination ('to'), source ('from'), service provider ('company'), time of day ('hour'), and week. The data have a full factorial structure with no replication, so the full five-way interaction at the bottom represents the 'error' or lowest-level variability.

Each row of the plot shows the estimated finite-population standard deviation of the corresponding group of parameters, along with 50% and 95% uncertainty intervals. We can immediately see that the lowest-level variation is more important in variance than

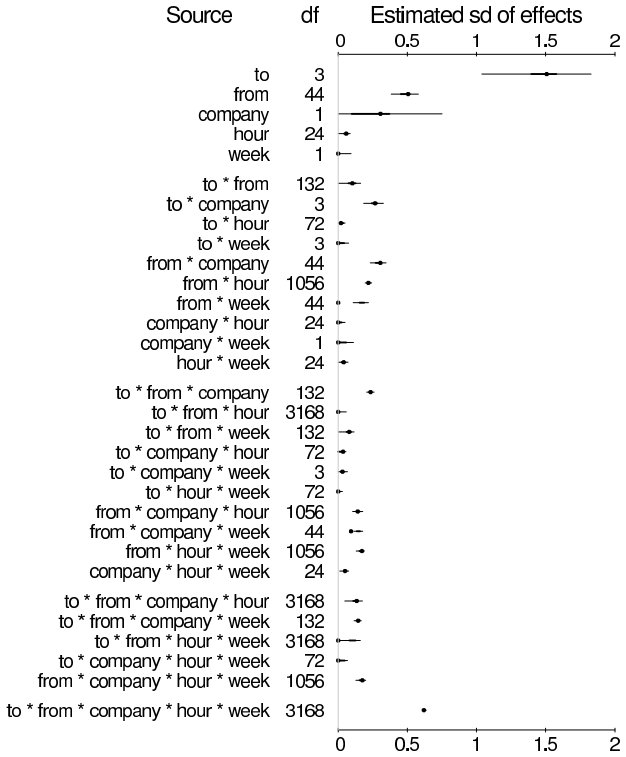


Figure 15.4 *Anova display for the World Wide Web data. The bars indicate 50% and 95% intervals for the finite-population standard deviations* s_m. *The display makes apparent the magnitudes and uncertainties of the different components of variation. Since the data are on the logarithmic scale, the standard deviation parameters can be interpreted directly. For example,* $s_m = 0.20$ *corresponds to a coefficient of variation of* $\exp(0.2) - 1 \approx 0.2$ *on the original scale, and so the exponentiated coefficients* $\exp(\beta_j^{(m)})$ *in this batch correspond to multiplicative increases or decreases in the range of 20%. (The dots on the bars show simple classical estimates of the variance components that can be used as starting points in a Bayesian computation.)*

any of the factors except for the main effect of the destination. `Company` represents a large amount of variation on its own and, perhaps more interestingly, in interaction with `to`, `from`, and in the three-way interaction.

Figure 15.4 would not normally represent the final statistical analysis for this sort of problem. The Anova plot represents a default model and is a tool for data exploration—for learning about which factors are important in predicting the variation in the data—and can be used to construct more focused models or design future data collection.

15.7 Hierarchical models for batches of variance components

We next consider an analysis of variance problem which has several variance components, one for each source of variation, in a $5 \times 5 \times 2$ split-plot latin square with five full-plot treatments (labeled A, B, C, D, E), and with each plot divided into two subplots (labeled 1 and 2).

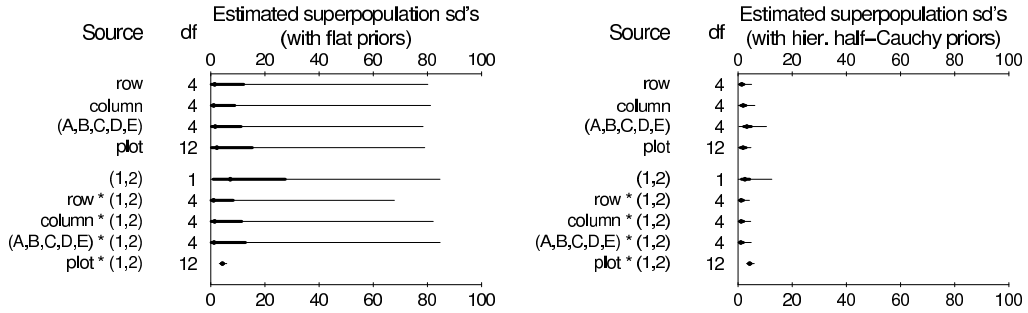


Figure 15.5 *Posterior medians, 50%, and 95% intervals for standard deviation parameters σ_k estimated from a split-plot latin square experiment. (a) The left plot shows inferences given uniform prior distributions on the σ_k's. (b) The right plot shows inferences given a hierarchical half-Cauchy model with scale fit to the data. The half-Cauchy model gives much sharper inferences, using the partial pooling that comes with fitting a hierarchical model.*

Source	df
row	4
column	4
(A,B,C,D,E)	4
plot	12
(1,2)	1
row × (1,2)	4
column × (1,2)	4
(A,B,C,D,E) × (1,2)	4
plot × (1,2)	12

Each row of the table corresponds to a different variance component, and the split-plot Anova can be understood as a linear model with nine variance components, $\sigma_1^2, \ldots, \sigma_9^2$—one for each row of the table. A simple noninformative analysis uses a uniform prior distribution, $p(\sigma_1, \ldots, \sigma_9) \propto 1$.

More generally, we can set up a hierarchical model, where the variance parameters have a common distribution with hyperparameters estimated from the data. Based on the analyses given above, we consider a half-Cauchy prior distribution with peak 0 and scale A, and with a uniform prior distribution on A. The hierarchical half-Cauchy model allows most of the variance parameters to be small but with the occasionally large σ_α, which seems reasonable in the typical settings of analysis of variance, in which most sources of variation are small but some are large.

Superpopulation and finite-population standard deviations

Figure 15.5 shows the inferences in the latin square example, given uniform and hierarchical half-Cauchy prior distributions for the standard deviation parameters σ_k. As the left plot shows, the uniform prior distribution does not rule out the potential for some extremely high values of the variance components—the degrees of freedom are low, and the interlocking of the linear parameters in the latin square model results in difficulty in estimating any single variance parameter. In contrast, the hierarchical half-Cauchy model performs a great deal of shrinkage, especially of the high ranges of the intervals. (For most of the variance parameters, the posterior medians are similar under the two models; it is the 75th and 97.5th percentiles that are shrunk by the hierarchical model.) This is an ideal setting for hierarchical modeling of variance parameters in that it combines separately imprecise estimates of each of the individual σ_k's.

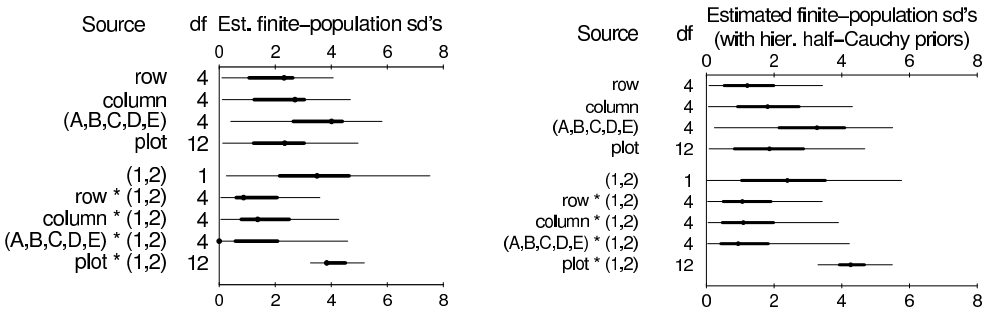


Figure 15.6 *Posterior medians, 50%, and 95% intervals for finite-population standard deviations s_k estimated from a split-plot latin square experiment. (a) The left plot shows inferences given uniform prior distributions on the σ_k's. (b) The right plot shows inferences given a hierarchical half-Cauchy model with scale fit to the data. The half-Cauchy model gives sharper estimates even for these finite-population standard deviations, indicating the power of hierarchical modeling for these highly uncertain quantities. Compare to Figure 15.5 (which is on a different scale).*

The σ_k's are *superpopulation* parameters in that each represents the standard deviation for an entire population of groups, of which only a few were sampled for the experiment at hand. In estimating variance parameters from few degrees of freedom, it can be helpful also to look at the *finite-population* standard deviation s_α of the corresponding linear parameters α_j.

For a simple hierarchical model of the form (5.21), s_α is simply the standard deviation of the J values of α_j. More generally, for more complicated linear models such as the split-plot latin square, s_α for any variance component is the root mean square of the coefficients' residuals after projection to their constraint space. In any case, this finite-population standard deviation s can be calculated from its posterior simulations and, especially when degrees of freedom are low, is more precisely estimated than the superpopulation standard deviation σ.

Figure 15.6 shows posterior inferences for the finite-population standard deviation parameters s_α for each row of the latin square split-plot Anova, showing inferences given the uniform and hierarchical half-Cauchy prior distributions for the variance parameters σ_α. The half-Cauchy prior distribution does slightly better than the uniform, with the largest shrinkage occurring for the variance component that has just one degree of freedom. The Cauchy scale parameter A was estimated at 1.8, with a 95% posterior interval of $[0.5, 5.1]$.

15.8 Bibliographic note

Gelman and Hill (2007) present a thorough elementary level introduction to statistical modeling with hierarchical linear models. Novick et al. (1972) describe an early application of Bayesian hierarchical regression. Lindley and Smith (1972) present the general form for the normal linear model (using a slightly different notation than ours); see also Hodges (1998). Many interesting applications of Bayesian hierarchical regression have appeared in the statistical literature since then; for example, Fearn (1975) analyzes growth curves, Hui and Berger (1983) and Strenio, Weisberg, and Bryk (1983) estimate patterns in longitudinal data, and Normand, Glickman, and Gatsonis (1997) analyze death rates in a set of hospitals. The business school prediction example at the end of Section 15.4 is taken from Braun et al. (1983), who perform the approximate Bayesian inference described in the text. Rubin (1980b) presents a hierarchical linear regression in an educational example and goes into

some detail on the advantages of the hierarchical approach. Other references on hierarchical linear models appear at the end of Chapter 5.

Random-effects or varying-coefficients regression has a long history in the non-Bayesian statistical literature; for example, see Henderson et al. (1959). Robinson (1991) provides a review, using the term 'best linear unbiased prediction'.[1] The Bayesian approach differs by averaging over uncertainty in the posterior distribution of the hierarchical parameters, which is important in problems such as the educational testing example of Section 5.5 with large posterior uncertainty in the hierarchical variance parameter.

Prior distributions and Bayesian inference for the covariance matrix of a multivariate normal distribution are discussed in Leonard and Hsu (1992), Yang and Berger (1994), Daniels and Kass (1999, 2001), and Barnard, McCulloch, and Meng (2000). Each of the above works on a different parameterization of the covariance matrix. Wong, Carter, and Kohn (2002) discuss prior distributions for the inverse covariance matrix. Verbeke and Molenberghs (2000) and Daniels and Pourahmadi (2002) discuss hierarchical linear models for longitudinal data.

Tokuda et al. (2011) present some methods for visualizing prior distributions for covariance matrices.

Hierarchical linear modeling has recently gained in popularity, especially in the social sciences, where it is often called *multilevel modeling*. An excellent summary of these applications at a fairly elementary level is provided by Raudenbush and Bryk (2002). Other texts in this area include Kreft and DeLeeuw (1998) and Snijders and Bosker (1999). Leyland and Goldstein (2001) provide an overview of multilevel models for public health research. Cressie et al. (2009) discuss hierarchical models in ecology.

Other key references on multilevel models in social science are Goldstein (1995), Longford (1993), and Aitkin and Longford (1986); the latter article is an extended discussion of the practical implications of undertaking a detailed hierarchical modeling approach to controversial issues in school effectiveness studies in the United Kingdom. Sampson, Raudenbush, and Earls (1997) discuss a study of crime using a hierarchical model of city neighborhoods.

Gelman and King (1993) discuss the presidential election forecasting problem in more detail, with references to earlier work in the econometrics and political science literature. Much has been written on election forecasating; see, for example, Rosenstone (1984) and Hibbs (2008). Boscardin and Gelman (1996) provide details on computations, inference, and model checking for the model described in Section 15.2 and some extensions.

Gelfand, Sahu, and Carlin (1995) discuss linear transformations for Gibbs samplers in hierarchical regressions, Liu and Wu (1999) and Gelman et al. (2008) discuss the parameter-expanded Gibbs sampler for hierarchical linear and generalized linear models. Pinheiro and Bates present an approach to computing hierarchical models by integrating over the linear parameters.

Much has been written on Bayesian methods for estimating many regression coefficients, almost all from the perspective of treating all the coefficients in a problem as exchangeable. Ridge regression (Hoerl and Kennard, 1970) is a procedure equivalent to an exchangeable normal prior distribution on the coefficients, as has been noted by Goldstein (1976), Wahba (1978), and others. Leamer (1978a) discusses the implicit models corresponding to stepwise regression and some other methods. George and McCulloch (1993) propose an exchangeable bimodal prior distribution for regression coefficients. Madigan and Raftery (1994) propose an approximate Bayesian approach for averaging over a distribution of potential regression models. Clyde, DeSimone, and Parmigiani (1996) and West (2003) present Bayesian

[1] Posterior means of regression coefficients and 'random effects' from hierarchical models are biased 'estimates' but can be unbiased or approximately unbiased when viewed as 'predictions,' since conventional frequency evaluations condition on all unknown 'parameters' but not on unknown 'predictive quantities'; the latter distinction has no meaning within a Bayesian framework. (Recall the example on page 94 of estimating daughters' heights from mothers' heights.)

methods using linear transformations for averaging over large numbers of potential predictors. Chipman, Kolaczyk, and McCulloch (1997) consider Bayesian models for wavelet decompositions.

The perspective on analysis of variance given here is from Gelman (2005); previous work along similar lines includes Plackett (1960), Yates (1967), and Nelder (1977, 1994), and Hodges and Sargent (2001). Volfovsky and Hoff (2012) propose a class of structured models for hierarchical regression parameters, going beyond the simple model of coefficients exchangeable in batches.

15.9 Exercises

1. Varying-coefficients models: express the educational testing example of Section 5.5 as a hierarchical linear model with eight observations and known observation variances. Draw simulations from the posterior distribution using the methods described in this chapter.
2. Fitting a hierarchical model for a two-way array:
   (a) Fit a standard analysis of variance model to the randomized block data discussed in Exercise 8.5, that is, a linear regression with a constant term, indicators for all but one of the blocks, and all but one of the treatments.
   (b) Summarize posterior inference for the (superpopulation) average penicillin yields, averaging over the block conditions, under each the four treatments. Under this measure, what is the probability that each of the treatments is best? Give a 95% posterior interval for the difference in yield between the best and worst treatments.
   (c) Set up a hierarchical extension of the model, in which you have indicators for all five blocks and all five treatments, and the block and treatment indicators are two sets of varying coefficients. Explain why the means for the block and treatment indicator groups should be fixed at zero. Write the joint distribution of all model parameters (including the hierarchical parameters).
   (d) Compute the posterior mode of the three variance components of your model in (c) using EM. Construct a normal approximation about the mode and use this to obtain posterior inferences for all parameters and answer the questions in (b). (Hint: you can use the general regression framework or extend the procedure in Section 13.6.)
   (e) Check the fit of your model to the data. Discuss the relevance of the randomized block design to your check; how would the posterior predictive simulations change if you were told that the treatments had been assigned by complete randomization?
   (f) Obtain draws from the actual posterior distribution using the Gibbs sampler, using your results from (d) to obtain starting points. Run multiple sequences and monitor the convergence of the simulations by computing $\widehat{R}$ for all parameters in the model.
   (g) Discuss how your inferences in (b), (d), and (e) differ.
3. Regression with many explanatory variables: Table 15.2 displays data from a designed experiment for a chemical process. In using these data to illustrate various approaches to selection and estimation of regression coefficients, Marquardt and Snee (1975) assume a quadratic regression form; that is, a linear relation between the expectation of the untransformed outcome, y, and the variables x_1, x_2, x_3, their two-way interactions, x_1x_2, x_1x_3, x_2x_3, and their squares, x_1^2, x_2^2, x_3^2.
   (a) Fit an ordinary linear regression model (that is, nonhierarchical with a uniform prior distribution on the coefficients), including a constant term and the nine explanatory variables above.
   (b) Fit a mixed-effects linear regression model with a uniform prior distribution on the constant term and a shared normal prior distribution on the coefficients of the nine

Reactor temperature (°C), x_1	Ratio of H_2 to n-heptane (mole ratio), x_2	Contact time (sec), x_3	Conversion of n-heptane to acetylene (%), y
1300	7.5	0.0120	49.0
1300	9.0	0.0120	50.2
1300	11.0	0.0115	50.5
1300	13.5	0.0130	48.5
1300	17.0	0.0135	47.5
1300	23.0	0.0120	44.5
1200	5.3	0.0400	28.0
1200	7.5	0.0380	31.5
1200	11.0	0.0320	34.5
1200	13.5	0.0260	35.0
1200	17.0	0.0340	38.0
1200	23.0	0.0410	38.5
1100	5.3	0.0840	15.0
1100	7.5	0.0980	17.0
1100	11.0	0.0920	20.5
1100	17.0	0.0860	19.5

Table 15.2 *Data from a chemical experiment, from Marquardt and Snee (1975). The first three variables are experimental manipulations, and the fourth is the outcome measurement.*

variables above. If you use iterative simulation in your computations, be sure to use multiple sequences and monitor their joint convergence.

(c) Discuss the differences between the inferences in (a) and (b). Interpret the differences in terms of the hierarchical variance parameter. Do you agree with Marquardt and Snee that the inferences from (a) are unacceptable?

(d) Repeat (a), but with a t_4 prior distribution on the nine variables.

(e) Discuss other models for the regression coefficients.

4. Analysis of variance:

(a) Create an analysis of variance plot for the educational testing example in Chapter 5, assuming that there were exactly 60 students in the study in each school, with 30 receiving the treatment and 30 receiving the control.

(b) Discuss the relevance of the finite-population and superpopulation standard deviation for each source of variation.

5. Modeling correlation matrices:

(a) Show that the determinant of a correlation matrix R is a quadratic function of any of its elements. (This fact can be used in setting up a Gibbs sampler for multivariate models.)

(b) Suppose that the off-diagonal elements of a 3×3 correlation matrix are 0.4, 0.8, and r. Determine the range of possible values of r.

(c) Suppose all the off-diagonal elements of a d-dimensional correlation matrix R are equal to the same value, r. Prove that R is positive definite if and only if $-1/(d-1) < r < 1$.

6. Analysis of a two-way stratified sample survey: Section 8.3 and Exercise 11.7 present an analysis of a stratified sample survey using a hierarchical model on the stratum probabilities. That analysis is not fully appropriate because it ignores the two-way structure of the stratification, treating the 16 strata as exchangeable.

(a) Set up a linear model for $\text{logit}(\phi)$ with three groups of varying coefficients, for the four regions, the four place sizes, and the 16 strata.

(b) Simplify the model by assuming that the ϕ_{1j}'s are independent of the ϕ_{2j}'s. This separates the problem into two generalized linear models, one estimating Bush vs. Dukakis preferences, the other estimating 'no opinion' preferences. Perform the computations for this model to yield posterior simulations for all parameters.

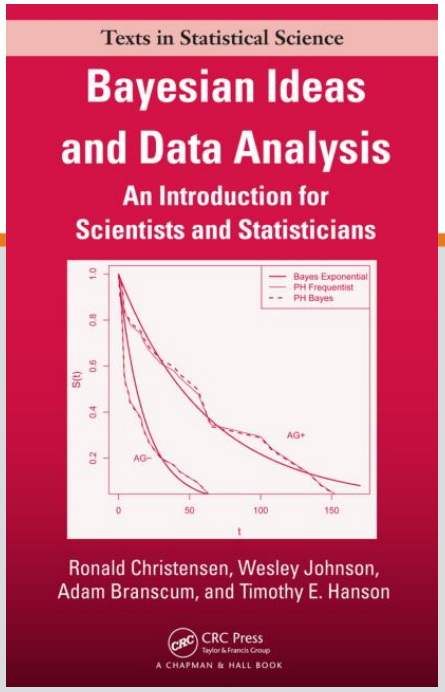
(c) Expand to a multivariate model by allowing the ϕ_{1j}'s and ϕ_{2j}'s to be correlated. Perform the computations under this model, using the results from Exercise 11.7 and part (b) above to construct starting distributions.

(d) Compare your results to those from the simpler model treating the 16 strata as exchangeable.

CHAPTER

5

NONPARAMETRIC MODELS



This chapter is excerpted from

Bayesian Ideas and Data Analysis: An Introduction for Scientists and Statisticians

by Ronald Christensen, Wesley Johnson, Adam Branscum, Timothy E Hanson.

Learn more

Chapter 15

Nonparametric Models

Measurements on the heights of a random sample of people are likely to be bimodal with women noticeably shorter on average than men. There are two ways to handle such data. If we know each individual's sex, i.e., if the covariate "sex" has been measured, we can examine the women and men separately, using, say, a normal distribution for each group. However, if we do not have the covariate available, we need a more general class of distributions than the normal to deal with the bimodal behavior. This is a very simple example but it conveys the spirit of nonparametric modeling. When the standard models that we have used fail to capture the salient aspects of the data, we need to develop more general models that are appropriate. The most common nonparametric models come in two flavors: incorporating more general families of distributions and incorporating more general mean structures. We can use broader classes of distributions or we can use more complicated regression functions but we seldom generalize both simultaneously.

In Chapters 7 through 11 we dealt with reasonably simple regression problems that involved simple distributions such as the binomial and normal and reasonably simple regression functions. In this chapter we will expand on both aspects of those procedures. Thus far our distributional models for data have been built on families indexed by one or two parameters such as Bern(p), $N(\mu, 1/\tau)$, Pois(λ), Exp(λ), and Weib(α, β). We now present some broader classes of distributions that involve many more parameters. Our earlier regression functions were known functions of unknown linear combinations of the predictor variables. Our treatment of nonparametric modeling of regression functions is (theoretically) in the same vein but the applications are considerably more complicated.

Many generalizations of the regression function can be viewed as straightforward adaptations of the procedures illustrated in Chapters 7 through 11. One approach is to use the current predictors to define additional predictors and then, as before, use known functions of unknown linear combinations from this expanded set of predictors to model regression functions. Another approach uses the simpler methods from earlier chapters but on subsets of the data and then combines the information from the subsets. Both approaches involve fitting many more parameters to the data.

Nonparametric models are anything but nonparametric. These models involve many parameters. In our discussion parameters are added to a basic model to increase the possible shapes for either the density function or the regression function. In Section 1 we discuss more general ways to model distributions. In Section 2 we examine more general ways for defining regression functions. In Section 3 we briefly discuss the application of Section 2 to estimating a baseline hazard for the Proportional Hazards model of Section 13.2.

We provide WinBUGS code on our website for many of the procedures illustrated here. Additionally, DPpackage (Jara, 2007; Jara et al., 2009) is an R package providing a library of functions for fitting various Bayesian nonparametric models for density estimation, generalized linear mixed models, generalized additive models, receiver operator characteristic curve analyses, meta-analysis, dependent processes, survival analysis, etc.

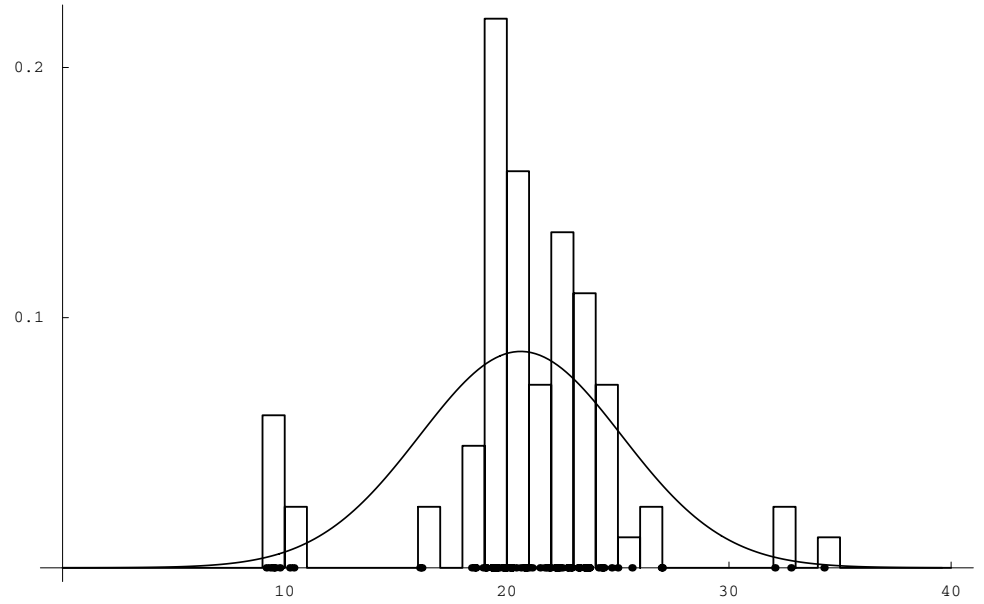


Figure 15.1: *Galaxy data: histogram and normal fit.*

15.1 Flexible Density Shapes

We focus on three methods for defining more flexible distributions: finite mixtures, Dirichlet process mixtures, and mixtures of Polya trees. The following example will be used to illustrate concepts.

EXAMPLE 15.1.1. *Galaxy Data*. Figure 15.1 shows a dot plot and histogram of $n = 82$ galaxy velocities obtained from Roeder (1990). Also on the plot is the predictive density from fitting a normal model with reference priors: specifically

$$y_1, \ldots, y_{82} | \mu, \tau \overset{iid}{\sim} N(\mu, \tau^{-1}),$$

$$\mu \sim N(0, 1000) \quad \perp\!\!\!\perp \quad \tau \sim \text{Gamma}(0.001, 0.001).$$

The simple normal model does not capture the three well-separated "clumps" of observations seen in the data.

15.1.1 Finite Mixtures

One way to enrich the class of possible density shapes is to consider a finite weighted mixture of two or more parametric distributions, cf. Sections 4.14 and 11.2. For the simple mixture model involving men and women's heights, let y be a random height. Let women's heights be $X_1 \sim N(\mu_1, 1/\tau_1)$, men's heights be $X_2 \sim N(\mu_2, 1/\tau_2)$, and W be an (unobserved) 0-1 indicator of sex with $W \sim \text{Bern}(p)$. An observation y has the distribution of $y = WX_1 + (1-W)X_2$. In other words, with probability p the observation comes from X_1, otherwise it comes from X_2. This mixture model has 5 parameters and density

$$f(y|\mu_1, \mu_2, \tau_1, \tau_2, p) = \frac{p}{\sqrt{2\pi/\tau_1}} e^{-0.5(y-\mu_1)^2\tau_1} + \frac{1-p}{\sqrt{2\pi/\tau_2}} e^{-0.5(y-\mu_2)^2\tau_2}. \tag{1}$$

Figure 15.2 shows some of the variety displayed by the densities in this class. Two of the curves are bimodal, but very different. The third curve is unimodal but skewed.

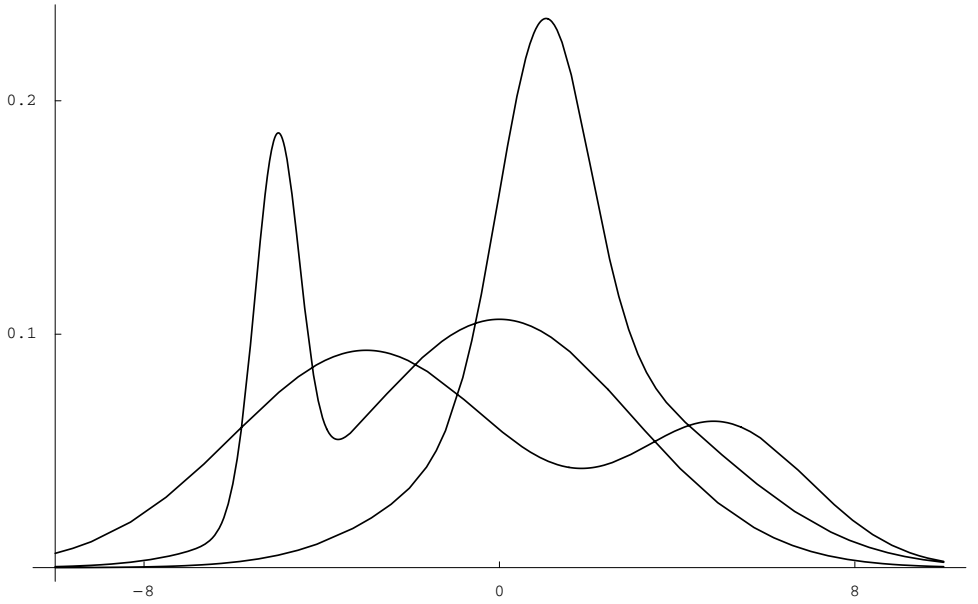


Figure 15.2: *Examples of two component normal densities.*

Finite mixture models often arise when the overall population is a combination of distinct subpopulations, e.g., sexes, genetic makeups, or subspecies. If the subpopulation for each observation is known, it is probably best to analyze the subpopulations separately and, if necessary, recombine the results for the overall population. Often the subpopulations are unknown and we must work directly with a mixture model.

EXERCISE 15.1. Show that the density in (1) integrates to one. Find the cdf in terms of the standard normal cdf $\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-0.5z^2} dz$.

In general, imagine K subpopulations each characterized with a density ϕ_k involving a vector parameter θ_k of dimension r_k and let w_k be the proportion of the overall population that comes from the kth subpopulation. The density is

$$f(y|\theta_1, \ldots, \theta_K; w_1, \ldots, w_K) = \sum_{k=1}^{K} w_k \phi_k(y|\theta_k).$$

Most often we take the ϕ_ks all the same, so

$$f(y|\theta_1, \ldots, \theta_K; w_1, \ldots, w_K) = \sum_{k=1}^{K} w_k \phi(y|\theta_k)$$

with $r_1 = \cdots = r_K \equiv r$. For example, in the normal mixture model we take $\theta_k = (\mu_k, \tau_k)'$ and each $\phi(y|\mu_k, \tau_k)$ is the density of a $N(\mu_k, 1/\tau_k)$.

Placing a prior on K leads to a model that changes dimension (number of parameters) with K. Such *trans-dimensional* models are difficult to fit. One approach uses *reversible jump MCMC*. The `WinBUGS` add-on *Jump* provides this capability. We will not discuss it further.

It is easier to analyze these models if you know (or pretend to know) the finite value of K. Non-Bayesians typically fit such models by first estimating K, then estimating the remaining model parameters, typically using maximum likelihood (via the *EM algorithm*). The number of components K can be estimated using model selection criteria as discussed in Section 4.9. We recommend a cross-validatory measure such as LPML.

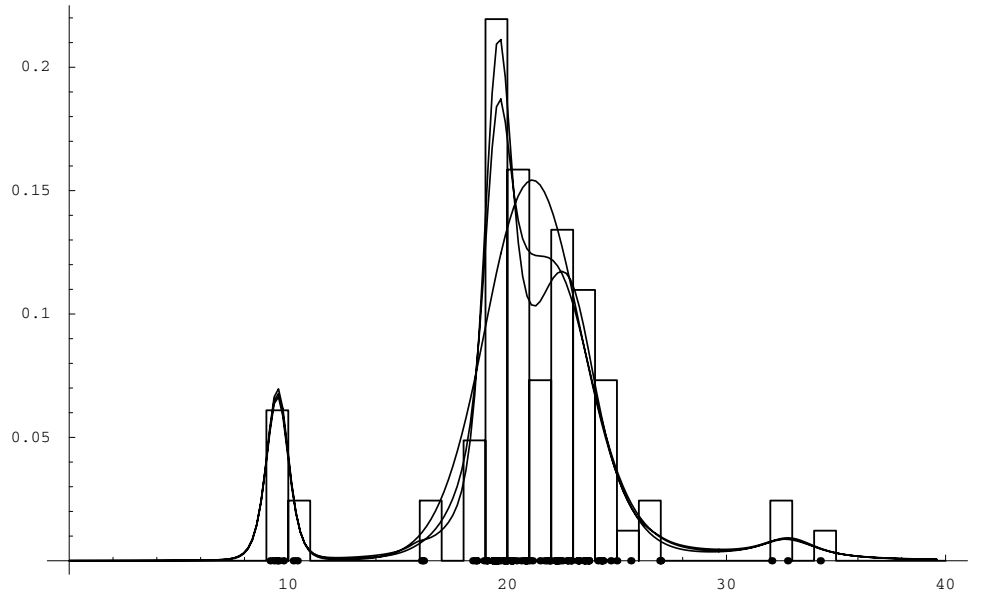


Figure 15.3: *Galaxy data: fits from finite mixture models,* $K = 3, 4, 6$.

EXAMPLE 15.1.2. *Galaxy Data*. To use a mixture model on these data, the first question is "How big should K be?" Looking at Figure 15.1, $K = 3$ or $K = 4$ might suffice. Picking K too small is a much bigger mistake than picking K too large. You cannot get three modes out of a mixture of two normals, but you can essentially ignore extra mixture terms and get bimodal distributions from mixtures of three or more distributions.

Using independent reference priors $\tau_i \sim \text{Gamma}(0.001, 0.001)$, $\mu_i \sim N(0, 1000)$, and $(w_1, \ldots, w_K) \sim \text{Dir}(1/K, \ldots, 1/K)$, the LPMLs for $K = 3, 4, 5, 6, 7$ are -193.2, -184.0, -180.4, -175.5, and -178.5, indicating $K = 6$ provides the best predictive fit. Figure 15.3 illustrates the results of fitting some mixture models. A simple normal model ($K = 1$) gives an LPML of -243.1 and was illustrated in Figure 15.1.

The number of mixands K can be chosen *a priori* to be quite large. The number of actual components being used can be monitored through an MCMC run and the analysis modified accordingly. The selection of K to a number a bit larger than is necessary, but not grossly more, achieves parsimony and can improve the predictive ability of the model.

When the densities ϕ_k are the same, it is often reasonable to think of the vectors θ_k as exchangeable in that their ordering should not matter. Depending on the situation, the w_ks may also be exchangeable. A general, yet convenient, prior for a mixture of normals is

$$\mu_1, \mu_2, \ldots, \mu_K \overset{iid}{\sim} N(\tilde{m}, 1/\tilde{\tau}) \quad \perp\!\!\!\perp \quad \tau_1, \tau_2, \ldots, \tau_K \overset{iid}{\sim} \text{Gamma}(a, b),$$

with independent weights

$$(w_1, w_2, w_3, \ldots, w_K) \sim \text{Dir}(\alpha/K, \ldots, \alpha/K).$$

To be useful, this prior needs some additional restrictions as discussed later. If desired, hyperpriors can be placed on $\tilde{m}, \tilde{\tau}, a, b$, and α.

Without further qualification or restriction, mixture models often lack identifiability. This means that there exists more than one set of parameter values that will generate the same distribution for the data, cf. Section 4.14 and the Hui-Walter model of Chapter 14. With an exchangeable prior like

that just presented, lack of identifiability can result in serious problems that we discuss near the end of this subsection. Fortunately, the identifiability problem is often resolvable. In a mixture of two normals, if we constrain the means so that $\mu_1 < \mu_2$, the model becomes identifiable. In the K normals mixture, placing an ordering on all K means similarly resolves the issue. Ordering the means is sufficient; we need impose no restrictions on the precisions. For a Bayesian, the simplest way to impose an ordering on the means is to specify a prior that is ordered with probability 1. Imposing such an ordering is easy in WinBUGS, see the code in Example 15.1.3. But just restricting the model to be identifiable does not guarantee well-behaved Markov chains!

Nonidentifiability is mitigated if an informative prior is incorporated into the analysis. For example, if the data are people's heights, we can construct informative priors for both males and females and typically the prior for male heights would be (stochastically) larger. (We might be sampling from a mixture of female basketball players and male gymnasts.) Thus, our information on females provides an informative prior on (μ_1, τ_1) and our information on males provides an informative prior on (μ_2, τ_2). The informative prior should reduce identifiability problems, but there seems little reason not to require $\mu_2 > \mu_1$, which definitely eliminates any identifiability problems. We might also have prior information on the proportion of females in the sample. Mixture distributions can be difficult to fit and, when it is available, good prior information certainly helps.

To analyze a random sample of mixture data, say, $y_1, \ldots, y_n$, we introduce independent subpopulation variables X_{ik} with $X_{ik} \sim \phi_k(y|\theta_k)$ for $k = 1, \ldots, K$ and independent index variables W_i with $\Pr[W_i = k] = w_k$. The random observation from the mixture is now

$$y_i = X_{iW_i} = \sum_{k=1}^{K} X_{ik} I_{\{k\}}(W_i).$$

The first equality provides a particularly convenient way to code mixtures in WinBUGS using the `dcat` option. Note that for $K = 2$, $(W_i - 1) \sim \text{Bern}(w_2)$.

EXAMPLE 15.1.3. The data y are a simulated random sample from a mixture of two distributions with $n = 100$, specifically $N(0,1)$ and $N(3,1/2)$ distributions with $w_1 = 0.34$. The complete data are on our website. Here W_i takes the values $(1,2)$ with probabilities $\{(w_1, w_2) : w_2 = 1 - w_1)\}$; $W_i = 1$ indicates that y_i was taken from the first population, and $W_i = 2$ indicates it was taken from the second. Our WinBUGS code is

```
model{
 for(i in 1:n){
   W[i] ~ dcat(w[1:2])
   y[i] ~ dnorm(mu[W[i]],tau[W[i]])
 }
 w[1:2] ~ ddirch(alpha[1:2])
 mu[1]  ~ dnorm(0,0.0001)
 mu[2]  ~ dnorm(0,0.0001)I(mu[1],)
 for(i in 1:2){ tau[i] ~ dgamma(0.001,0.001) }
}
list(w=c(0.3,0.7), mu=c(0,2))
list(n=100, alpha = c(1,1),
     y = c(-1.902,-1.579,-1.368,-1.204,
           -1.068,-0.949,-0.842,-0.744,
           -0.652,-0.566,...,4.883,5.172))
```

The prior has $\mu_1 \sim N(0, 10000)$ and, using a WinBUGS feature discussed in Chapter 12, $\mu_2|\mu_1 \sim N(0, 10000)I_{(\mu_1,\infty)}(\mu_2)$, so that μ_2 is $N(0, 10000)$ but conditioned on $\mu_2 > \mu_1$. Our MCMC initial value for μ_2 is larger than for μ_1. Unfortunately, we got a *trap* message using related code. If you get trapped, close the `trap` window and click on the `update` button. You may be able to get a

reasonable sample this way. Alternatively, you could change the initial value for μ_2, increase its prior mean, or use a random walk prior as discussed after Exercise 15.2.

EXERCISE 15.2. (a) Run the code from Example 15.1.3 to obtain estimates of the means, precisions, and the mixing parameter. (b) Modify the code to obtain the predictive density of a future observation from the model (see the code in Exercise 15.3). Take the WinBUGS output from CODA and make a density plot in R. Plot the true mixture density on the same graph and evaluate the estimate. (c) Now run the code with different priors including one with $\mu_1 \sim N(0,1)$, $\mu_2|\mu_1 \sim N(2,1)I_{(\mu_1,\infty)}(\mu_2)$ and independently $\tau_k \sim \text{Gamma}(1,1)$. Report on what priors you tried, what worked, and what didn't. In no way are we advocating trial and error for obtaining priors. We are illustrating the importance of good prior information for mixture problems.

An alternative to the exchangeable prior with mean restrictions is to use a *random walk prior*. In this prior $\mu_{k+1} = \mu_k + \delta_k$ where δ_k a positive random variable. A reference prior for the mixture model places a reference prior on μ_1 and independent reference priors on each δ_k. In Exercise 15.3, $K = 2$ and we take δ_1 as a half-normal distribution. Specifically, our prior has $\mu_1 \sim N(0,1000)$ and $\mu_2 = \mu_1 + \delta_1$ where $\delta_1 \sim N(0,1000)$ constrained so that $\delta_1 > 0$; hence $\mu_2 > \mu_1$. The exercise also involves real data and illustrates computation of the predictive density.

EXERCISE 15.3. Consider $n = 66$ log-transformed ELISA scores from cows infected with Johnes' disease (Choi et al., 2006a). The authors analyzed these data using a simple normal model. Assume instead that the data y_i are a random sample from a mixture of two normal distributions. In WinBUGS use

```
model{
 # Sampling model
 for(i in 1:n){
 W[i] ~ dcat(w[1:2])
 y[i] ~ dnorm(mu[W[i]],tau[W[i]])
 }
 # Prior
 w[1:2] ~ ddirch(alpha[1:2])
 mu[1] ~ dnorm(0,0.001)
 delta1 ~ dnorm(0,0.001)I(0,)
 mu[2] <- mu[1]+delta1
 for(i in 1:2){tau[i] ~ dgamma(0.001,0.001)}
 # Density estimate
 for(i in 1:100){
   grid[i] <- -3.5+5*(i-1)/99
   f[i] <- 0.3989*w[1]*sqrt(tau[1])
           *exp(-0.5*pow(grid[i]-mu[1],2)*tau[1])
          +0.3989*w[2]*sqrt(tau[2])
           *exp(-0.5*pow(grid[i]-mu[2],2)*tau[2])
 }
}
list(w=c(0.5,0.5), mu=c(-2,NA), delta1=2, tau=c(1,1)) # inits
list(n=66, alpha=c(1,1), y=c(-2.3,0.66,-1.11,0.52,0.93,0.8,0.07,
0.43,-2.66,0.46,-0.34,0.53,0.17,-0.04,0.35,-2.81,0.4,0.13,0.88,
0.83, -2.3,-2.53,-1.35,-1.71,0.55,-3.22,0.13,0.6,-0.07,-1.56,1.02,
0.61,0.52,0.03,0.6,-2.04,0.76,-1.39, 0.51,0.15,-0.26,0.98,0.8,
0.24,0.12,0.67,-0.07,-0.31,0.41,-1.05,-1.35,-1.71,0.92,-1.2,-2.3,
-1.27,0.49,-0.26,-2.81,-0.82,-0.73,-0.2,0.59,0.58,-1.77,-2.04))
```

(a) Run the code and obtain inferences for the means, precisions, and the mixing parameter. Plot the fitted density on top of a histogram of the raw data. (b) Repeat this exercise with a simple (one-component) normal model; comment on the fits of the one-component versus two-component models.

*15.1.1.1 Identifiability Issues**

Mixture models are notorious for presenting identifiability problems. The general concept of identifiability was discussed in Section 4.14 including an example of a mixture of two normals with known variances. This discussion presupposes familiarity with the earlier one.

Imagine sampling from a mixture of K subpopulations, each characterized with a density ϕ involving an identifiable vector parameter θ_k of dimension r and let w_k be the proportion of the overall population that comes from the kth subpopulation. Let X_k denote an observation from the kth subpopulation and let W randomly determine the subpopulation sampled. Specifically, for $k = 1,\ldots,K$, consider the well-defined (identifiable) sampling model for all subpopulations

$$X_k|\theta_k \overset{ind}{\sim} \phi(x|\theta_k) \quad \perp\!\!\!\perp \quad W|w_1,\ldots,w_K \sim \Pr[W=k] = w_k.$$

The w_ks must be nonnegative and sum to 1. For the complete set of random variables $Z \equiv (X_1,\ldots,X_K,W)'$, write the parameter vector as $\Theta \equiv (\theta_1',\ldots,\theta_K',w_1,\ldots,w_K)'$. Θ is identifiable for Z. The K normals mixture model is the special case that takes $\theta_k = (\mu_k,\tau_k)'$ and each $\phi(y|\mu_k,\tau_k)$ to be the density of a $N(\mu_k, 1/\tau_k)$.

Unfortunately, in the mixture model an observation is randomly chosen from one subpopulation, so we only observe

$$y = \sum_{k=1}^{K} X_k I_{\{k\}}(W).$$

By considering how to compute $\Pr[y \in A]$ for some set A, it is easy to see that the density of y is

$$f(y|\Theta) \equiv f(y|\theta_1,\ldots,\theta_K;w_1,\ldots,w_K) = \sum_{k=1}^{K} w_k \phi(y|\theta_k).$$

Just as for the mixture of two normals in Section 4.14, the originally identifiable parameter Θ for the vector Z becomes nonidentifiable when only y is observed.

For the K normals mixture model, one often assumes $\mu_1 < \cdots < \mu_K$. Specifically, $\Theta = [(\mu_1,\tau_1,w_1),\cdots,(\mu_K,\tau_K,w_K)]$ is the original parameter vector. To obtain an identifiable parameterization for y, consider $y|g(\Theta)$ where $g(\Theta) = [(\mu_{(1)},\tau^{(1)},w^{(1)}),\cdots,(\mu_{(K)},\tau^{(K)},w^{(K)})]$ with $\mu_{(1)} < \mu_{(2)} < \cdots < \mu_{(K)}$ and if $\mu_{(k)} = \mu_j$, then $\tau^{(k)} \equiv \tau_j$ and $w^{(k)} \equiv w_j$. Another common, and quite general, way to impose identifiability in mixture models is to require $w_1 < w_2 < \cdots < w_K$.

Restricting the parameters of the sampling distribution makes the corresponding likelihood painful to use. It is difficult to maximize for frequentists and it is difficult to integrate for Bayesians. However, using MCMC, Bayesians have a convenient way of avoiding this pain. After finding an appropriate restriction on the parameters that is identifiable, it can be relatively easy to sample from a prior distribution that gives probability 1 to parameter values that satisfy the restriction. So rather than actually restricting the parameters of the sampling distribution, it is often easier to let the prior do the work by defining a prior that satisfies those same restrictions with probability 1. In a K normals mixture model, rather than assuming $\mu_1 < \cdots < \mu_K$ in the sampling distribution, we can choose a prior with $\Pr[\mu_1 < \cdots < \mu_K] = 1$ to accomplish the same thing. In general, this would often occur by defining a prior on Θ but restricting it to $g(\Theta)$. For example, the exchangeable prior for normal mixtures is restricted to ordered means.

As discussed in Section 4.14, it is tempting for Bayesians to be careless about identifiability because answers can be obtained without it. In particular, informative priors often eliminate or mitigate identifiability problems. For example, an informative prior that happens to have a large

probability for $\Pr[\mu_1 < \cdots < \mu_K]$ will mitigate identifiability problems in K normals mixtures even if the user is ignoring them.

Priors can help with identifiability problems, but they can also exacerbate them. One might assume a prior with

$$\theta_1, \cdots, \theta_K \quad \perp\!\!\!\perp \quad w_1, \cdots, w_K$$

and, as discussed above, it often seems reasonable to think of the vectors θ_k as exchangeable in that their ordering should not matter, so assume

$$\theta_1, \theta_2, \ldots, \theta_K \overset{iid}{\sim} p(\theta).$$

Then, given our assumptions, for any k

$$y|W = k \ \sim \ X_k \sim \ \int \phi(y|\theta) p(\theta) d\theta$$

which is a distribution that does not depend on k. Thus the data y are independent of W and there is no information to be gained about W from the data. In the normal case with $X_k|\theta_k \sim N(\theta_k, 1)$ and $\theta_k \sim N(0,1)$, we get $X_k \sim N(0,2)$ for all k. Since y comes from one of these K indistinguishable distributions, it is no wonder y cannot give information on which subpopulations are more likely to have generated it.

Marin and Robert (2007) contains a good discussion of Bayesian finite mixture models.

15.1.2 Dirichlet Process Mixtures: Infinite Mixtures

In the previous subsection, we considered finite mixtures

$$f(y|\theta_1, \ldots, \theta_K; \, w_1, \ldots, w_K) = \sum_{k=1}^{K} w_k \phi(y|\theta_k).$$

We now look at infinite mixtures

$$f(y|\theta_1, \theta_2, \ldots; \, w_1, w_2, \ldots) = \sum_{k=1}^{\infty} w_k \phi(y|\theta_k). \tag{2}$$

One way to generate a density such as (2) with fixed values for w_k and θ_k is to think of the sampling distribution $\phi(y|\theta)$ and place a discrete prior distribution G on θ with density $p_G(\theta_k) = w_k$, $k = 1, 2, \ldots$. The marginal distribution of y is then,

$$f(y) = \int \phi(y|\theta) dG(\theta) \equiv \sum_{k=1}^{\infty} w_k \phi(y|\theta_k) \tag{3}$$

where the notation $dG(\theta)$ turns the integral into the appropriate sum because the distribution is discrete. G depends on the values chosen for $\theta_1, \theta_2, \ldots$ and $w_1, w_2, \ldots$. We can think of these as hyperparameters of the prior distribution G and rewrite (3) as

$$f(y|\theta_1, \theta_2, \ldots; w_1, w_2, \ldots) \equiv f(y|G) = \int \phi(y|\theta) dG(\theta) = \sum_{k=1}^{\infty} w_k \phi(y|\theta_k).$$

The next thing we are going to do is put a prior on these (hyper)parameters, thus creating a hierarchical model as in Section 4.12.

Placing priors on these parameters amounts to picking a discrete distribution G that is chosen at random. Picking a random distribution sounds strange, but we do it for the ys all the time. If $y|\theta$ has density $f(y|\theta)$ and θ has a prior distribution with density $p(\theta)$, we are conceptually picking a θ at

random and applying the random sampling distribution with density $f(y|\theta)$, so we have a random distribution for y. Our usual marginal distribution is an average of these random distributions,

$$f(y) = \int f(y|\theta)p(\theta)d\theta,$$

which is a form of mixture distribution. Similarly, when we have data available, our usual predictive distribution for $\tilde{y}$ given data y is also a mixture,

$$f_p(\tilde{y}|y) = \int f_p(\tilde{y}|\theta)p(\theta|y)d\theta,$$

again obtained by averaging over a random density $f_p(\tilde{y}|\theta)$ with a distribution on θ.

We are just moving the procedure back a step. Now we are picking a random distribution G for θ by putting a prior on $\theta_1, \theta_2, \ldots$; $w_1, w_2, \ldots$. Placing a reasonable prior on the θs is easy. Simply take them as iid from a single distribution that is our best prior guess for the distribution of θ, call this G_0. However, putting a prior on the w_is is more difficult because they have to be randomly chosen numbers that are between 0 and 1 that add to 1.

Rather than putting a prior on the sequence of probabilities $w_1, w_2, w_3, \ldots$, we instead put a prior on another sequence $q_1, q_2, q_3, \ldots$ that just consists of numbers between 0 and 1. From this we induce a prior on $w_1, w_2, w_3, \ldots$ by the transformation

$$w_k = q_k \prod_{j=1}^{k-1}(1 - q_j) \tag{4}$$

with inverse transformation

$$q_k = w_k \Big/ \left(1 - \sum_{j=1}^{k-1} w_j\right). \tag{5}$$

For (5) to be the inverse of (4) requires $(1 - \sum_{j=1}^{k-1} w_j) = \prod_{j=1}^{k-1}(1 - q_j)$. This is easily proved by induction after observing from (5) that $1 - q_k = (1 - \sum_{j=1}^{k} w_j)/(1 - \sum_{j=1}^{k-1} w_j)$. The choice of the transformation (4) is discussed below. In particular, we take

$$q_i \overset{iid}{\sim} \text{Beta}(1, \alpha).$$

The question now becomes, "How do we know that these random w_ks define a set of probabilities, i.e., are nonnegative and sum to 1?" A proof is given on our website.

This approach to defining a prior distribution on a sequence of weights $w_1, w_2, \ldots$ can be thought of as breaking a stick of length 1. Each w_k is the length of a piece we break off. The w_ks should add up to 1. First we break off a fraction $q_1 \sim \text{Beta}(1, \alpha)$ from the stick. The length of the broken off piece is $w_1 = q_1$. There is $1 - w_1$ left of the stick. Now break off the fraction $q_2 \sim \text{Beta}(1, \alpha)$ of what remains. The length of this new piece is $w_2 = q_2(1 - w_1) = q_2(1 - q_1)$. Again, break off a fraction $q_3 \sim \text{Beta}(1, \alpha)$. The length is $w_3 = q_3(1 - w_1 - w_2)$. This is essentially equation (5) with $k = 3$, so by (4) $w_3 = q_3(1 - q_1)(1 - q_2)$. Continuing this process gives

$$w_k = q_k \prod_{j=1}^{k-1}(1 - q_j), \qquad q_1, q_2, \ldots \overset{iid}{\sim} \text{Beta}(1, \alpha).$$

Taking

$$\theta_1, \theta_2, \ldots \overset{iid}{\sim} G_0$$

together with the distribution of the w_ks, these determine our random distribution G by placing a prior on all of G's hyperparameters. The prior on G depends on only two parameters: the weight α in the Beta distributions and the distribution G_0.

This method for determining a random probability distribution G is known as a *Dirichlet process (DP)* and is written

$$G \sim \mathcal{D}(\alpha, G_0).$$

It was introduced by Ferguson (1973, 1974). The stick-breaking representation is due to Sethuraman (1994). Dirichlet process methods have dominated much of modern Bayesian nonparametric modeling.

Dirichlet processes have two particularly nice properties. First, if you take one observation $\tilde{\theta}$ with cdf $G(\theta|\theta_1,\theta_2,\theta_3,\ldots,\ w_1,w_2,w_3,\ldots)$ and the θ_ks and w_ks are randomly chosen as described here so that $G \sim \mathcal{D}(\alpha, G_0)$, then the marginal distribution of $\tilde{\theta}$ is just the G_0 distribution. Symbolically, if

$$\tilde{\theta}|G \sim G; \qquad G \sim \mathcal{D}(\alpha, G_0)$$

then

$$\tilde{\theta} \sim G_0.$$

Second, Dirichlet processes are closed under sampling. In other words, the posterior distribution is also a Dirichlet process. Specifically, suppose that $\tilde{\theta}_1,\ldots,\tilde{\theta}_n$ is a random sample with the cdf $G(\theta|\theta_1,\theta_2,\theta_3,\ldots,\ w_1,w_2,w_3,\ldots)$ and with θ_ks and w_ks chosen so that $G \sim \mathcal{D}(\alpha, G_0)$, then we write

$$\tilde{\theta}_1,\ldots,\tilde{\theta}_n|G \overset{iid}{\sim} G; \qquad G \sim \mathcal{D}(\alpha, G_0).$$

It turns out that

$$G|\tilde{\theta}_1,\ldots,\tilde{\theta}_n \sim \mathcal{D}\left(\alpha+n, \frac{\alpha}{\alpha+n}G_0 + \frac{1}{\alpha+n}\sum_{j=1}^{n}\delta_{\tilde{\theta}_j}\right)$$

where $\delta_{\tilde{\theta}_j}$ *is the distribution that gives probability 1 to* $\tilde{\theta}_j$. Clearly small values of α let the data play a larger role.

Sampling from (2) with the mixture determined by a Dirichlet process is called sampling from a *Dirichlet process mixture* (*DPM*) distribution. This has its roots in Antoniak (1974), Lo (1984), and others. A DPM density is written hierarchically as

$$f(y|G,\phi) \equiv \int \phi(y|\theta)dG(\theta) \equiv \sum_{k=1}^{\infty} w_k\phi(y|\theta_k),$$

$$G \sim \mathcal{D}(\alpha, G_0).$$

Dirichlet process mixtures involve an infinite mixture of $\phi(y|\theta)$s but we can only compute finite mixtures. Simple approximations truncate the sequences $\theta_1,\theta_2,\theta_3,\ldots$ and $w_1,w_2,w_3,\ldots$ for some large value of K into $\theta_1,\theta_2,\theta_3,\ldots,\theta_K$ and $w_1,w_2,w_3,\ldots,w_K$. Thus finite approximations to the Dirichlet process use a distribution G_K that has $\Pr[\theta = \theta_k] = w_k$ for $k = 1,\ldots,K$ and define distributions for the θ_ks and w_ks. A DPM is approximated similar to (3),

$$f(y\,|\,G_K) = f(y\,|\,\theta_1,\ldots,\theta_K,w_1,\ldots,w_K) = \int \phi(y|\theta)dG_K(\theta) \equiv \sum_{k=1}^{K} w_k\phi(y|\theta_k).$$

To make G_K random, it is natural to take $\theta_k \overset{iid}{\sim} G_0$. Another natural approximation defines the distribution of $w_1,w_2,w_3,\ldots,w_{K-1}$ as in a DP but then takes $w_K \equiv 1-\sum_{k=1}^{K-1} w_k$. Ishwaran and Zarepour (2002) [IZ] discuss a different approximation to the DP that has been used by many authors and that is simpler to compute. They take $(w_1,\ldots,w_K) \sim \text{Dirich}(\alpha/K,\ldots,\alpha/K)$. IZ argue that their distribution for G_K converges to the Dirichlet process $\mathcal{D}(\alpha, G_0)$ as $K \to \infty$. This is by no means obvious. We present an heuristic justification.

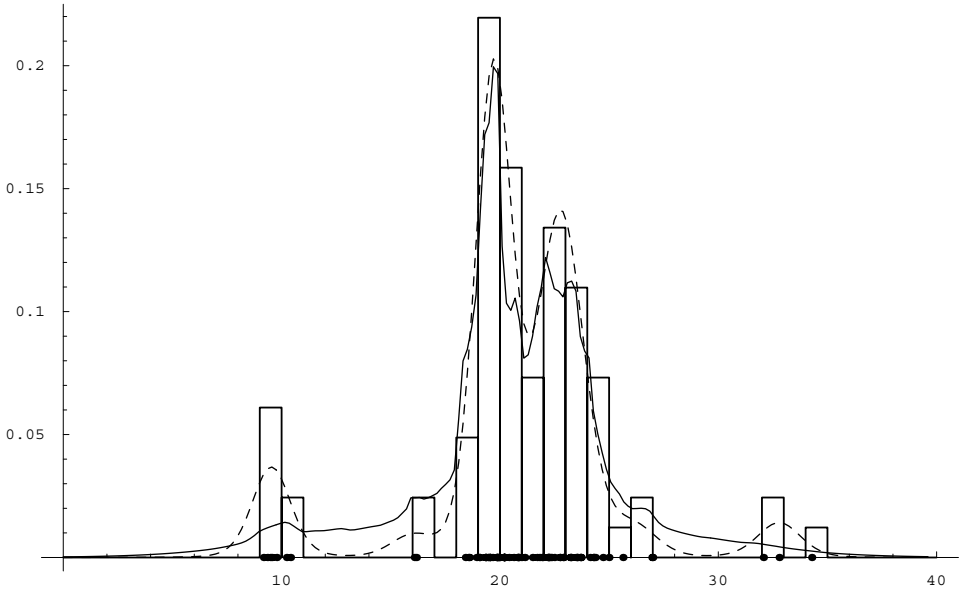


Figure 15.4: *Galaxy data: Dirichlet process mixture (dashed) and mixture of Polya trees (solid) fits.*

The IZ approximation is based on the original Ferguson (1973) definition of the Dirichlet process. By definition, $G \sim \mathcal{D}(\alpha, G_0)$ if for any J and any partition of the sample space $\{A_j : j = 1,\ldots,J,\}$, the random probabilities of the partition sets satisfy

$$[G(A_1), G(A_2), \ldots, G(A_J)] \sim \text{Dirich}[\alpha G_0(A_1), \alpha G_0(A_2), \ldots, \alpha G_0(A_J)].$$

Keeping the θ_ks fixed in G_K, place a Dirich$(\alpha/K, \ldots, \alpha/K)$ distribution on the w_ks to make G_K random. With the *empirical distribution* $F_K = (1/K)\sum_{k=1}^{K} \delta_{\theta_k}$, it is not difficult to see that for any partition

$$[G_K(A_1), G_K(A_2), \ldots, G_K(A_J)] \Big| \theta_1, \theta_2, \theta_3, \ldots, \theta_K \sim \text{Dirich}[\alpha F_K(A_1), \alpha F_K(A_2), \ldots, \alpha F_K(A_J)],$$

so by Ferguson's definition

$$G_K | \theta_1, \theta_2, \theta_3, \ldots, \theta_K \sim \mathcal{D}(\alpha, F_K).$$

Finally, if the θ_ks are iid from G_0, by standard results from probability theory, the empirical distribution F_K tends to G_0 as K grows. So heuristically, G_K is approximately distributed as $\mathcal{D}(\alpha, G_0)$ for large K.

EXAMPLE 15.1.4. *Galaxy Data*. Consider a Dirichlet process mixture of normals with $\alpha = 1$ and G_0 taken as a bivariate distribution with independent components. The first variable corresponds to the mean and the second to the precision of the normal distributions. We used a reference prior with $N(0, 1000)$ for the mean and a Gamma$(0.001, 0.001)$ for the precision. The IZ Dirichlet process approximation with $K = 50$ takes independent samples $\mu_k \sim N(0, 1000)$, $\tau_k \sim \text{Gamma}(0.001, 0.001)$, and $(w_1, \ldots, w_K) \sim \text{Dir}(1/50, \ldots, 1/50)$. Note the similarity to Example 15.1.2. For fitting this model, LPML is -158.4 which is considerably better than the LPML values for the low dimensional mixtures. The fit is illustrated in Figure 15.4 along with the mixture of Polya trees fitted model developed in the next subsection.

EXERCISE 15.4. WinBUGS code for fitting Example 15.1.4 is available in `DPMdensity.odc` on the book website. The code includes lines for computing the CPO statistics as well as the predictive density over a grid of points. Use the code to duplicate the Dirichlet process mixture of normals fit in Figure 15.4. Rerun the code with $\alpha = 0.1$, $\alpha = 1$, and $\alpha = 10$, being sure to monitor the number of "active" components via the node `total`. Do the predictive density, LPML, and number of components change very much with α?

DPpackage provides a slightly different analysis for Example 15.1.4.

EXERCISE 15.5. Install DPpackage into your version of R, cf. Appendix C. The function `DPdensity` fits a Dirichlet process mixture of normal distributions as proposed by Escobar and West (1995). Type `help(DPdensity)` in R to see a description of the model. How does this model differ from Example 15.1.4? Use this DP function to obtain a plot similar to Figure 15.4. Code with a "default" prior specification is found in `Chap15DPpackage.txt` on our website. The `DPdensity` function allows for random α. Try $\alpha \sim \text{Gamma}(1,1)$ and $\alpha \sim \text{Gamma}(1,0.1)$. How do posterior inferences change, especially the number of clusters (`ncluster`), the predictive density, and the LPML?

15.1.3 Mixtures of Polya Trees

Our discussion follows Christensen, Hanson, and Jara (2008). The general definition of mixtures of Polya trees (MPTs), such as that in Lavine (1992, 1994), is quite broad. Using the simpler definition in Hanson (2006), we use Polya trees to generalize the $N(\mu, \sigma^2)$ family of distributions, see Figure 15.5a. Other parametric families are generalized similarly.

The generalization goes through a number of stages, say J. At each stage we introduce new parameters to generalize the previous stage. At the first stage, we split the real number line, that is, the support of the normal distribution, into two intervals divided by the median μ. We then allow changes in the probabilities of being below or above μ but we retain the shape of the normal density both below μ and above μ. Figure 15.5b illustrates the density for the case when the probability of being below μ is 0.45.

The new parameters at the first stage are θ_{11}, the probability of being no greater than μ, and θ_{12}, the probability of being above μ. Formally, let X_1 have the first stage distribution, then

$$\theta_{11} \equiv \Pr[X_1 \le \mu]$$

and

$$\theta_{12} \equiv \Pr[X_1 > \mu] = 1 - \theta_{11}.$$

Because we retain the shape of the normal on both sets, if $a \le \mu$ and $Y \sim N(\mu, \sigma^2)$, conditionally we have

$$\Pr[X_1 \le a | X_1 \le \mu] \equiv \frac{\Pr[Y \le a]}{0.5} = 2\Phi[(a-\mu)/\sigma)]$$

where $\Phi(\cdot)$ is the cdf of a standard normal. Similarly, if $b > \mu$,

$$\Pr[X_1 > b | X_1 > \mu] \equiv 2\Pr[Y > b] = 2\{1 - \Phi[(b-\mu)/\sigma]\}.$$

Alternatively, we can write

$$\begin{aligned} \Pr[X_1 \le a] &= \Pr[X_1 \le a \,|\, X_1 \le \mu]\Pr[X_1 \le \mu] = \Pr[Y \le a]\, 2\theta_{11} \\ \Pr[X_1 > b] &= \Pr[X_1 > b \,|\, X_1 > \mu]\Pr[X_1 > \mu] = \Pr[Y > b]\, 2\theta_{12}. \end{aligned}$$

The density of the stage 1 distribution is

$$f(x_1|\mu, \sigma^2, \theta_{11}, \theta_{12}) = \frac{2^1}{\sqrt{2\pi}\sigma} e^{-(x_1-\mu)^2/2\sigma^2} \left[\theta_{11} I_{(-\infty,\mu]}(x_1) + \theta_{12} I_{(\mu,\infty)}(x_1)\right].$$

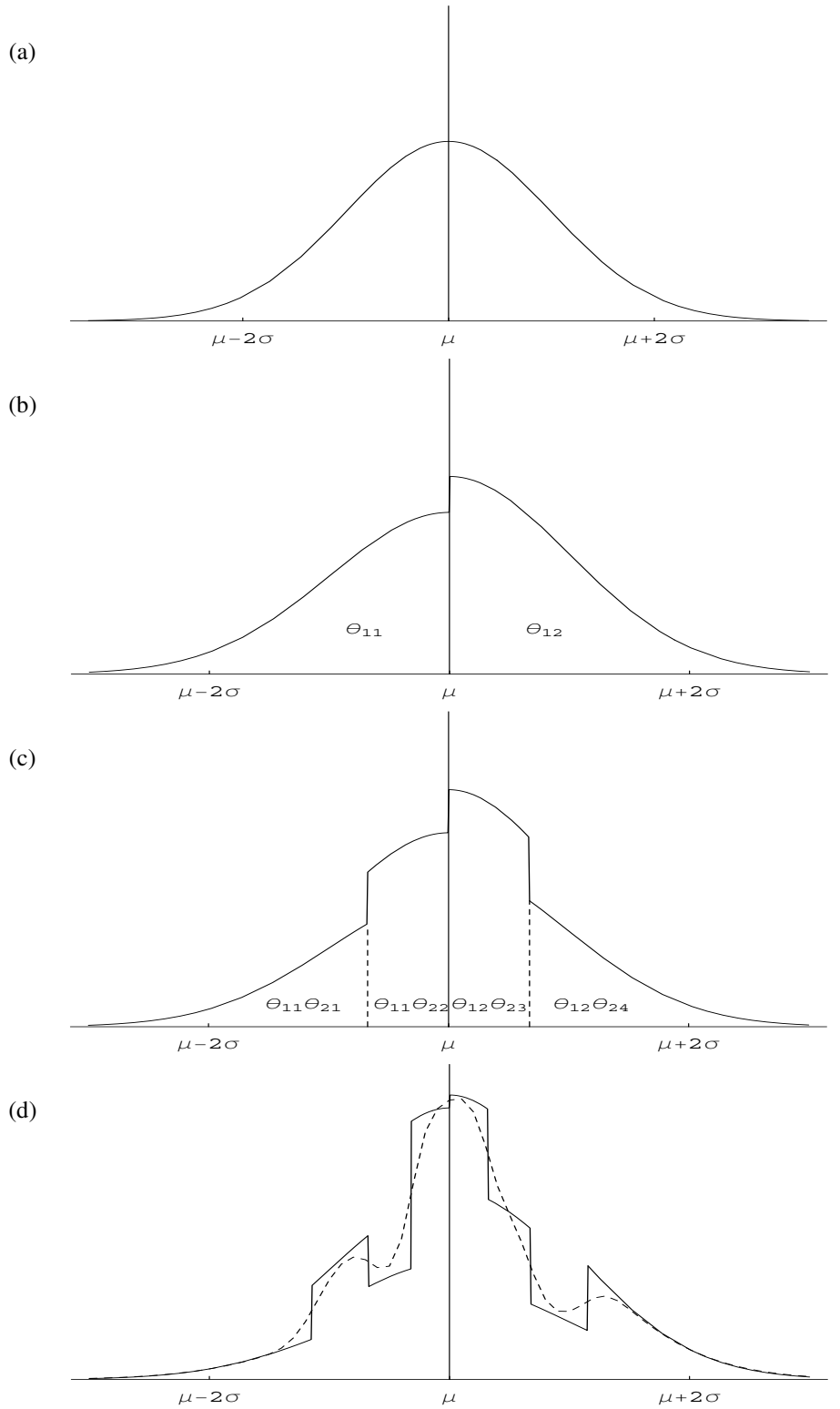


Figure 15.5: *Finite PT densities. (a) First:* $N(\mu, \sigma^2)$ *centering density. (b) Second:* $\theta_{11} = 0.45$. *(c) Third:* $\theta_{21} = 0.4$, $\theta_{23} = 0.6$. *(d) Fourth:* $\theta_{31} = 0.3$, $\theta_{33} = 0.3$, $\theta_{35} = 0.6$, $\theta_{37} = 0.3$*; centering family mixed over* $\mu \sim N(\mu_0, (\sigma/10)^2)$, $\sigma \sim N(\sigma_0, (\sigma_0/10)^2)$.

In the first stage of the process, we split the real line at the median μ of the $N(\mu,\sigma^2)$ distribution. For the second stage, we split the line at the quartiles, say, q_1, μ, q_3 of the original distribution leading us to consider the sets $(-\infty,q_1]$, $(q_1,\mu]$, $(\mu,q_3]$, (q_3,∞). For the normal, the quartiles are $q_1 \equiv \mu - 0.6745\sigma$, μ, and $q_3 \equiv \mu + 0.6745\sigma$. Under the original distribution, each of these sets has probability 0.25, but in stage 2 we allow the probabilities of the sets to change in a manner that is consistent with stage 1. An illustration of a stage 2 density is Figure 15.5c. Letting X_2 have the second stage distribution, we introduce new parameters, θ_{21}, θ_{22}, θ_{23}, θ_{24}, defined as conditional probabilities relative to the sets used in stage 1:

$$\begin{aligned}
\theta_{21} &= \Pr[X_2 \le q_1 | X_2 \le \mu] \\
\theta_{22} &= \Pr[q_1 < X_2 \le \mu | X_2 \le \mu] \\
\theta_{23} &= \Pr[\mu < X_2 \le q_3 | X_2 > \mu] \\
\theta_{24} &= \Pr[q_3 < X_2 | X_2 > \mu].
\end{aligned}$$

Note that $\theta_{21} = 1 - \theta_{22}$ and $\theta_{23} = 1 - \theta_{24}$. Unconditionally, the four sets have the probabilities

$$\begin{aligned}
\Pr[X_2 \le q_1] &= \theta_{11}\theta_{21} \\
\Pr[q_1 < X_2 \le \mu] &= \theta_{11}\theta_{22} \\
\Pr[\mu < X_2 \le q_3] &= \theta_{12}\theta_{23} \\
\Pr[q_3 < X_2] &= \theta_{12}\theta_{24}.
\end{aligned}$$

Within each set, we again use the shape of the original normal density so, for example, if $\mu < a < b \le q_3$ and $Y \sim N(\mu,\sigma^2)$,

$$\begin{aligned}
\Pr[a < X_2 \le b] &= \Pr[a < X_2 \le b \,|\, \mu < X_2 \le q_3]\Pr[\mu < X_2 \le q_3] \\
&= \Pr[a < Y \le b \,|\, \mu < Y \le q_3]\Pr[\mu < X_2 \le q_3] \\
&= \frac{\Pr[a < Y \le b]}{\Pr[\mu < Y \le q_3]}\Pr[\mu < X_2 \le q_3] \\
&= \Pr[a < Y \le b]\frac{\theta_{12}\theta_{23}}{0.25}.
\end{aligned}$$

In Figure 15.5c the density has $\theta_{11} = 0.45$, $\theta_{21} = 0.4$, $\theta_{23} = 0.6$. In general, the density of the stage 2 distribution is

$$\begin{aligned}
f(x_2|\mu,\sigma^2,\theta_{11},\theta_{12},\theta_{21},\theta_{22},\theta_{23},\theta_{24}) = \frac{2^2}{\sqrt{2\pi}\sigma}e^{-(x_2-\mu)^2/2\sigma^2} \times \\
\left[\theta_{11}\theta_{21}I_{(-\infty,q_1]}(x_2) + \theta_{11}\theta_{22}I_{(q_1,\mu]}(x_2) + \theta_{12}\theta_{23}I_{(\mu,q_3]}(x_2) + \theta_{12}\theta_{24}I_{(q_3,\infty)}(x_2)\right].
\end{aligned}$$

Subsequent stages follow a similar pattern with stage 3 breaking the support of the $N(\mu,\sigma^2)$ distribution into eight sets based on the octiles so that each set has probability $1/8 = 1/2^3$ under the original parametric distribution. One introduces parameters $\theta_{31},\ldots,\theta_{38}$ for the conditional probabilities of these sets given the stage 2 sets. Each parameter whose second subscript is even equals 1 minus the previous parameter, for example $\theta_{32} = 1 - \theta_{31}$ and $\theta_{38} = 1 - \theta_{37}$. Figure 15.5d illustrates stage 3 for $\theta_{31} = 0.3$, $\theta_{33} = 0.3$, $\theta_{35} = 0.6$, $\theta_{37} = 0.3$, and the previous θ_{js}s. One continues these stages to a level J with 2^J sets that each have probability $1/2^J$ under the original parametric distribution and whose conditional probabilities given the stage $J-1$ sets are the parameters $\theta_{J1},\ldots,\theta_{J,2^J}$ in which $\theta_{J,2k-1} = 1 - \theta_{J,2k}$, $k = 1,\ldots,2^{J-1}$.

At the final stage J, the density at a point x_J depends on the string of sets from the various stages that contain x_J. For example, with $J = 3$, if x_3 is between the fifth and sixth octals, that is, if $\mu + 0.3186\sigma < x_3 \le \mu + 0.6745\sigma$, then x_3 is also in the sets $(\mu,\mu+0.6745\sigma]$ and (μ,∞). The corresponding θ parameters are $\theta_{36},\theta_{23},\theta_{12}$. Let $\Theta(x_J)$ be the collection of θ_{js}s corresponding to

the sets containing x_J. There are J such parameters. Define a step function that serves as a weighting factor

$$r(x_J) = 2^J \prod_{\theta_{js} \in \Theta(x_J)} \theta_{js}. \tag{6}$$

For our x_3 between the fifth and sixth octals, $r(x_3) = 2^3 \theta_{36}\theta_{23}\theta_{12}$. The density at stage J is just the product of the weighting factor and the original parametric density, that is,

$$f(x_J|\mu, \sigma^2, \theta_{js}, j = 1, \ldots, J, s = 1, \ldots, 2^j) = \psi(x_J) r(x_J), \tag{7}$$

where ψ denotes the normal density

$$\psi(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(x-\mu)^2/2\sigma^2}.$$

Note that the θ_{js}s only appear in the weight function $r(\cdot)$. Obviously, $\psi(\cdot)$ can be replaced by the density for any other parametric family but corresponding changes in $r(\cdot)$ must be made. This is a highly flexible model because the θ_{js} parameters are numerous. There are $2^{J+1} - 2$ of them, so for $J = 6$ there are 126 and for $J = 8$ there are 510. One often determines J as a function of the number of independent sampling units n. A good practical choice seems to be $J \doteq \log_2(n)$.

To perform a Bayesian analysis with these sampling distributions, we need a joint prior distribution on the θ_{js} parameters. The θ_{js}s are easily interpretable, so meaningful prior information may exist on many of them. An extreme case is choosing $\theta_{11} = \theta_{12} = 0.5$ with prior probability 1. This gives prior probability 1 to the median being μ for any J. Nonetheless, there are far too many parameters to choose a distribution that reflects meaningful prior information on all of the θ_{js}s, so reference priors are also incorporated. Typically, meaningful prior information would be restricted to parameters from the first few stages j.

These θ_{js} parameters are all probabilities, so it is convenient to use beta distributions. When the second subscript is odd, we assume for $j = 1, \ldots, J$, $k = 1, \ldots, 2^{j-1}$ that

$$\theta_{j,2k-1} \sim \text{Beta}(\alpha_{j,2k-1}, \alpha_{j,2k})$$

with $\theta_{j,2k} = 1 - \theta_{j,2k-1}$. In alternative notation, the consecutive pairs have Dirichlet distributions,

$$(\theta_{j,2k-1}, \theta_{j,2k}) \sim \text{Dirich}(\alpha_{j,2k-1}, \alpha_{j,2k}).$$

We assume that all such pairs are independent in the prior.

For any parameters on which meaningful prior information is available, the α_{js}s are chosen to reflect that information. However, in terms of defining a workable prior for all of the θ_{js} parameters, we have not accomplished a great deal. We have reduced the problem of choosing a joint prior on $2^{J+1} - 2$ parameters to choosing $2^{J+1} - 2$ hyperparameters, the α_{js}s. To make this more manageable, for all parameters without meaningful prior information we typically assume that $\alpha_{js} = \alpha\rho(j)$ for some constant α and nondecreasing function $\rho(\cdot)$. For fixed $\rho(\cdot)$ the hyperparameter α indicates strength of prior belief in the original parametric family. Often we take $\rho(j) = j^2$.

One advantage of the $\alpha_{js} = \alpha\rho(j)$ priors is that on average they give the same probabilities as the original parametric distributions. Thus, if $Y \sim N(\mu, \sigma^2)$, X_J has the stage J distribution, and A is any set,

$$\text{E}[\Pr(X_J \in A)] = \Pr(Y \in A)$$

where the expectation is over the prior on the θ_{js} parameters. To see this, first observe that by our construction, if one fixed $\theta_{js} = 0.5$ for all j and s, then clearly $X_J \sim N(\mu, \sigma^2)$ and, in particular, the weight function (6) is $r(x_j) = 1$, so the result follows from (7). With these reference priors, for any x_J, the θ_{js} parameters in $r(x_J)$ are independent with mean 0.5, so $\text{E}[r(x_J)] = 1$ and thus, again using (7), the average density is the normal density. Consequently, these reference priors are particularly

appropriate if we believe the data may come from the original parametric family but want to allow for other possibilities.

Correspondingly, prior or posterior distributions that focus high probability on regions around $\theta_{js} = 0.5$ for all js will behave very much like normal distributions. This occurs whenever α is large in $\alpha_{js} = \alpha\rho(j)$. Moreover, with $\rho(j)$ increasing there is a strong prior tendency when j is large for a new level to behave like the previous level. This tendency can be overcome by data but this tendency also allows us to use numbers of parameters that are comparable to the number of observations in the data without "overfitting" the model.

On the other hand, when α is small, the distribution is more "nonparametric." Let A_J be a set in the Jth level partition. When α is small, an observation in A_J has a large effect on all the posterior beta distributions of θ_{js}s associated with A_J, thus causing high probability for A_J in the posterior distribution. Since A_J is a set in the finest partition considered, this causes jagged, approximating discrete, behavior in the posterior.

The Jth stage generalized sampling distribution, say G, depends on the θ_{js}s and the original $N(\mu,\sigma^2)$ distribution. G together with the reference prior on the θ_{js}s determined by α and ρ defines a random distribution that, because it is random, itself has a distribution called a finite Polya tree, which we write

$$G \sim PT_J(\alpha,\rho,N(\mu,\sigma^2)).$$

A prior on (μ,σ) implies that the median μ, the quartiles, octiles, and so on, are uncertain. This has the effect of smoothing out the abrupt jumps at these points that are noticeable in Figure 15.5. Figure 15.5d contains a realization of a third stage Polya tree that is conditional on $\mu = \mu_0$, $\sigma^2 = \sigma_0^2$, and the θ_{js}s. It also contains a realization of a mixture of a third stage Polya tree that is integrated over μ and σ^2, but still conditional on the specified θ_{js}s. Specifically, $\mu \sim N(\mu_0,(\sigma/10)^2)$ and $\sigma \sim N(\sigma_0,(\sigma_0/10)^2)$.

The distribution on G obtained by randomly generating the $\{\theta_{js}\}$s according to the reference prior with $\alpha_{js} = \alpha\rho(j)$, but averaged over a prior on (μ,σ) is called a *mixture of Polya trees (MPT)*. Write

$$G \sim \int PT_J(\alpha,\rho,N(\mu,\sigma^2))p(\mu,\sigma^2)d\mu\, d\sigma^2.$$

Hanson (2006) shows that for typical priors, for example, $(\mu,\tau) \sim N(\mu_0,V) \times \text{Gamma}(a,b)$, that the random MPT density $g(u) = dG(u)/du$ is smooth. Polya trees and other versions of mixtures of Polya trees do not necessarily have this property, see Barron et al. (1999), Paddock (1999), and Berger and Guglielmi (2001).

EXAMPLE 15.1.5. *Galaxy Data*. For the MPT model on the galaxy data we used $J = 5$, $\rho(j) = j^2$, normal-gamma reference priors, and an additional prior on α, $\alpha \sim \text{Gamma}(5,1)$. The predictive density is given in Figure 15.4. The corresponding LPML is -220.9, which is not very good compared to the mixtures of normals models. That is because the MPT smooths the tails of the distribution more than the mixtures of normals, see for example the MPT density in Figure 15.4 around 15 and around 30.

EXAMPLE 15.1.6. *Toenail Data*. These data were previously examined in Example 8.5.1. Recall that the model there was a *logistic regression* with *random effects* and the goal was to model the probability of moderate or severe toenail separation as a function of time and treatment. Random effects were assumed to be normally distributed and were used to model heterogeneity across individuals in the study and to model correlation among repeated observations on the same individual. Here, we replace the normality assumption on random effects with

$$\gamma_1,\ldots,\gamma_{294}|G \overset{iid}{\sim} G,$$

$$G|\mu,\sigma^2 \sim PT_8(\alpha,j^2,N(\mu,\sigma^2)).$$

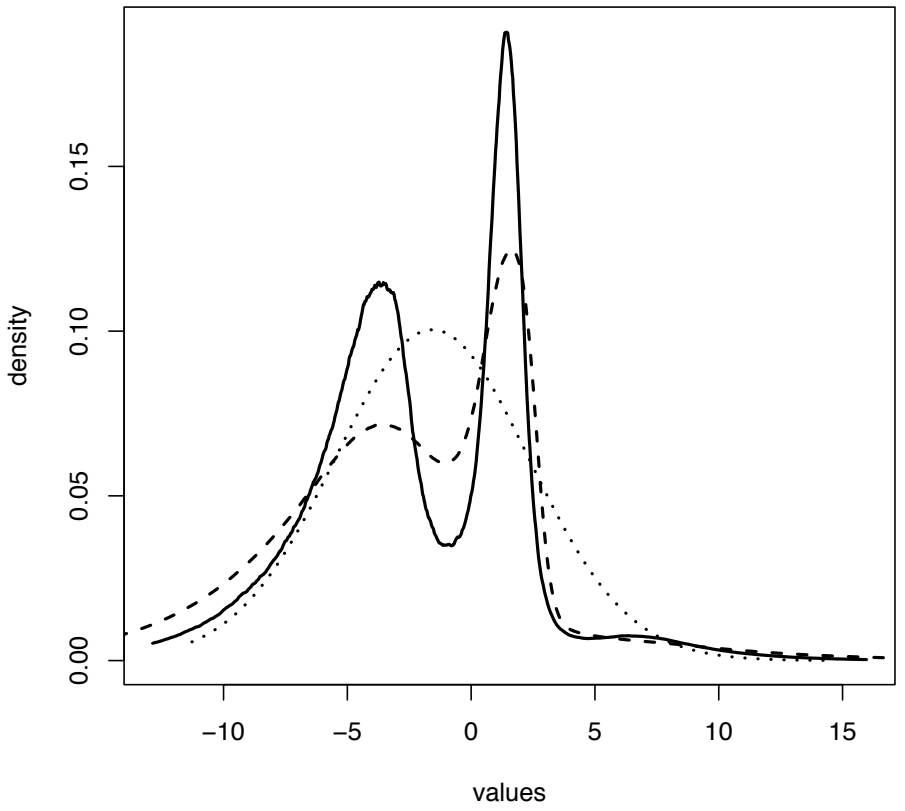


Figure 15.6: *Toenail data estimated random effects distribution under the MPT ($\alpha = 0.1$, solid line), MPT ($\alpha = 1$, dashed line), and normal ($\alpha \to \infty$, dotted line) models.*

Together with the prior on (μ, σ^2) from Example 8.5.1, this constitutes an MPT prior on the γ_is. This methodology allows the evaluation of the normality assumption of the random effects and the implications of potential mis-specification. For example, it allows for multiple modes in the random effects distribution. Multiple modes suggest that there may be distinct subpopulations of individuals in the study whose probabilities of toenail separation differ. Different reference priors for the θ_{js}s were considered using three values of α, $\alpha = 0.1, 1, 10$, to reflect increasing degrees of belief in normality for the random effects.

Figure 15.6 compares the predictive distributions of a new random effect, say γ_{295}, not in the observed data, using the two best MPT models as determined by DIC and LPML (see Table 15.1) and the normal theory model. The plot clearly shows deviation from normality in such a way that the patients could be divided into two or three groups according to their resistance against infection and accompanying toenail separation. Note the smooth appearance of the MPT predictive distributions. The fact that the density of the generalized distributions contains jumps typically washes out in the posterior analysis of an MPT.

Table 15.1 presents results from fitting the normal model and the three MPT models. The Polya trees outperform the normal model using either the LPML or DIC statistic, suggesting that the MPT model is better both for explaining the observed data and from a predictive viewpoint.

Table 15.1 also shows the effects, in this instance, of incorrectly assuming a normal distribution

Table 15.1: *Posterior means, model comparison criteria, and 95% probability intervals for parameters of the toenail GLMM:* $\beta_1(Trt)$, $\beta_2(Time)$, $\beta_3(Trt \times Time)$.

		MPT		
	Normal	$\alpha = 10$	$\alpha = 1$	$\alpha = 0.1$
β_1	−0.159	−0.051	0.292	0.364
β_2	−0.393	−0.390	−0.393	−0.376
β_3	−0.138	−0.136	−0.130	−0.129
μ	−1.604	−1.999	1.510	−0.181
σ^2	16.025	16.728	52.457	23.631
DIC	964.2	954.5	906.3	909.9
LPML	−484.0	−482.5	−465.3	−470.5
		Posterior Intervals		
β_1	(−1.290, 0.966)	(−1.145, 1.090)	(−0.681, 1.186)	(−0.486, 1.198)
β_2	(−0.478, −0.305)	(−0.478, −0.304)	(−0.488, −0.307)	(−0.472, −0.289)
β_3	(−0.271, −0.005)	(−0.270, −0.003)	(−0.274, 0.010)	(−0.266, 0.007)
μ	(−2.473, −0.790)	(−3.544, −0.479)	(−3.402, 3.884)	(−6.869, 5.664)
σ^2	(10.445, 22.053)	(9.562, 24.867)	(8.541, 116.273)	(1.436, 42.767)

for the random intercepts γ_i. The intervals for μ become substantially narrower as the random effects become more normal (α increases). Comparing the interval estimates to 0, none of the models shows a statistically important baseline effect (β_1) for treatments. This should be the case for a randomized experiment because the first measurements should be taken after treatment assignment but before treatments are actually applied or effective. All models show time effects (β_2 different from 0) and that the treatment coded 1 works better over time ($\beta_3 > 0$). The posterior probability of $\beta_3 > 0$ was 2.03%, 2.18%, 3.56%, and 3.24% for $\alpha = \infty$, $\alpha = 10$, $\alpha = 1$, and $\alpha = 0.1$, respectively. Although all models show evidence of differential treatment (β_3) effects, the two MPT models with weak normality assumptions show a reduction in the posterior evidence. Note that $\alpha = 1$ is the best fitting model. It fits better than $\alpha = 0.1$ which is "more nonparametric."

Christensen, Hanson, and Jara (2008) give an example on Ache monkey hunting and discuss full conditional distributions.

EXERCISE 15.6. WinBUGS code for fitting Example 15.1.5 is available in `MPTdensity.odc` on the book website. The code includes computing the CPO statistics as well as the predictive density (and interval estimates) over a grid of points. Duplicate the mixture of Polya trees fit in Figure 15.4. Rerun the code with $J = 6$. Do the predictive density and LPML change much? Try fixing $\alpha = 1$ and $\alpha = 10$ (with $J = 5$). How do the predictive densities and LPML statistics change? How well is the MCMC mixing?

EXERCISE 15.7. The function `PTdensity` from DPpackage fits a finite or *infinite mixture of Polya trees* to data using a reference prior on (μ, σ). Use the function to duplicate the mixture of Polya trees fit in Figure 15.4. Rerun the function with $J = 6$ (called `M` in the function). Do the predictive density and LPML change much? Try fixing $\alpha = 1$ and $\alpha = 10$ (with $J = 5$). How do the predictive densities and LPML statistics change? How do the normal parameters (μ, σ) mix as α gets smaller? Code is in the file `Chap15DPpackage.txt`.

15.2 Flexible Regression Functions

The nonparametric regression problem is typically cast as estimating the mean function $m(\cdot)$ from data $\{(x_i', y_i)\}_{i=1}^n$ in the model

$$y_i = m(x_i) + \varepsilon_i, \qquad \mathrm{E}(\varepsilon_i) = 0,$$

where the ε_is are iid and the x_is are treated as fixed. In some applications the shape of the ε_i distribution is of interest as well. We initially assume x_i is univariate but later discuss the case when x_i is a vector of predictors.

One approach to this problem, the only one we consider, relies on the fact that when m is smooth and defined on a closed bounded set, it can be represented as a linear combination of basis functions. In other words, given basis functions $\{\phi_k\}_{k=1}^{\infty}$, we can write

$$m(x) = \sum_{k=1}^{\infty} \beta_k \phi_k(x).$$

If the basis functions are orthonormal in the sense of having

$$\int \phi_k^2(x)dx = 1; \qquad \int \phi_j(x)\phi_k(x)dx = 0,\ j \neq k,$$

then it is easily seen that for m with $\int [m(x)]^2 dx < \infty$,

$$\beta_k = \int m(x)\phi_k(x)dx.$$

Orthonormal bases make certain common mathematical calculations trivial, but are not required for this approach. Moreover, in practice it is not the integral properties of the functions that are important but rather their properties when evaluated at the finite number of points in our data. Popular choices for $\{\phi_k\}$ are polynomials, the Fourier series (sines and cosines), wavelet bases, spline bases, and B-spline bases. Figure 15.7 presents two cosine basis functions from among $\{\cos[x(k-1)\pi]\}_{k=1}^{\infty}$ and three Haar wavelet basis functions. Standard basis functions are usually defined on the interval $[0,1]$ so, in applications, the predictor variables usually need to be rescaled before they are used.

It is impossible to estimate $\{\beta_j\}_{j=1}^{\infty}$ with finite data. Instead we must rely on an approximation

$$m(x) \doteq \sum_{k=1}^{K} \beta_k \phi_k(x).$$

The basis functions are typically ordered in some fashion from broad functions that indicate a rough trend to functions that model detailed local behavior. Typically, $\phi_1(x) \equiv 1$. If one fixes K and assumes iid normal errors then a standard linear model is obtained:

$$y_i = \beta_1 + \beta_2\phi_2(x_i) + \cdots + \beta_K\phi_K(x_i) + \varepsilon_i, \qquad \varepsilon_i \overset{iid}{\sim} N(0, 1/\tau)$$

with

$$X = \begin{bmatrix} 1 & \phi_2(x_1) & \cdots & \phi_K(x_1) \\ 1 & \phi_2(x_2) & \cdots & \phi_K(x_2) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \phi_2(x_n) & \cdots & \phi_K(x_n) \end{bmatrix}$$

in the characterization $Y = X\beta + e$ of Chapter 9.

EXAMPLE 15.2.1. Brinkman (1981) presents data on the amount of nitric oxide and nitric dioxide in the exhaust of a single-cylinder test engine fueled by ethanol. The response is in μgs (micrograms) per joule and the predictor is a measure of the air-to-fuel ratio, called the equivalence ratio. The data are part of a larger set used throughout MathSoft (1999) to illustrate various smoothing techniques and are available on our website. We restrict the predictor values $\tilde{x}$ to the domain $0.5 \leq \tilde{x} \leq 1.3$ and rescale $\tilde{x}$ into x so that $0.0 \leq x \equiv (\tilde{x} - 0.5)/0.8 \leq 1$. We considered two sets of basis functions both with $K = 5$: the cosine, $\phi_k(x) = \cos[x(k-1)\pi]$, and Legendre polynomial bases. Legendre polynomials are simply a set of orthogonal polynomials. (They can be obtained

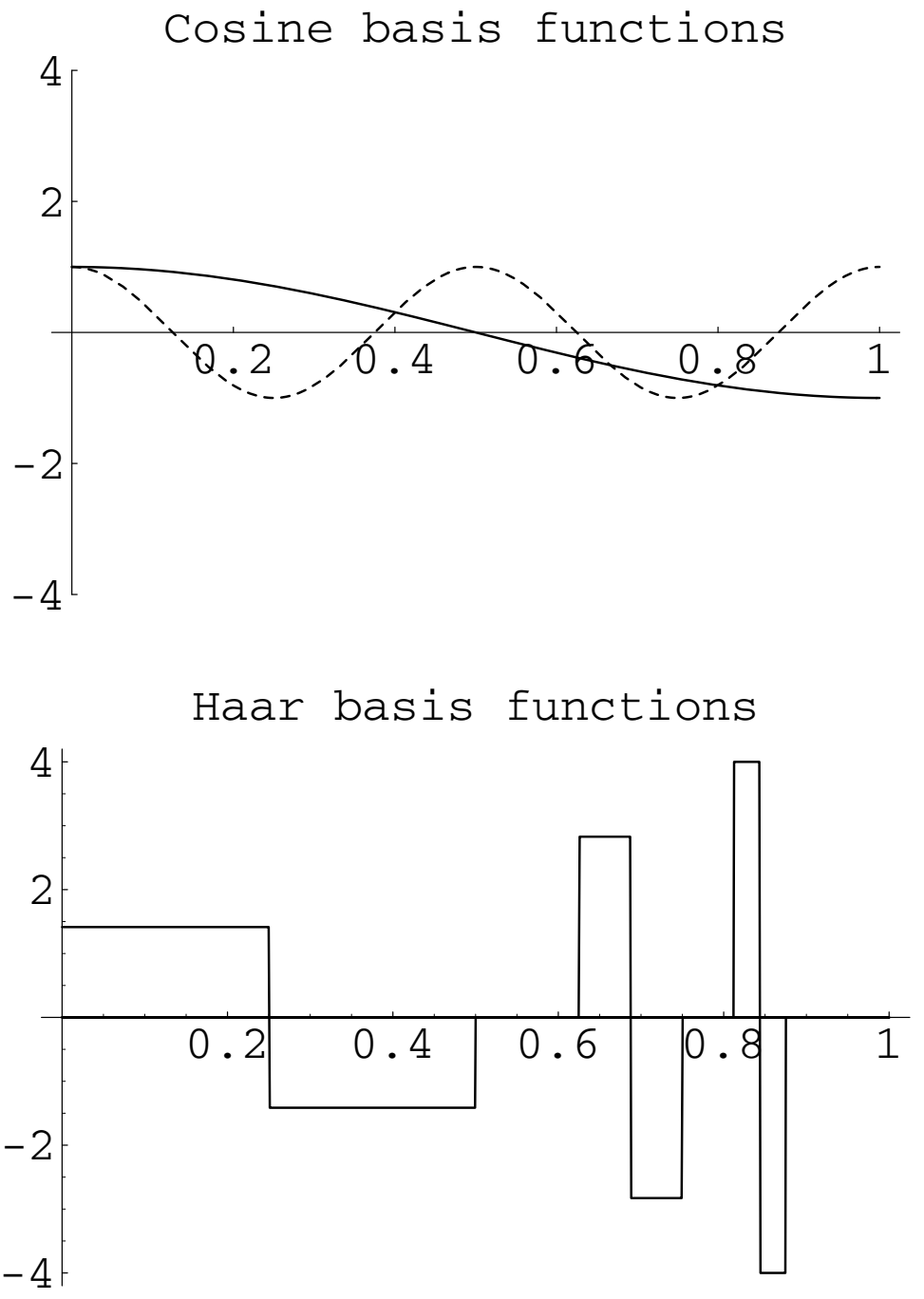


Figure 15.7: *Top: Cosine basis functions* $\cos(x\pi)$ *(solid) and* $\cos(x4\pi)$ *(dashed). Bottom: Three Haar basis functions with supports* $[0,1/2^1]$, $[5/2^3,6/2^3]$, $[13/2^4,14/2^4]$, *i.e.,* $\phi_{2,1}$, $\phi_{4,6}$, *and* $\phi_{5,14}$.

by applying the Gram-Schmidt procedure to the sequence of polynomials $1,x,x^2,\ldots$ defined on the unit interval.) Independent $N(0,1000)$ priors were placed on the regression coefficients independent of the precision prior $\tau \sim \text{Gamma}(0.001,0.001)$. The posterior mean of the regression function and 95% probability intervals are given in Figure 15.8 for $\tilde{x}$ in the domain. Trying to extrapolate results

beyond the range of the data is always dangerous but for these data the figure illustrates that it would be particularly dangerous using the polynomial model. Informative priors could be incorporated as discussed in Chapter 9 but typically K is large enough that only partial prior information would be practical. The choice of basis functions, domain of $\tilde{x}$, and K all affect posterior inference.

When fitting models that use basis functions defined on $[0,1]$, the predictor variable $\tilde{x}$ must be rescaled to x on $[0,1]$. However, in our plots the horizontal axis corresponds to the original scale of the predictor. It is a simple matter to transform any inference about $m(x)$ such as posterior means, medians, or intervals into an inference about the corresponding regression function for $\tilde{x}$. For example, with original data $\{(\tilde{x}_i, y_i) : i = 1, \ldots, n\}$ one often obtains fitted values $\hat{y}_i = \hat{m}(x_i)$. The appropriate plot of the fitted values on the original scale is obtained from $\{(\tilde{x}_i, \hat{y}_i) : i = 1, \ldots, n\}$. When plotting a function q over a grid, one need only keep track of the $\tilde{x}$ value that corresponds to each point $(x, q(x))$ and plot $(\tilde{x}, q(x))$.

Using too many basis functions can be a problem. It is well known that an $(n-1)$ degree polynomial fits data $\{(x_i, y_i)\}_{i=1}^n$ perfectly but that the functions can do very bizarre things at x values between those in the data. This constitutes an example of overfitting, i.e., including too many parameters for the number of observations. Other basis functions like sines and cosines, that are positive almost everywhere on the unit interval, can display many of the more bizarre features encountered when overfitting polynomials. Two approaches to dealing with overfitting are using the data to determine a reasonable value of K or letting K be large but using the data to essentially eliminate individual basis functions. Reasonable values of K are often chosen using model selection criteria. Frequentists often use one equivalent to the C_p statistic, see Christensen (2001a, Section 7.4). Eliminating, or at least deemphasizing, individual basis functions is known as *thresholdling*.

Thresholding requires that substantial data-driven evidence exists for its importance before allowing a basis function into the model. Typically more evidence is required for larger k. One simple approach is to put a prior distribution on each β_k that incorporates positive probability that the parameter is 0. This approach, advocated by Smith and Kohn (1996), can be formulated as writing

$$\beta_k \equiv \gamma_k \beta_k^*$$

where $\gamma_k \sim \text{Bern}(q_k)$. The β_k^*s have continuous distributions and everything is assumed independent. The q_ks are known and are typically decreasing towards 0. For moderate K this can be programmed in WinBUGS. More generally, Bayesian thresholding places mixture priors on basis coefficients that give positive probability to some coefficients being very small or zero. Clyde and George (2004) discuss priors of this type in more detail.

EXAMPLE 15.2.2. For the ethanol data using the cosine basis with $K = 10$ we consider the rather naive prior γ_k iid Bern(0.5). Heretically, in this example we use a prior on β_k^* determined by the data. (Every congregation of four or more contains a heretic.) Using least squares to fit the linear model with all basis functions up to K, let b_k be the least squares estimate of β_k. This has variance $\sigma^2 v_k$. The data based prior has $\beta_k^* \mid \sigma^2 \sim N(b_k, 10\,\sigma^2 v_k)$ and a reference prior on the precision. Figure 15.9 shows the estimate of the regression function on the $\tilde{x}$ scale with 95% probability intervals. Five of the 10 basis functions have posterior probability $\Pr(\gamma_k = 0|y)$ less than the prior value of 0.5.

A popular Bayesian alternative to fixing the number of components when fitting basis function models is to place a prior on K and implement the reversible jump algorithm of Green (1995). *Reversible jump MCMC* approximates posterior inference over a model space where each model has a parameter vector of possibly different dimension. A prior probability is placed on each of $K = 1, 2, \ldots, K_0$, where K_0 is some natural upper bound chosen such that consideration of $K > K_0$ would be superfluous. Reversible jump for the regression problem (in the context of a spline basis) is discussed in Denison et al. (2002) and used, for example, by Mallick et al. (1999) and Holmes and Mallick (2001).

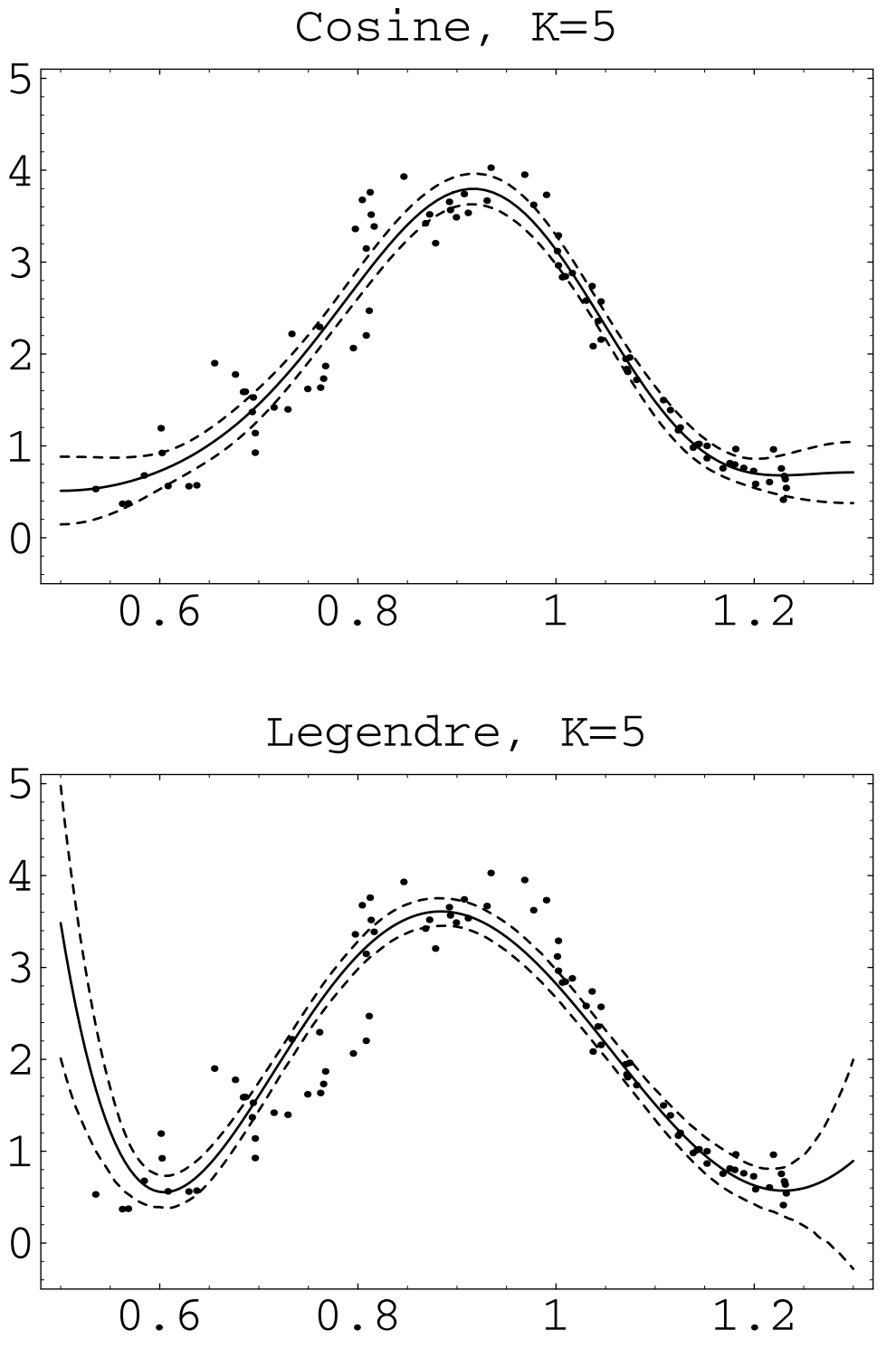


Figure 15.8: *Ethanol data: Mean function estimate using cosine and Legendre bases,* $K = 5$.

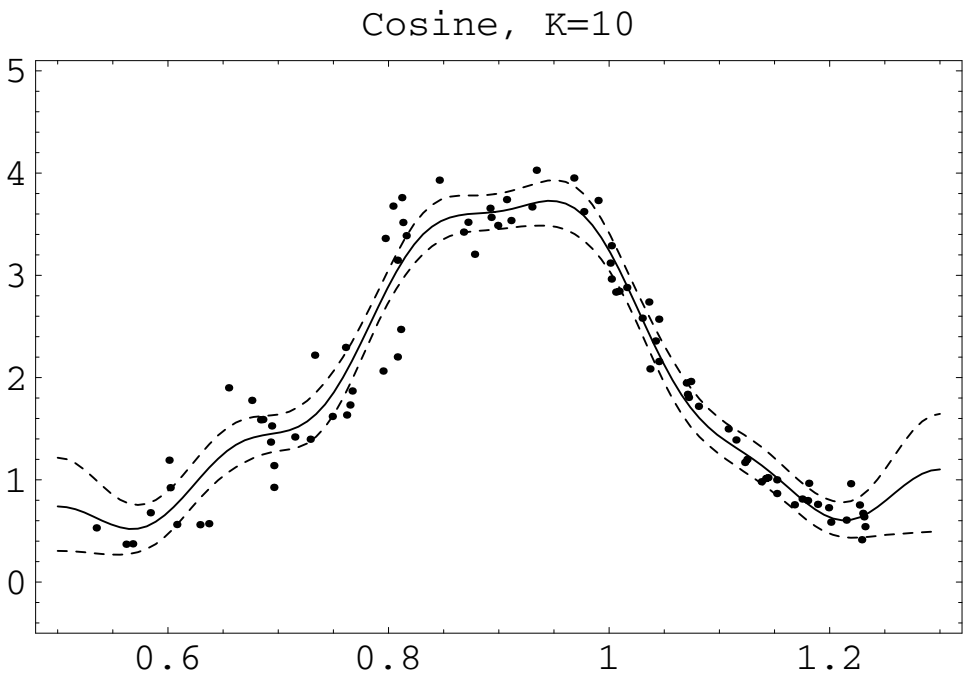


Figure 15.9: *Ethanol data: Estimates of regression mean functions using a cosine basis and thresholdling.*

An unusual class of bases are wavelet bases. Wavelets are useful for modeling functions whose behavior changes dramatically at different locations. Such functions are sometimes called "spatially inhomogeneous." Think of a topographical map of the western United States. Much of the map will have relatively flat homogeneous areas but at the edges of mountains the scale changes abruptly. Wavelets can capture such phenomena and so are used extensively in image processing. A nice, short introduction to Bayesian wavelets and thresholding is Vidakovic (1998). Müller and Vidakovic (1999) discuss Bayesian wavelet modeling in detail.

The key feature of wavelets is that they are 0 except over an increasingly smaller range of x values. The simplest wavelet basis was developed by Haar (1910). On the interval $[0,1]$ fitting the Haar basis to level $k-1$ is equivalent to fitting a step function in which each step has length $1/2^k$, that is, each step is a multiple of an indicator function $I_{(\{j-1\}/2^k, j/2^k]}(x)$, $j = 1,\ldots,2^k$. Rather than using increasingly smaller indicator functions, the Haar wavelet basis combines indicator functions so that the individual basis functions will be orthonormal in the sense defined earlier.

The Haar wavelet basis (as well as other wavelet bases) is conveniently enumerated using a double index. The basis functions can be derived from a *mother wavelet* ψ that for the Haar basis is defined as

$$\psi(x) = \left\{ \begin{array}{cc} 1 & 0 \le x < 0.5 \\ -1 & 0.5 \le x \le 1 \\ 0 & \text{otherwise} \end{array} \right\}.$$

Individual wavelet basis functions are defined through $\phi_{kj}(x) = \psi(2^k x - j)2^{k/2}$ for $k = 0,\ldots,\infty$, and $j = 0,\ldots,2^k - 1$. Thus $\phi_{kj}(x)$ is nonzero only on the interval $[j/2^k, (j+1)/2^k]$. Figure 15.7 showed three of the Haar basis functions: $\phi_{1,1}$, $\phi_{3,6}$, and $\phi_{4,14}$. The orthonormal basis consists of the Haar wavelets together with the *father wavelet* $I_{[0,1]}(x)$. With the father corresponding to the

intercept, the wavelet basis form for the regression function is

$$m(x) = \beta_0 + \sum_{k=0}^{\infty} \sum_{j=0}^{2^k-1} \beta_{kj}\phi_{kj}(x).$$

In practice, $k = 0, \ldots, K$. In this notation k indexes the scale of the basis function whereas j determines location. The most important ideas in the definition of wavelets are that the support of the mother function is bounded and that the subsequent (children) functions are defined as indicated by reductions in scale and by translations. Having orthogonal ϕ_{kj} functions is of little importance.

It seems that just about any function that is 0 off of the unit interval can be used as a mother wavelet ψ. Some commonly assumed properties are that ψ is continuous and that it integrates to 0 over the unit interval. Many commonly used mother wavelets are continuous but cannot be written in closed form. We present a few continuous wavelets that can be written down. These functions might be adjusted so that they integrate to 0 on the unit interval (not important) but they need to be defined so that they are (for practical purposes) 0 outside the unit interval. Shannon's mother is

$$\psi(x) = \frac{\sin(2\pi x) - \sin(\pi x)}{\pi x} I_{[0,1]}(x).$$

A wavelet mother shaped like a Mexican hat is given by

$$\psi(x;\sigma) = \frac{1}{\sqrt{2\pi}\sigma^3}\left(1 - \frac{(x-0.5)^2}{\sigma^2}\right)\exp[-(x-0.5)^2/2\sigma^2].$$

This is essentially the negative second derivative of a normal density and depends on a scaling hyperparameter σ. The Mexican hat can also be approximated by a difference of Gaussian (normal) densities using two scaling parameters

$$\psi(x;\sigma_1,\sigma_2) = \frac{1}{\sigma_1\sqrt{2\pi}}\exp\left(-\frac{(x-0.5)^2}{2\sigma_1^2}\right) - \frac{1}{\sigma_2\sqrt{2\pi}}\exp\left(-\frac{(x-0.05)^2}{2\sigma_2^2}\right).$$

Finally, a wavelet mother based on the beta distribution is

$$\psi(x|\alpha,\beta) \propto \left[\frac{\beta-1}{1-x} - \frac{\alpha-1}{x}\right] x^{\alpha-1}\cdot(1-x)^{\beta-1} I_{[0,1]}(x).$$

For large k, wavelet basis functions can model very localized behavior. In Figure 15.7, contrast the Haar basis functions to the cosine basis functions that oscillate over the entire region. Wavelets can model highly inhomogeneous functions but also require extra care to ensure that mean estimates do not follow the data too closely, i.e., avoid overfitting. For moderate K you might fit polynomials or trig functions without thresholding but you probably don't want to fit wavelets without thresholding.

EXAMPLE 15.2.3. We fit the Haar wavelet model to the ethanol data obtaining a step function with steps of length $1/16$ on the x scale. The model is

$$y_i = \beta_0 + \sum_{k=0}^{3}\sum_{j=1}^{2^k} \beta_{kj}\phi_{kj}(x_i) + \varepsilon_i, \qquad \varepsilon_i \overset{iid}{\sim} N(0,\sigma^2).$$

To incorporate thresholding, we introduce 0-1 parameters γ_{kj} that determine whether ϕ_{kj} is included into the model

$$y_i = \beta_0 + \sum_{k=0}^{3}\sum_{j=1}^{2^k} \gamma_{kj}\beta^*_{kj}\phi_{kj}(x_i) + \varepsilon_i, \qquad \varepsilon_i \overset{iid}{\sim} N(0,\sigma^2).$$

We use independent Bern$(1/2^k)$ priors on the γ_{kj}s, so that it becomes progressively more difficult

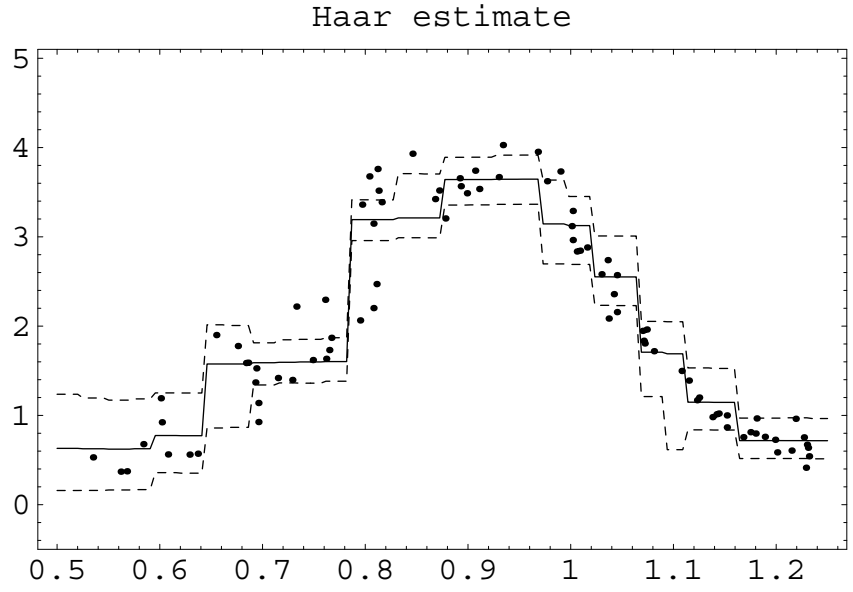


Figure 15.10: *Ethanol data: Estimates of regression mean functions using Harr wavelets.*

to include higher order ϕ_{kj} functions. We use independent prior distributions, $N(0,1000)$ on the β^*_{kj}s and Gamma$(0.001,0.001)$ on the precision. Figure 15.10 shows the resultant mean function estimate on the $\tilde{x}$ scale with 95% probability intervals. Four of the 16 basis functions had posterior probabilities less than 0.1 of being included in the model.

Another approach to nonparametric regression is the use of splines. The fundamental idea of using splines is simply connecting the dots in an x,y plot. Suppose we have data (x_i,y_i), $i = 1,\ldots,n$ in which the x_is are ordered from smallest to largest. Using linear splines to fit a regression function simply fits a line segment between the consecutive pairs of points. Cubic splines fit a cubic polynomial between every pair of points rather than a line. The reason for using cubic splines is to make the curve look smooth. Although there is only one line you can fit between two points, there are many cubic polynomials. You pick the cubic polynomials so that the overall curve connecting all the data points has continuous first and second derivatives. These conditions determine what each individual cubic polynomial must be. Note that all of the action here has to do with what the function looks like between the data points. All of the data points are fitted perfectly.

EXERCISE 15.8. Find a cubic spline function for connecting the points $(0,0)$, $(1,1/3)$, $(2,4/3)$, $(3,1/3)$, and $(4,0)$. Hint: The first function is $(x^3/3)I_{[0,1]}(x)$. The value, first, and second derivatives of the second function must agree with those of the first function at $x = 1$ and the second function is maximized at $x = 2$ providing four equations for the four unknown parameters of the second cubic polynomial.

Of course, subtleties are often added. Instead of connecting the dots at the data points, the polynomials can be connected at other points called *knots*. Many spline bases are built from truncated polynomials. For example $\{(x-a_j)^3_+\}^J_{j=1}$ is a subset of a cubic spline basis where $\{a_j\}^J_{j=1}$ are the knots and $(x)_+$ is equal to x when $x > 0$ and equal to 0 otherwise. With splines, it is not crucial to rescale the predictor between 0 and 1. Also, regression coefficients can be penalized, that is, shrunk towards some predetermined value (often 0) to give imperfect fitting of the data but presumably superior predictive ability. Penalties exist to avoid overfitting. Priors serve the same purpose

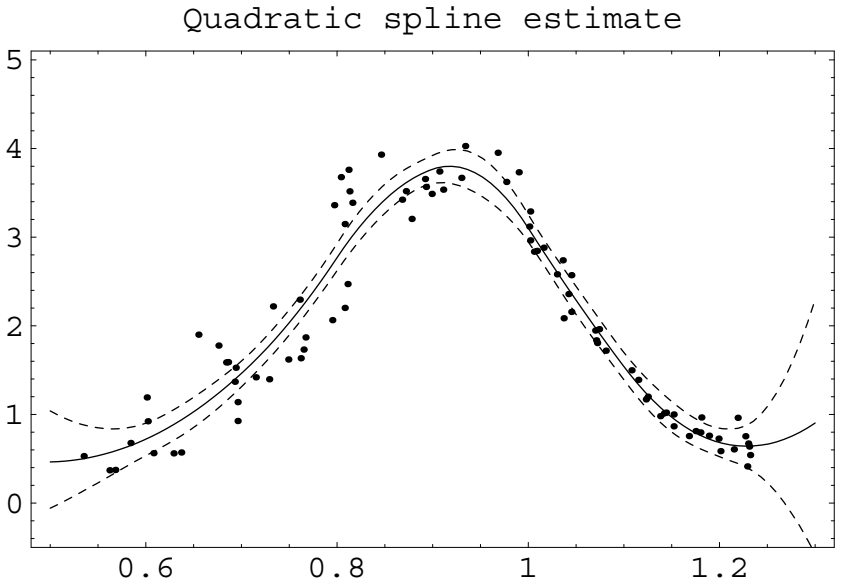


Figure 15.11: *Ethanol data: Estimate of mean regression function using quadratic splines.*

as penalizing regression coefficients because they shrink coefficients towards the center of the prior distribution. The extent of the penalty relates to the prior variability.

EXAMPLE 15.2.4. Crainiceanu et al. (2004) outline a strategy for fitting spline models in WinBUGS. We apply their approach by fitting a quadratic spline model to the ethanol data. Specifically, the model is

$$y_i = \beta_0 + \beta_1 \tilde{x}_i + \beta_2 \tilde{x}_i^2 + \sum_{k=1}^{9} b_k (\tilde{x}_i - a_k)_+^2 + \varepsilon_i,$$

where $b_k|\sigma_b \overset{iid}{\sim} N(0,\sigma_b^2)$ independent of $\varepsilon_i|\sigma_\varepsilon \overset{iid}{\sim} N(0,\sigma_\varepsilon^2)$. Here, the knots $\{a_k\}_{k=1}^9$ are defined as $a_i = 0.4 + 0.1i$, evenly spaced over the range of the predictor variable. Figure 15.11 gives the spline estimate along with 95% pointwise probability intervals.

Another popular approach is the use of *basis splines* or *B-splines*, see Lang and Brezger (2004) or Jara et al. (2009) for Bayesian treatments. Although B-splines borrows an idea from splines, it is more similar to using wavelets in that the individual basis functions are 0 outside of a small interval. The "mother" function for a B-spline basis of degree 2 is nonzero over $[0,3]$ and defined as

$$\psi(x) = \frac{x^2}{2} I_{[0,1]}(x) - \left\{[x-1.5]^2 - 0.75\right\} I_{[1,2]}(x) + \frac{[3-x]^2}{2} I_{[2,3]}(x).$$

This is a bell-shaped curve, similar to a normal density centered at 1.5, but it is 0 outside the interval $[0,3]$ while still being smooth in that it is differentiable everywhere. The "spline" in B-spline is because ψ is a quadratic spline function, i.e., quadratics have been pasted together as a smooth function. A B-spline basis mother function ψ of degree d splices together $(d+1)$ different d-degree polynomials over a finite interval so that the whole function is differentiable $d-1$ times and looks like a mean-shifted Gaussian density, but nonzero only on a finite interval. Commonly d is either 2 or 3.

Although ψ could be used as a perfectly reasonable mother function for fitting wavelets, typically B-splines are used a bit differently. Nonetheless, the basis functions ϕ_k are simply scale reduced and location-shifted versions of $\psi(x)$.

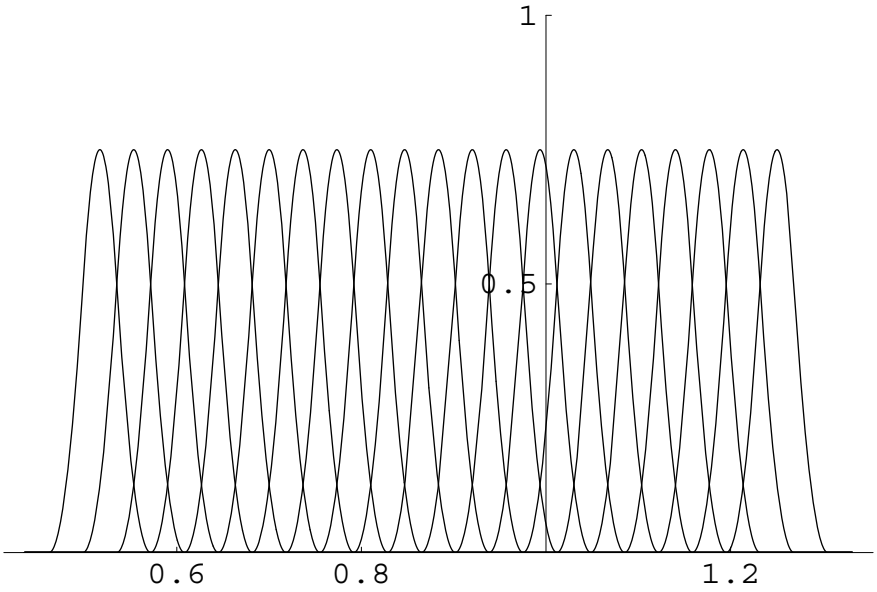


Figure 15.12: $K = 21$ *quadratic B-spline basis functions* $\{\phi_1(\cdot), \ldots, \phi_{21}(\cdot)\}$ *(from left to right) equispaced over* $[0.535, 1.232]$.

EXAMPLE 15.2.5. Figure 15.12 shows $K = 21$ quadratic B-spline basis functions equally spaced over the range of the equivalence ratio predictor x for the ethanol data.

If $[x_l, x_u]$ is the range of the observed predictor variable, the basis functions $\phi_k(x)$ are defined so that they are centered at equally spaced locations, they have substantial overlap with one another, and they "bleed" over the ends of the interval $[x_l, x_u]$. There are many ways to do this and it does not seem to matter very much how you do it or even if you use a B-spline ψ function, as opposed to, say, a normal density. However, we employ the traditional B-spline approach.

The idea is best illustrated with an example. Suppose we want to use $K = 5$ quadratic basis functions to cover the unit interval, i.e., $[0,1] = [x_l, x_u]$. Divide the unit interval into $K - 2 = 3$ subintervals: $[0,1/3], [1/3,2/3], [2/3,1]$. Determining these subintervals is the only place that K enters the process. Add two more comparable intervals onto each end of these so we have

$$[-2/3,-1/3], [-1/3,0], [0,1/3], [1/3,2/3], [2/3,1], [1,4/3], [4/3,5/3].$$

Just as the ψ function is defined over three subintervals of $[0,3]$, rescale and translate ψ to give a first basis function ϕ_1 that covers the three intervals $[-2/3,-1/3]$, $[-1/3,0]$, $[0,1/3]$. Similarly, ϕ_2 covers the three subintervals contained in $[-1/3,2/3]$, ϕ_3 covers $[0,1]$, ϕ_4 covers $[1/3,4/3]$, and ϕ_5 covers $[2/3,5/3]$. For any K, each of the subintervals of our target interval $[0,1]$ has three nonzero ϕ_k functions defined over it. For K quadratic basis functions on an arbitrary interval $[x_l, x_u]$ this procedure becomes

$$\phi_k(x) = \psi\left\{\frac{x - x_l}{\Delta} + 3 - k\right\}, \tag{1}$$

where $\Delta = (x_u - x_l)/(K-2)$ and $k = 1, \ldots, K$. The B-spline regression function is taken as

$$m(x) \doteq \beta_0 + \sum_{k=1}^{K} \beta_k \phi_k(x). \tag{2}$$

EXERCISE 15.9. For $d = 3$, the cubic spline mother function is

$$\psi(x) = \frac{x^3}{3} I_{[0,1]}(x) + \left\{ -x^3 + 4x^2 - 4x + \frac{4}{3} \right\} I_{[1,2]}(x) + \left\{ -[4-x]^3 + 4[4-x]^2 - 4[4-x] + \frac{4}{3} \right\} I_{[2,3]}(x) + \frac{[4-x]^3}{3} I_{[3,4]}(x).$$

For $K = 5$ over $[x_l, x_u] = [0,1]$, find the cubic B-spline basis functions. Can you find a formula for arbitrary K and $[x_l, x_u]$ similar to equation (1)? How does ψ relate to your answer to Exercise 15.8?

The key characteristic of B-splines is that, similar to wavelets, the basis functions are 0 off of intervals that become increasingly small. Technically, for B-splines to constitute a basis function approach, we should be able to write the regression function exactly as an infinite linear combination of basis functions. To do that we would need to use a double index for the basis functions like we did for wavelets. The definition of ϕ_k in (1) actually depends on K as well. With

$$\phi_{Kk}(x) \equiv \phi_k(x),$$

we could write

$$m(x) = \beta_0 + \sum_{K=1}^{\infty} \sum_{k=1}^{K} \beta_{Kk} \phi_{Kk}(x). \tag{3}$$

Using a finite approximation to this, as with our other basis functions, the higher order terms (K large) are less "smooth," so there is a premium on shrinking the corresponding regression coefficients towards 0 to smooth any fitted model.

In practice, model (2) is used rather than an approximation to (3), so shrinking coefficients towards 0 seems inappropriate. All of the functions in (2) have the same level of smoothness. To ensure smoothness in the fitted model, we instead cause the regression coefficients to be similar for similar values of k.

One way to achieve smoothness is by using a first-order *random walk prior*. Specifically, with an informative or reference prior on β_0, let

$$\beta_k | \beta_0, \beta_1, \ldots, \beta_{k-1}, \lambda \sim N(\beta_{k-1}, 1/\lambda), \quad k = 1, 2, \ldots, K.$$

Alternatively, we could have standard priors on β_0 and β_1 with the random walk starting at $k = 2$. The larger λ is, the smaller the jumps can be between neighboring basis functions, and therefore $m(x)$ in (2) is smoother. The parameter λ is often called a penalty term and has a frequentist interpretation related to fitting B-spline models through penalized likelihoods (Eilers and Marx, 1996). It is possible to elicit reasonable prior information on how "jumpy" the trends are, or one could employ a reference prior on λ. A reference prior allows λ to reflect an overall level of smoothness in the regression, but can yield quite bumpy fits if there are marked jumps in the data.

With the regression model defined by (2), β_0 is nonidentifiable if and only if for every x_i there exist real numbers α_k such that $\sum_{k=1}^{K} \alpha_k \phi_k(x_i) = 1$, see Christensen (2002, Proposition 2.1.6). In practice, this is extremely unlikely to happen unless the ϕ_ks have been defined so that there exist α_ks with $\sum_{k=1}^{K} \alpha_k \phi_k(x) = 1$ for all $x \in [x_l, x_u]$. The basis functions in (1) have this property, see Exercise 15.10. In fact, for any way of defining basis functions that satisfy the criteria of small support, overlapping functions, and bleeding over the ends, it is quite likely that there exist α_ks with $\sum_{k=1}^{K} \alpha_k \phi_k(x_i) \doteq 1$, causing a severe collinearity problem. A likely symptom of this is horrendous mixing problems within the MCMC. Both the identifiability and the numerical difficulty can be alleviated by imposing a side condition. The condition $\sum_{k=1}^{K} \beta_k = 0$ is often used but, theoretically, dropping the intercept or any one of the ϕ_k functions should also work. None of these side conditions should have any effect on the fitted regression function (except through using different priors).

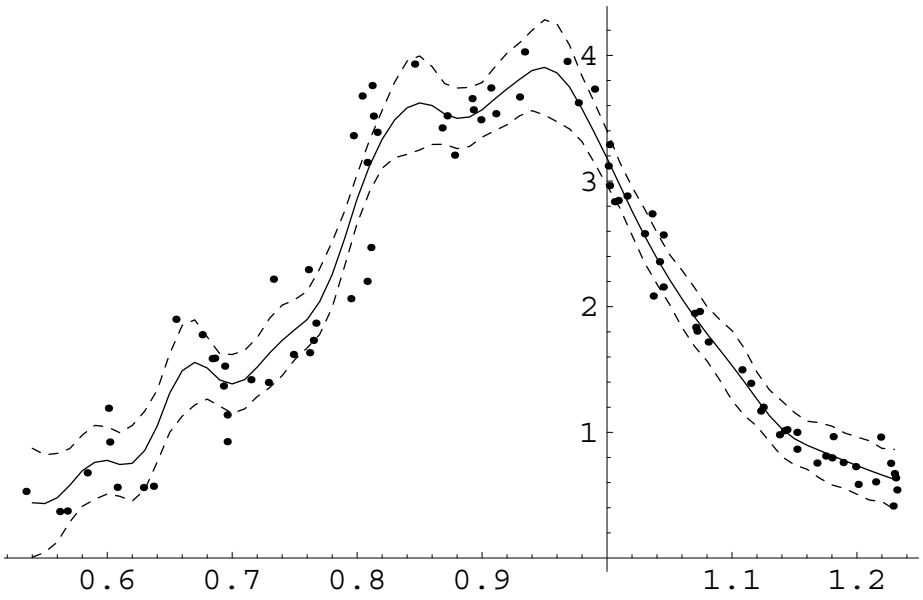


Figure 15.13: *Estimated trend using quadratic B-splines with $K = 21$ knots.*

EXERCISE 15.10. For the basis functions in (1), show that $\sum_{k=1}^{K} \phi_k(x_i)$ is a constant. Hint: Recall that exactly three nonzero basis functions overlap on every subinterval and argue that it is enough to show that for $x \in [0,1]$ the function $x^2/2 - [(1+x) - 1.5]^2 + [3-(2+x)]^2/2$ is a constant.

Eilers and Marx (1996), citing de Boor (1977), indicate that when using the basis functions in (1) there also exist different sets of α_ks that for all $x \in [x_l, x_u]$ make $\sum_{k=1}^{K} \alpha_k \phi_k(x) = x$ and make $\sum_{k=1}^{K} \alpha_k \phi_k(x) = x^2$. This implies that the quadratic B-spline basis can fit second degree polynomials. Using a B-spline ψ function of degree d, d is also the degree of the polynomial that can be fitted.

EXAMPLE 15.2.6. *Ethanol Data.* The model

$$y_i = \beta_0 + \sum_{k=1}^{21} \beta_k \phi_k(x_i) + \varepsilon_i$$

was fitted to the ethanol data with a random walk prior determined by

$$\beta_0 \sim N(0, 100000) \quad \perp\!\!\!\perp \quad \lambda, \tau \overset{iid}{\sim} \text{Gamma}(0.0001, 0.0001).$$

The posterior mean regression function is given in Figure 15.13 along with a 95% interval. Here $[x_l, x_u] = [0.535, 1.232]$.

Multivariate predictors $x_i = (x_{i1}, \ldots, x_{ir})'$ can be accommodated into series expansions by considering products of univariate basis functions. For example, in two dimensions with $x = (x_1, x_2)'$, simple products are formed as $\phi_{jk}(x_1, x_2) = \phi_j(x_1)\phi_k(x_2)$. The regression model is then

$$y_i = \sum_{j=1}^{K_1} \sum_{k=1}^{K_2} \beta_{jk} \phi_{jk}(x_{i1}, x_{i2}) + \varepsilon_i.$$

Unfortunately, these methods quickly run into a "curse of dimensionality." With $r = 1$ predictor variable, it might take, say, $K = 8$ parameters to get an adequate approximation to a regression

function $m(\cdot)$. With $r = 2$ predictor variables, we could expect to need approximately $K^2 = 8^2 = 64$ parameters in the linear model. Given a few hundred observations, that is doable. However, with $r = 5$ predictor variables, we could expect to need about $K^5 = 8^5 = 32,768$ parameters.

One way to avoid this problem is to fit *generalized additive models*, see Hastie et al. (2001). For example, with three predictor variables, $x = (x_1, x_2, x_3)'$, we might expect to need $K^3 = 8^3 = 512$ terms to approximate $m(\cdot)$. To simplify the problem, we might assume that $m(\cdot)$ follows a generalized additive model such as

$$m(x) = m_1(x_1) + m_{23}(x_2, x_3). \tag{4}$$

We might further approximate

$$m_1(x_1) = \sum_{k=1}^{K} \beta_{1k}\phi_k(x_1) \quad \text{and} \quad m_{23}(x_2, x_3) = \sum_{j=1}^{K}\sum_{k=1}^{K} \beta_{23jk}\phi_j(x_2)\phi_k(x_3).$$

If we need 8 terms to approximate $m_1(\cdot)$ and 64 terms to approximate $m_{23}(\cdot,\cdot)$, the generalized additive model (4) involves fitting only 72 parameters rather than 512. With the 8 term approximations and 5 predictor variables, a generalized additive model that includes all of the possible $m_{jk}(\cdot,\cdot)$s involves only 640 terms, rather than the 32,768 required by a full implementation of a nonparametric regression.

EXERCISE 15.11. The function `PSgam` in DPpackage fits generalized additive models using B-splines. There is another predictor for the nitrogen oxides response y_i in the ethanol data besides the equivalence ratio x_i, namely the compression ratio z_i. Look at a scatterplot matrix of the three variables. Use the `PSgam` function to fit several models $y_i = m(x_i, z_i) + \varepsilon_i$: (a) $m(x_i, z_i) = \beta_0 + m_1(x_i) + m_2(z_i)$ where $m_1(\cdot)$ and $m_2(\cdot)$ are both modeled using B-splines, (b) $m(x_i, z_i) = \beta_0 + m_1(x_i)$ using equivalence ratio only as in all of the examples thus far, and (c) $m(x_i, z_i) = \beta_0 + m_1(x_i) + \beta_2 z_i$, nonlinear in x_i but linear in z_i. Compare the fits qualitatively and through LPML. Code is provided on our website in `Chap15DPpackage.txt`. Note that default plots obtained from `PSgam` remove the overall intercept β_0 and force the functions $m_i(\cdot)$ to integrate to 0 for identifiability, so the vertical scale of the plots will be different than the figures in this chapter for the ethanol data. See the comments on identifiability near the end of Section 3.

EXERCISE 15.12. The compression ratio z_i was observed at only 5 levels. What happens to the LPML when this variable is treated as categorical instead of continuous?

EXERCISE 15.13. Use the `PSgam` function to explore the "optimal" transformation of temperature in the O-ring data of Chapter 8 using a logistic link. Specifically, fit the model

$$\log \frac{\theta_i}{1-\theta_i} = \beta_0 + m(T_i),$$

where T indicates temperature and $m(T)$ is modeled using cubic B-splines with $K = 20$. Play with the prior on the penalty parameter $\lambda \sim \text{Gamma}(\tau_{b1}, \tau_{b2})$. How does the posterior trend change for priors that favor *large* values of λ? Compare LPML for the best fitting B-spline model to LPML for simple regression models fitted using the logit, probit, and complementary log-log links.

15.3 Proportional Hazards Modeling

We now introduce alternatives to the step function model of Subsection 13.2.2 for the baseline hazard in Proportional Hazards models. Several authors (e.g., Royston, 2001; Hennerfeind et al.,

2006; Li, Hu, and Greene, 2009) have advocated modeling the log of the baseline hazard $h_0(t)$ using flexible smooth functions such as those discussed in Section 2. Those methods are restricted to estimating functions over a finite interval which is a problem for estimating hazard functions. As discussed in Subsection 13.2.1, a hazard function must integrate to infinity. If the survival times have an upper bound, that means that the hazard function should approach infinity as time approaches the upper bound. None of the methods in Section 2 accommodate that requirement. As a practical matter, we just pick an upper bound that contains the observed data and admit that we have no idea what the hazard function looks like beyond that point.

BayesX (Belitz et al., 2009) is a flexible, easy to use, and free Windows-based program for fitting generalized additive mixed models with structured (e.g., spatially dependent) random effects, primarily written by Christiane Belitz, Andreas Brezger, Thomas Kneib, and Stefan Lang. It is available for download at `http://www.stat.uni-muenchen.de/~bayesx/bayesx.html`. BayesX provides computer functions for fitting proportional hazards models with the baseline hazard $h_0(t)$ modeled as a piecewise constant like (13.2.4) or $\log[h_0(t)]$ modeled using cubic B-splines. Generalizations include additive models, time-varying regression coefficients, time-dependent covariates, and exchangeable and spatially referenced random effects or frailty terms.

Without delving into great detail, that is, without discussing likelihoods and priors, we fit a PH model to the leukemia data of Chapter 12 using the B-spline option in BayesX. Write the model as

$$h(t|x,\beta) = h_0(t)e^{x\beta} = e^{\beta_0+m_0(t)}e^{x\beta_1}, \tag{5}$$

where $x = 0, 1$ denotes AG− or AG+. We also fit a non-PH time varying hazards ratio model

$$h(t|x,\beta) = e^{\beta_0+m_1(t)}e^{xm_2(t)}. \tag{6}$$

Here $m_0(t)$, $m_1(t)$, and $m_2(t)$ are all modeled using cubic B-splines. The functions $m_0(t)$ and $m_1(t)$ integrate to 0 but $m_2(t)$ has no such restriction. There are identifiability issues associated with these functions but they are completely unrelated to the identifiability issue associated with B-splines. These issues are discussed at the end of the section.

Posterior results for the proportional hazards model (5) include

node	mean	sd	2.5%	med	97.5%
β_0	−2.804	0.408	−3.641	−2.782	−1.993
β_1	−1.304	0.432	−2.138	−1.308	−0.445

Figure 15.14 contains the posterior median along with 80% and 95% pointwise PIs for the zero-centered log-baseline hazard. Note the overall "bathtub" shape indicating elevated hazard at the beginning and the end of the study period. It appears to be just barely possible to fit a flat hazard (for an exponential model) between the 95% limits.

For the model (6), Figures 15.15 and 15.16 give medians and intervals for $m_1(t)$ and $m_2(t)$, respectively. For the simple two-group case, model (6) is equivalent to fitting two distinct hazard functions, one for each group. Figure 15.15 gives estimated deviations from the "average" log-hazard for the $AG-$ group. Adding the posterior median $\tilde{\beta}_0 = -2.782$ to the posterior medians $\tilde{m}_1(t)$ from Figure 15.15 provides an estimate of the log-hazard for people with $AG-$. Figure 15.16 gives the deviations of $AG+$ log-hazards from the $AG-$ log-hazards. The $AG+$ deviations tend to be negative, so log-hazards are lower for $AG+$s, reducing the $AG+$ hazards by a multiplicative effect relative to the $AG-$ hazards, the amount of the reduction changes with time. With this model, we could easily have the $AG+$ differential log-hazards change from negative to positive over time, something that cannot happen in a proportional hazards model. Flat hazards fit easily between the 80% limits in Figures 15.15 and 15.16, suggesting the plausibility of a two-group exponential model.

The DIC for the PH model (5) is 300.6 whereas the more complex model (6) gives 301.8. The simpler model is preferred, so the group effect can be described in a single hazard ratio estimated as

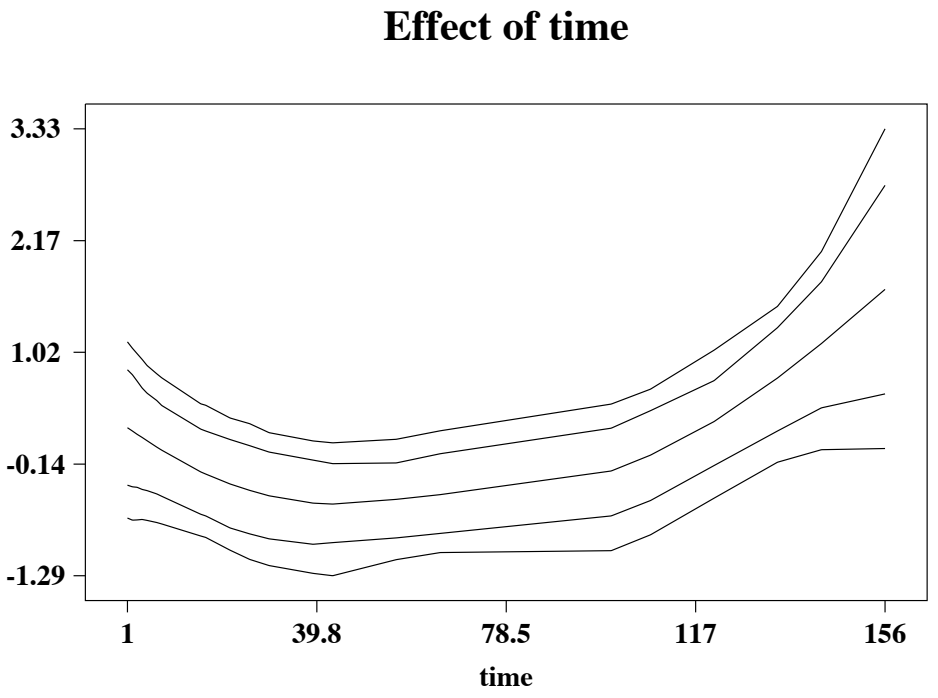


Figure 15.14: *Zero-centered log-baseline hazard $m_0(t)$ of model (5). Posterior median, 80%, and 95% intervals.*

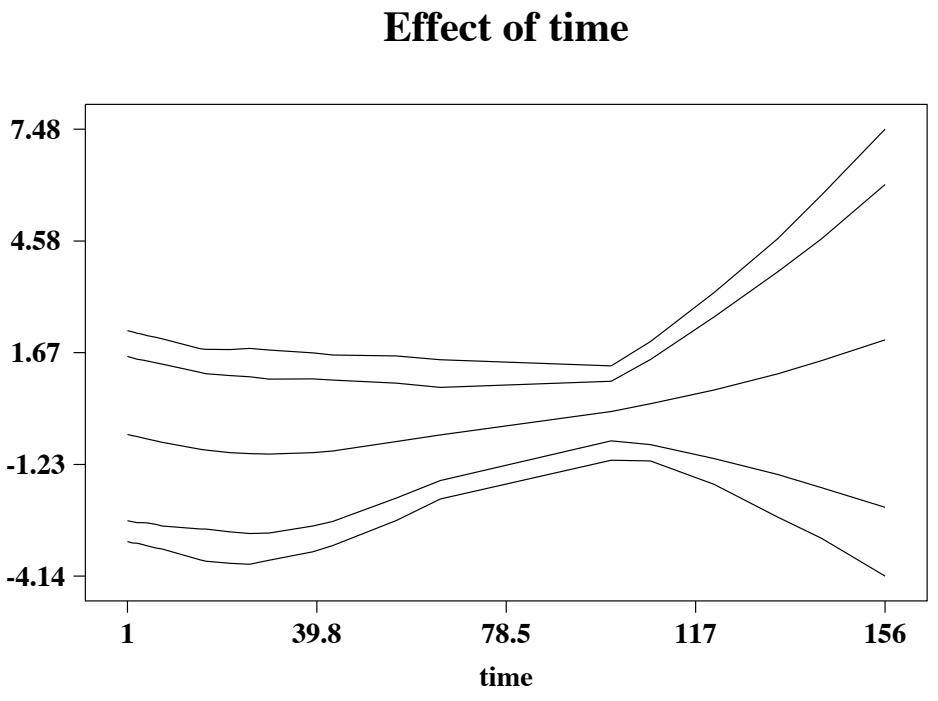


Figure 15.15: *Zero-centered log-baseline hazard $m_1(t)$ of model (6). Posterior median, 80%, and 95% intervals.*

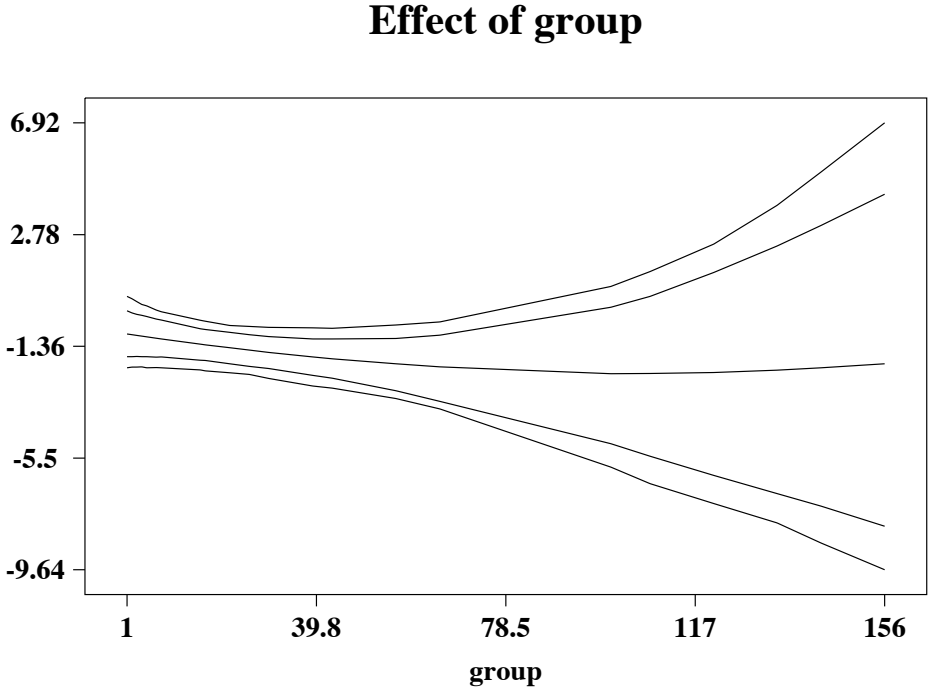


Figure 15.16: *Varying group effect $m_2(t)$ of model (6). Posterior median, 80%, and 95% intervals.*

$e^{1.304} = 3.68$. AG− increases the hazard of death by a factor of about 3.7. Recall that our estimate using the step function model (13.2.4) for h_0 was 3.5.

The BayesX script for fitting these models is available on the book website.

As promised, we discuss the identifiability issues associated with model (6). Model (5) is simpler and easier. In model (6) with $x = 0, 1$, the overall log-hazard function is

$$m(t,x) = \beta_0 + m_1(t) + xm_2(t).$$

Here $m(t,0)$ is the log-hazard function for the $AG-$ group and $m(t,1)$ is the log-hazard function for $AG+$ people. The function $m(t,x)$ is identifiable because it uniquely determines the distributions of the survival times for both the $AG-$ and $AG+$ groups. The function m_2 is also identifiable because it is a function of identifiable quantities, i.e.,

$$m_2(t) = m(t,1) - m(t,0).$$

However, β_0 and m_1 are not identifiable because we only know

$$\beta_0 + m_1(t) = m(t,0).$$

There are many choices for β_0 and $m_1(t)$ that will determine the same hazard function, and thus the same survival distribution for the $AG-$ group. Imposing the artificial side condition $\beta_0 \equiv \int m(t,0)\,dt$ uniquely determines β_0 and it follows that $m_1(t) = m(t,0) - \beta_0$ is both uniquely determined and integrates to 0.

Presumably, the goal of any estimation procedure for model (6) is to provide an estimate $\hat{m}(t,x)$. Obviously, this leads to $\hat{m}_2(t) \equiv \hat{m}(t,1) - \hat{m}(t,0)$. Estimates of β_0 and $m_1(t)$ depend on applying the artificial side condition to $\hat{m}(t,0)$. The thought process is a bit like that for linear models. With $Y = X\beta + e$, the first order of business is to estimate $\mathrm{E}(Y)$, which is the one thing that you can

obviously estimate (using Y if nothing else). Beyond that, parameters are identifiable or nonidentifiable depending on whether they are uniquely determined by $\mathrm{E}(Y)$, see Christensen (2002, Sec. 2.1). Naturally, any estimate of $\mathrm{E}(Y)$ provides unique estimates of identifiable functions.

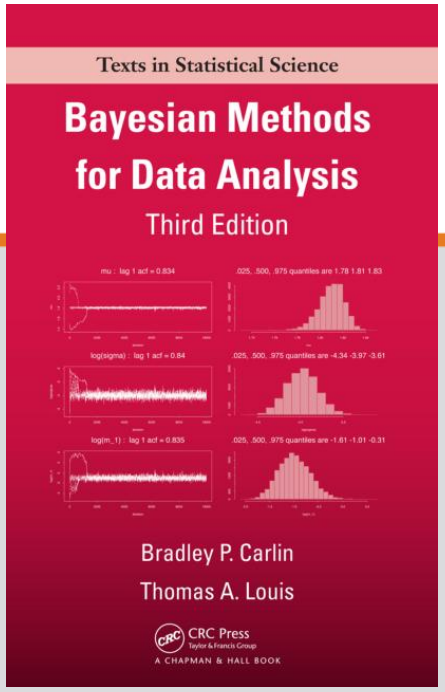
The methods in Sections 2 and 3 are closely related to functional data analysis, a subject to which Crainiceanu and Goldsmith (2009) provide a nice introduction.

This is the end of the last chapter. If you've completed this long, hard journey with us, all we can say is, "Good on ya, mate!" We hope you have learned as much from reading our book as we did writing it.

CHAPTER

6

MODEL CRITICISM AND SELECTION



This chapter is excerpted from

Bayesian Methods for Data Analysis, Third Edition

by Bradley P. Carlin, Thomas A. Louis.

Learn more

CHAPTER 4

Model criticism and selection

To this point we have seen the basic elements of Bayesian methods, arguments on behalf of their use from several different philosophical standpoints, and an assortment of computational algorithms for carrying out the analysis. We have observed that the generality of the methodology coupled with the power of modern computing enables consideration of a wide variety of hierarchical models for a given dataset. Given all this, the most natural questions for the reader to ask might be:

1. How can I tell if any of the assumptions I have made (e.g., the specific choice of prior distribution) is having an undue impact on my results?
2. How can I tell if my model is providing adequate fit to the data?
3. Which model (or models) should I ultimately choose for the final presentation of my results?

The first question concerns the *robustness* of the model, the second involves *assessment* of the model, and the third deals with *selection* of a model (or group of models). An enormous amount has been written on these three subjects over the last fifty or so years because they are the same issues faced by classical applied statisticians. The Bayesian literature on these subjects is of course smaller, but still surprisingly large, especially given that truly applied Bayesian work is a relatively recent phenomenon. We group the three areas together here since they all involve criticism of a model that has already been fit to the data.

Subsection 2.3.3 presented the fundamental tool of Bayesian model selection, the Bayes factor, while Section 2.5 outlined the basics of model assessment and model averaging. Armed with the computational techniques presented in Chapter 3, we now revisit and expand on these model building tools. With new MCMC-based model checking ideas arriving all the time (e.g., Dey et al., 1998; O'Hagan, 2003; Bayarri and Castellanos, 2007), we cannot hope to review all of what has been done. Still, in this chapter we will attempt to present the tools most useful for the applied Bayesian, along with sufficient exemplification so that the reader may employ the approaches independently.

4.1 Bayesian modeling

We begin this chapter with a discussion of several broad principles and strategies in Bayesian statistical modeling, intended to build and expand on the introductory hierarchical modeling material in Subsection 2.4. In particular, we wish to illustrate the basics of linear and nonlinear modeling for normal and binary data using `WinBUGS` in both the "flat" (nonhierarchical; no random effects) and hierarchical cases. Readers wishing to learn even more advanced modeling "tricks" (in both the statistical and `WinBUGS` senses of this word) may wish to consult the two books by Congdon (2003; 2007a) or the forthcoming book by Spiegelhalter et al. (2008).

4.1.1 Linear models

We begin then with the general linear model for normally distributed data, arguably the single most important contribution of statistics to the field of scientific inquiry. The bulk of statistical models appearing in print have this basic form, with regression and analysis of variance models being particularly widely used special cases. A Bayesian analysis of this model was first presented in the landmark paper by Lindley and Smith (1972), which we summarize here. Suppose that $\mathbf{Y}|\boldsymbol{\theta}_1 \sim N(A_1\boldsymbol{\theta}_1, C_1)$, where $\mathbf{Y}$ is an $n \times 1$ data vector, $\boldsymbol{\theta}_1$ is a $p_1 \times 1$ parameter vector, A_1 is an $n \times p_1$ known design matrix, and C_1 is an $n \times n$ known covariance matrix. Suppose further that we adopt the prior distribution $\boldsymbol{\theta}_1 \sim N(A_2\boldsymbol{\theta}_2, C_2)$, where $\boldsymbol{\theta}_2$ is a $p_2 \times 1$ parameter vector, A_2 is a $p_1 \times p_2$ design matrix, C_2 is a $p_1 \times p_1$ covariance matrix, and $\boldsymbol{\theta}_2, A_2$, and C_2 are all known. Then the *marginal* distribution of $\mathbf{Y}$ is

$$\mathbf{Y} \sim N(A_1A_2\boldsymbol{\theta}_2\,,\; C_1 + A_1C_2A_1')\,, \tag{4.1}$$

and the *posterior* distribution of $\boldsymbol{\theta}_1$ is

$$\boldsymbol{\theta}_1|\mathbf{y} \sim N(D\mathbf{d}\,,\; D)\,, \tag{4.2}$$

where

$$D^{-1} = A_1'C_1^{-1}A_1 + C_2^{-1}\,, \tag{4.3}$$

and

$$\mathbf{d} = A_1'C_1^{-1}\mathbf{y} + C_2^{-1}A_2\boldsymbol{\theta}_2\,. \tag{4.4}$$

Thus $E(\boldsymbol{\theta}_1|\mathbf{y}) = D\mathbf{d}$ provides a point estimate for $\boldsymbol{\theta}_1$, with associated variability captured by the posterior covariance matrix $Var(\boldsymbol{\theta}_1|\mathbf{y}) = D$.

Example 4.1 As a concrete illustration, we revisit the "linearized" version of the dugong (sea cow) growth data originally plotted in Figure 2.8 and analyzed in Example 2.10. There we employed a simple linear regression model,

$$Y_i = \beta_0 + \beta_1 \log(x_i) + \epsilon_i\,,$$

where Y_i is the length of the dugong in meters, x_i is the log of its age in

years, and the ϵ_i are i.i.d. normal with mean zero and precision $\tau = 1/\sigma^2$. This model can be cast in our general linear model framework by setting $\theta_1 = \boldsymbol{\beta} = (\beta_0, \beta_1)'$, $C_1 = \sigma^2 I_n$, and

$$A_1 = X = \begin{pmatrix} 1 & \log(x_1) \\ 1 & \log(x_2) \\ \vdots & \vdots \\ 1 & \log(x_n) \end{pmatrix} .$$

A noninformative prior is provided by taking $C_2^{-1} = \mathbf{0}$, i.e., setting the prior precision matrix equal to a $p_1 \times p_1$ matrix of zeroes. Then from equations (4.3) and (4.4), we have

$$D^{-1} = X'(\sigma^2 I_n)^{-1} X + \mathbf{0} = \frac{1}{\sigma^2}(X'X) , \tag{4.5}$$

and

$$\mathbf{d} = X'(\sigma^2 I_n)^{-1}\mathbf{y} + \mathbf{0} = \frac{1}{\sigma^2}(X'\mathbf{y}) , \tag{4.6}$$

so that the posterior mean is given by

$$D\mathbf{d} = \left[\frac{1}{\sigma^2}(X'X)\right]^{-1} \frac{1}{\sigma^2}(X'\mathbf{y}) = (X'X)^{-1}X'\mathbf{y} = \hat{\boldsymbol{\beta}}_{LS} ,$$

the usual least squares estimate of $\boldsymbol{\beta}$. From (4.2), the posterior distribution of $\boldsymbol{\beta}$ is

$$\boldsymbol{\beta}|\mathbf{y} \sim N(\hat{\boldsymbol{\beta}}_{LS},\, \sigma^2(X'X)^{-1}) . \tag{4.7}$$

Recall that the sampling distribution of the least squares estimate is given by $\hat{\boldsymbol{\beta}}_{LS}|\boldsymbol{\beta} \sim N(\boldsymbol{\beta},\, \sigma^2(X'X)^{-1})$, so that classical and noninformative Bayesian inferences regarding $\boldsymbol{\beta}$ will be formally identical in this example.

If we return to a $N_2(\boldsymbol{\mu}, R^{-1})$ prior for $\boldsymbol{\beta}$, but continue to (unrealistically) assume for the moment that $\tau, \boldsymbol{\mu}$, and R are all fixed and known, then equations (4.5) and (4.6) yield

$$D^{-1} = X'(\sigma^2 I_n)^{-1} X + R = \tau X'X + R ,$$

and

$$\mathbf{d} = X'(\sigma^2 I_n)^{-1}\mathbf{y} + R\boldsymbol{\mu} = \tau(X'\mathbf{y}) + R\boldsymbol{\mu} .$$

Taking $R = 0$ delivers the simple posterior distribution for $\boldsymbol{\beta}$ given in (4.7).

The closed form solutions we have just described are easily studied in the `R` language. For our dugong data, suppose we make our bivariate normal prior for $\boldsymbol{\beta}$ virtually "flat" by by setting $R = Diag(0.001, 0.001)$. We also set $\boldsymbol{\mu} = (0, 0)'$ and consider two values of τ, 1 and 100. The `R` code to set up the data and prior values is

R code
```
X <-cbind(rep(1,n),lgage)
mu <- c(0,0)
R <- matrix(c(0.001,0,0,0.001), nrow=2)
tau <- c(1,100)
```

where `Y, x` and `lgage` are as previously defined in Example 2.10. A simple function to determine the posterior of $\boldsymbol{\beta}$ is then

R code
```
postfn <- function(Y, X, mu, R, tau){
  p <- dim(X)[2]
  D <- solve( tau*t(X)%*%X + R)
  d <- tau*t(X)%*%Y + R%*%diag(rep(1,p))%*%mu
  postmean <- D%*%d
  postsd <- sqrt(diag(D))
  return(list(mean= D%*%d, sd=sqrt(diag(D)) ))
  }
```

Note this function makes liberal use of R's matrix multiply function, `%*%`. We now apply our new `postfn` function to the dugong data to obtain the posterior mean and variance of the slope β_1, as well as a 95% confidence interval. For $\tau = 1$ we have

R code
```
post1 <- postfn(Y, X, mu, R, tau[1]) # posterior for tau=1
beta <- post1$mean[2]
postsd <- post1$sd[2]
CI1 <- c(beta+qnorm(0.025)*postsd , beta+qnorm(0.975)*postsd)
```

Now typing `CI1` reveals the interval to be (–0.132, 0.687). Repeating the above calculation for $\tau = 100$

R code
```
post2 <- postfn(Y, X, mu, R, tau[2]) # posterior for tau=100
beta <- post1$mean[2]
postsd <- post1$sd[2]
CI2 <- c(beta+qnorm(0.025)*postsd , beta+qnorm(0.975)*postsd)
```

This instead delivers a CI of (0.236, 0.318). Note that this second interval is very similar to the frequentist interval we obtained in Example 2.10. The first interval is enormously wider because the far smaller τ value implies far lower faith in the data.

Finally, we can easily compute and plot the two posterior densities:

R code
```
theta <- seq(-0.2,0.6,length.out=100)
dens1<-dnorm(theta, mean=post1$mean[2], sd=post1$sd[2] )
dens2<-dnorm(theta, mean=post2$mean[2], sd=post2$sd[2] )
plot(theta, dens1, xlab=expression(beta), xlim=c(-0.2, 0.6),
  ylim=c(0,20), ylab="posterior density", type="n")
lines(theta, dens1, lty=1)        # posterior density for tau=1
lines(theta, dens2, lty=2)        # posterior density for tau=100
```

Setting $\tau = 1$ (i.e., high data precision) leads to the solid (`lty = 1`, line type #1) posterior in Figure 4.1. If we instead use $\tau = 100$, we obtain the dashed (`lty = 2`, line type #2) curve in the figure. Again, the increase in posterior precision arises from the tighter prior's willingness to lend more credence to the data.

Finally, we can also draw a sample from this posterior, and add this histogram to our plot along with a legend:

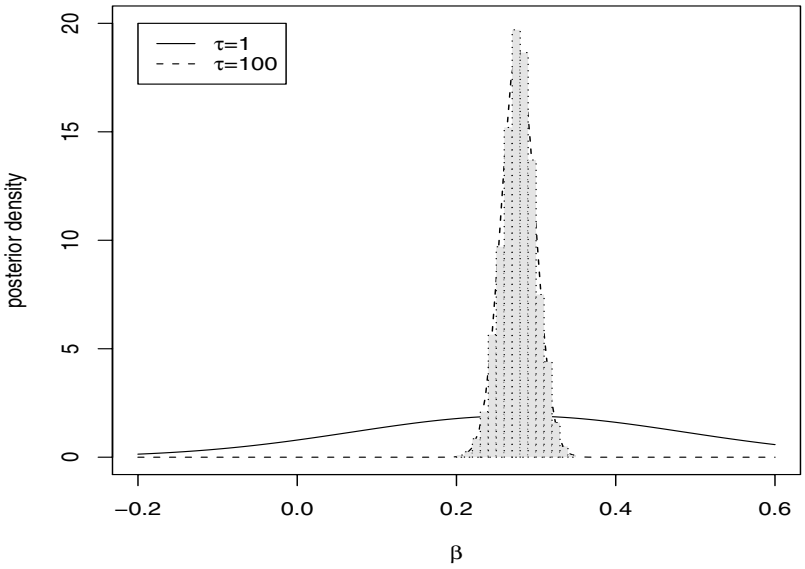


Figure 4.1 *Posterior for β_1, linearized dugong data, for two values of the data precision τ. A histogram of posterior samples is also shown in the $\tau = 100$ case.*

R code
```
postdraw <- rnorm(2000, mean=post2$mean[2], sd=post2$sd[2] )
r1<-hist(postdraw,freq=F,breaks=20, plot=F)
lines(r1,lty=3, freq=F, col="gray90")   #posterior draws, tau=100
legend(-0.2, 20, legend=c(expression(paste(tau,"=1", sep="")),
  expression(paste(tau,"=100", sep=""))), lty=c(1,2), ncol=1)
```

These values are shown as the dotted (`lty = 3`, line type #3) histogram in Figure 4.1. ■

If $C_1 = Var(\mathbf{Y}|\boldsymbol{\theta}_1)$ is unknown in a general linear model, or if the model has more than two levels (say, a third-stage hyperprior, in addition to the likelihood and the prior), then a closed form analytic solution for the posterior mean of $\boldsymbol{\beta}$ will typically be unavailable. Fortunately, sampling-based computational methods are available to escape this unpleasant situation, as we have already seen in Example 2.10 and elsewhere.

Example 4.2 In the previous example, we adopted a $N_2(\boldsymbol{\mu}, R^{-1})$ prior for $\boldsymbol{\beta}$, but assumed $\boldsymbol{\mu}$ and R were fixed and known. (Our flat priors on β_0 and β_1 are asymptotically equivalent to setting $R^{-1} = 0$.) Now suppose we wish to place third-stage hyperpriors on these parameters; specifically, a flat hyperprior on $\boldsymbol{\mu}$ and a $Wishart(\nu, \Omega)$ hyperprior on the prior precision

matrix R. Recall that we can make this latter hyperprior vague (but still proper) by setting $\nu = rank(R) = 2$.

Because we must use sampling-based methods in WinBUGS to handle this setting, we follow our original Example 2.10 dugong model and place a uniform prior on the data standard deviation σ as well. The WinBUGS code for this problem is

BUGS code
```
model{
  for( i in 1:n) {
    logage[i] <- log(x[i])
    Y[i] ~ dnorm(meen[i] , tau)
    meen[i] <- beta[1]+ beta[2]*(logage[i] - mean(logage[]))
    }

  beta[1:2] ~ dmnorm(mu[], R[ , ])
  mu[1] ~ dflat()
  mu[2] ~ dflat()
  R[1:2, 1:2] ~ dwish(Omega[ , ], 2)

  tau <- 1/(sigma*sigma)
  sigma ~ dunif(0.01, 100)
}
```

Note that now β_1 is the intercept and β_2 is the slope, since WinBUGS does not permit 0 subscripts. The full conditional for $\boldsymbol{\beta}$ remains exactly as in (4.7), since when developing the full conditional for a parameter we are allowed to condition on all other parameters in the model. Thus we would now *label* the full conditional as $p(\boldsymbol{\beta}|\mathbf{y}, \boldsymbol{\mu}, \sigma, R)$ instead of $p(\boldsymbol{\beta}|\mathbf{y})$, but its form is completely unchanged. The full conditional for $\boldsymbol{\mu}$ is also bivariate normal, though WinBUGS generates its two components separately as univariate normals, since the flat priors on the two $\boldsymbol{\mu}$ components are being specified separately in the code. The full conditional for R is Wishart by conditional conjugacy, while σ's nonconjugate specification over a bounded domain causes WinBUGS to select slice sampling as its updating method.

The Data and Inits files are natural extensions of the ones seen in Example 2.10:

BUGS code
```
# Data:
list(x = c( 1.0,  1.5,  1.5,  1.5, 2.5,   4.0,  5.0,  5.0,  7.0,
            8.0,  8.5,  9.0,  9.5, 9.5,  10.0, 12.0, 12.0, 13.0,
           13.0, 14.5, 15.5, 15.5, 16.5, 17.0, 22.5, 29.0, 31.5),
     Y = c(1.80, 1.85, 1.87, 1.77, 2.02, 2.27, 2.15, 2.26, 2.47,
           2.19, 2.26, 2.40, 2.39, 2.41, 2.50, 2.32, 2.32, 2.43,
           2.47, 2.56, 2.65, 2.47, 2.64, 2.56, 2.70, 2.72, 2.57),
     n = 27,
    Omega = structure(.Data = c(0.1, 0, 0, 0.1), .Dim = c(2, 2)))

# Inits:
```

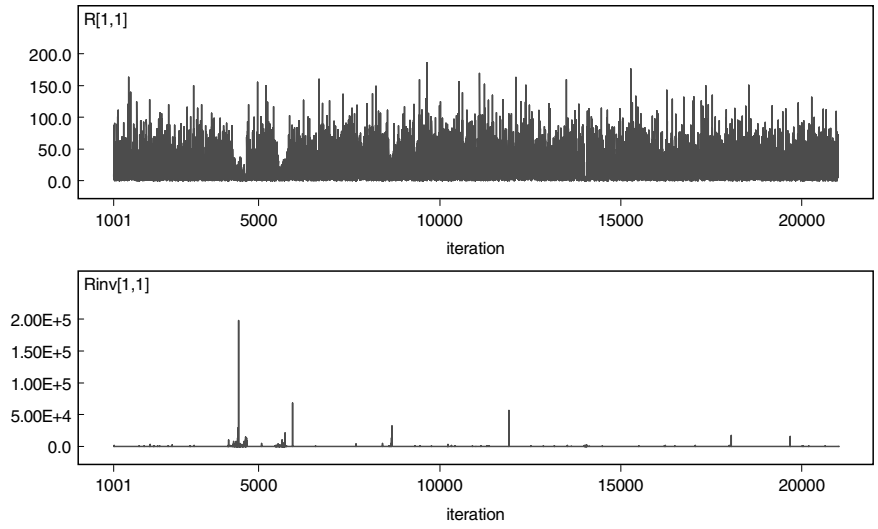


Figure 4.2 *Traceplots for* R_{11} *and* $(R^{-1})_{11}$*, linearized dugong model.*

```
list( beta = c(0, 0), sigma=1, mu = c(0,0),
       R = structure(.Data = c(1,0,0,1), .Dim = c(2, 2)))
```

The primary differences here are the initialization of $\boldsymbol{\beta}$ and $\boldsymbol{\mu}$ as vectors, the initialization of R, and the specification of the Ω matrix (a rough guess for R/ν). Here we simply set $\Omega = Diag(0.1, 0.1)$ because, while we expect the βs to be roughly uncorrelated thanks to our centering of the `logage` covariate, we have little idea as to their scale (at least not without "peeking" at the β posteriors obtained in previous examples). Running this code produces very similar DIC scores and posterior estimates for $\boldsymbol{\beta}$ and σ as obtained in Example 2.10, or even Example 4.1 when fixing $\tau = 100$.

The sampler converges well, but the hyperparameters remain hard to estimate. To see this, we monitor both R and R^{-1}. To do this we must first explicitly add the latter into our code using the intrinsic `inverse` function:

BUGS code

```
Rinv[1:2, 1:2] <- inverse(R[ , ])
priorsd[1] <- sqrt(Rinv[1,1])
priorsd[2] <- sqrt(Rinv[2,2])
```

Notice we have also defined a bivariate node `priorsd` to capture the prior standard deviations of the intercept and slope, respectively. Traceplots for R_{11} and $(R^{-1})_{11}$ based on 20,000 post-burn-in iterations are shown in Figure 4.2. While the former traces are reasonably stable and well-behaved, the occasional near-singularity of R causes a few extreme values to dominate the $(R^{-1})_{11}$ traces. This instability is inherited by `priorsd`: both of its component distributions are extremely heavy-tailed, with posterior

means that fluctuate over time but reasonably stable posterior medians (both around 0.49 after 60,000 iterations).

A related point is that these two estimates are not the same as the posterior standard deviations of the βs themselves (which are about 0.018 and 0.020, respectively – quite a bit smaller). These latter values reflect the bulk of information in the data, while the `priorsd` values essentially just reflect the minimal contribution of the Wishart hyperprior. This illustrates why a three-stage model is perhaps unnecessary here; there are no "replications" $\boldsymbol{\beta}_k$ from which to estimate the second-stage variability captured by R^{-1}. If however we had multiple *species* or *herds* of dugong indexed by k, then the third-stage of our current model would make perfect sense, and the posteriors of the `priorsd` parameters would now likely reflect any cross-species or cross-herd variability in the data, and not just whatever information we placed in the hyperprior. ■

4.1.2 Nonlinear models

Next we consider the case of nonlinear modeling, where we replace the linear mean structure $Y_i = \mathbf{x}_i'\boldsymbol{\beta} + \epsilon_i$ with a more generic form $Y_i = g(\mathbf{x}_i, \boldsymbol{\beta}) + \epsilon_i$ for some known function g. Here the Lindley and Smith (1972) result (4.2) typically will not apply to the full conditional distribution of the main effects $\boldsymbol{\beta}$, but may still be useful for higher stage parameters having linear model hyperpriors. In any case, posterior samples may still be generated using a mixture of Gibbs and Metropolis steps.

Example 4.3 Carlin and Gelfand (1991b) present a nonconjugate Bayesian analysis of the dugong data set, following the approach of Ratkowsky (1983). The idea is to model the untransformed data (closed circles in Figure 4.4 below) using the nonlinear growth model

$$Y_i = \alpha - \beta\gamma^{x_i} + \epsilon_i,\ i = 1, \ldots, n\ , \tag{4.8}$$

where $\alpha > 0$, $\beta > 0$, $0 \le \gamma \le 1$ and as usual $\epsilon_i \overset{iid}{\sim} N(0, \sigma^2)$ for $\sigma^2 > 0$. Thus α corresponds to the average length of a fully grown dugong, $(\alpha - \beta)$ is the length of a dugong at birth, and γ determines the growth rate. Specifically, lower values of γ produce an initially steep growth curve (rapid progression to adulthood immediately after birth), while higher γ values lead to much more gradual, almost linear growth.

The nonlinearity of the model eliminates any hope for a closed form full conditional for γ regardless of our choice of prior, so we proceed with a sampling-based solution. The `WinBUGS` code to fit this model (adapted from that given in the `Examples Vol II` section of the `Help` pull-down) is

BUGS code
```
model{
  for( i in 1 : N){
    Y[i] ~ dnorm(mu[i], tau)
```

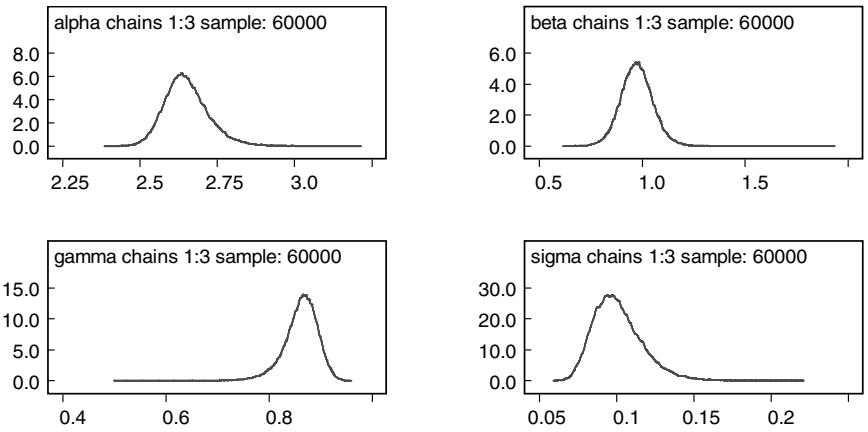


Figure 4.3 *Posterior density estimates, nonlinear dugong model.*

```
    mu[i] <- alpha - beta * pow(gamma,x[i])
    }
  alpha ~ dflat()
  beta ~ dflat()
  gamma ~ dunif(0.5, 1.0)

  tau <- 1/(sigma*sigma)
  sigma ~ dunif(0.01, 100)
}
```

Note flat priors are perfectly suitable for the two "endpoint" parameters α and β, but the harder-to-estimate growth parameter γ benefits from a tighter (albeit uniform) specification.

We run three parallel Gibbs sampling chains of 20,000 iterations each following a 1000-iteration burn-in, with the chains initialized as

BUGS code

```
# Inits:
  list(alpha = 1, beta = 1, sigma = 1, gamma = 0.9)
  list(alpha = 10, beta = 10, sigma = 10, gamma = 0.7)
  list(alpha = 100, beta = 100, sigma = 100, gamma = 0.5)
```

The resulting Gibbs sampler seems to mix well, as indicated by rapid agreement among the three chains' sample traces (not shown). Figure 4.3 shows the resulting density estimates for all four model parameters. Estimation appears good, with the fitted growth curve (obtained by plugging point estimates for α, β, and γ into equation (4.8)) closely following the observed data; again see Figure 4.4.

Autocorrelation plots (not shown) obtained via the `auto cor` button on the `Sample` monitor tool suggest rather high autocorrelations (i.e., rather different from 0 even at lag 20) only for α and γ. The bivariate scatterplot

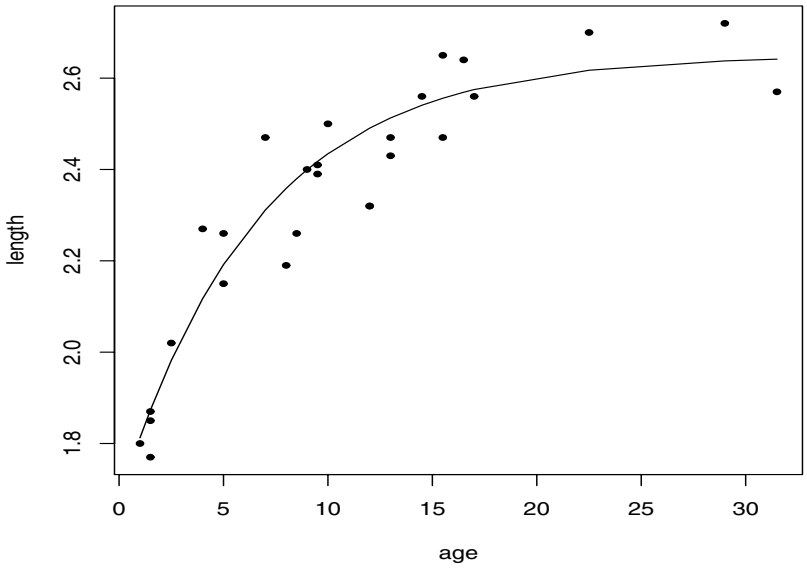


Figure 4.4 *Plot of the untransformed dugong data (length versus age) with fitted nonlinear growth curve superimposed.*

of the post-burn-in sampled values of these two parameters shown in Figure 4.5 reveals the reason for this slow convergence: these two parameters are highly correlated with a posterior that is roughly "boomerang-shaped." Larger values of α (the length of full grown dugongs) are generally associated with larger values of γ (i.e., a more gradual growth curve), but this relationship is complex and highly nonlinear. This complexity does not in and of itself cast doubt on the model, but is certainly typical of the problems we face when fitting nonlinear models: they often produce awkward joint posterior surfaces that will cause almost any MCMC algorithm to struggle. ■

4.1.3 Binary data models

Having handled the case of continuous response data, the next natural case to address is that of discrete response (binary or count) data. We do this in the context of the dugong data somewhat artificially by discretizing the response, as in the following example.

Example 4.4 Consider the following binary version of the dugong exam-

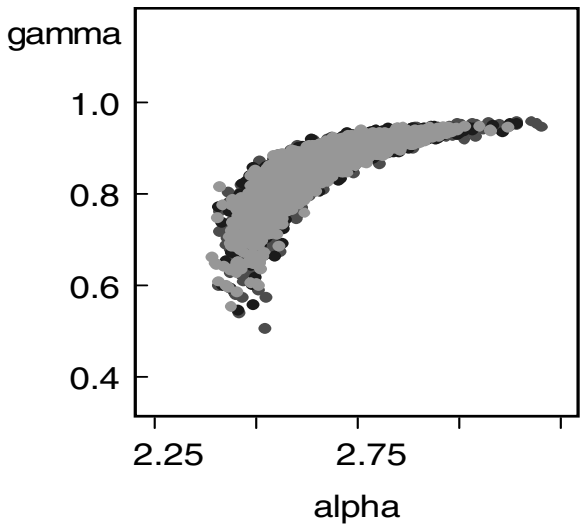


Figure 4.5 *Bivariate sample plot, α versus γ, nonlinear dugong model.*

ple, where we define the response

$$Z_i = \begin{cases} 1 & \text{if } Y_i > 2.4 \text{ (i.e., the dugong is full grown)} \\ 0 & \text{otherwise} \end{cases}$$

That is, $Z_i = 1$ if the dugong is "full grown," and 0 otherwise. We can then model $p_i = P(Z_i = 1)$ as

$$logit(p_i) = \log[p_i/(1-p_i)] = \beta_0 + \beta_1 log(age) \ ,$$

a classical, nonhierarchical logistic regression model. In practice, two other commonly used link functions are the *probit*

$$probit(p_i) = \Phi^{-1}(p_i) = \beta_0 + \beta_1 log(age) \ ,$$

and the *complementary log-log* (cloglog),

$$cloglog(p_i) = \log[-\log(1-p_i)] = \beta_0 + \beta_1 log(age) \ .$$

All of these models are easily handled in `WinBUGS` using the code:

BUGS code

```
model{
  for( i in 1:n) {
    logage[i] <- log(x[i])
    z[i] ~ dbern( p[i] )
    logit(p[i]) <- beta0 + beta1*(logage[i] - mean(logage[]))
#    p[i] <- phi(beta0 + beta1*(logage[i] - mean(logage[])))
#    cloglog(p[i]) <- beta0 + beta1*(logage[i] - mean(logage[]))
    }
  beta0 ~ dflat()
```

```
  beta1 ~ dflat()
}  #  end of WinBUGS code
```

Notice that the probit and cloglog models are commented out for the moment, using the pound sign (#). We also remark that this formulation of the probit (specifying p_i directly using the standard normal cdf `phi`) appears to be more numerically stable than using the `probit` function in `WinBUGS`.

The `Data` file is the obvious extension of the one used in previous dugong examples, namely

BUGS code
```
# Data:
list(x = c( 1.0,  1.5,  1.5,  1.5, 2.5,   4.0,  5.0,  5.0,  7.0,
            8.0,  8.5,  9.0,  9.5, 9.5,  10.0, 12.0, 12.0, 13.0,
           13.0, 14.5, 15.5, 15.5, 16.5, 17.0, 22.5, 29.0, 31.5),
     z = c(0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1,
           1, 1, 1, 1, 1, 1, 1, 1, 1),
     n = 27)
```

For our `Inits` values, we run three parallel sampling chains, overdispersed with respect to the true values of β_0 and β_1:

BUGS code
```
# Inits:
list( beta0 = -10, beta1 = -5)
list( beta0 =  0, beta1= 10)
list( beta0 = 10, beta1= 25)
```

We hasten to add that these starting values are too far from the true posterior to enable convergence in the probit case, and so for this model we instead use the "easier" values,

BUGS code
```
list(beta0 = -.8, beta1 = 3.3)
```

Using MCMC production runs of 20,000 post-burn in iterations each, Table 4.1 gives the model choice summaries, with fit captured by $\overline{D}$, effective size by p_D, and their sum providing the overall DIC score. Clearly the fit of all three models is very similar, with the cloglog enjoying a small advantage in $\overline{D}$ score. Since all three agree on p_D score (just under 2.0 "effective parameters"), this produces a slight win for the cloglog on DIC score. It would be a mistake to read too much into this "win," however, because the magnitude of the difference in $\overline{D}$ score is only slightly greater than its own Monte Carlo variability (crudely judged by running a few more MCMC iterations and recomputing the score).

Noting the fitted values (posterior means) of β_0 and β_1 for all three models, we may plot the data and fitted curves using the following `R` code:

R code
```
xgrid <- seq(0.5,32,length=101)
Y <- c(1.80, 1.85, 1.87, 1.77, 2.02, 2.27, 2.15, 2.26, 2.47,
       2.19, 2.26, 2.40, 2.39, 2.41, 2.50, 2.32, 2.32, 2.43,
       2.47, 2.56, 2.65, 2.47, 2.64, 2.56, 2.70, 2.72, 2.57)
```

Model	$\overline{D}$	p_D	DIC
logit	19.62	1.85	21.47
probit	19.30	1.87	21.17
cloglog	18.77	1.84	20.61

Table 4.1 *Model selection table, binarized dugong example.*

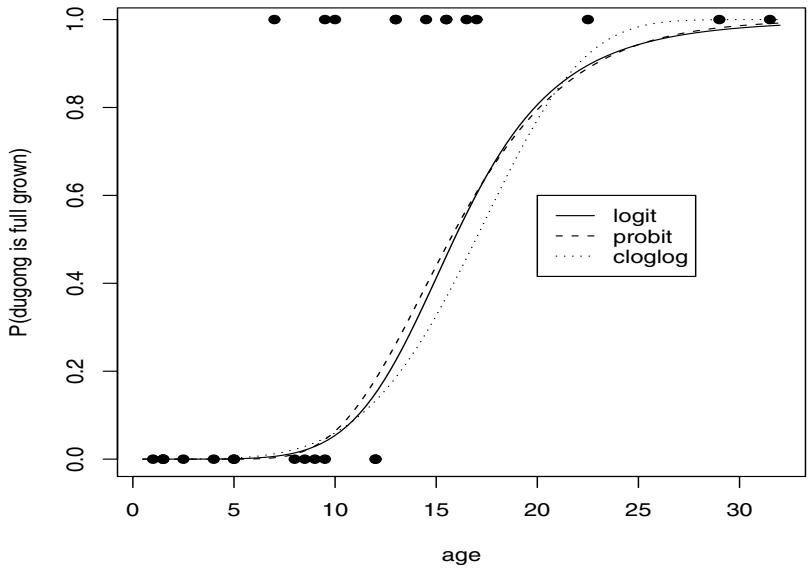


Figure 4.6 *Plot of the fitted logit, probit, and complementary log-log (cloglog) models to the binarized dugong data (shown as filled circles).*

```
lgage <- log(xgrid)
ctlgage <- lgage - mean(lgage)
z <- as.integer(Y>2.4)

beta0 <- -1.52;  beta1 <- 6.19
p_logit <- exp(beta0 + beta1*ctlgage)/(1 + exp(beta0 + beta1*ctlgage))

beta0 <- -0.79;  beta1 <- 3.39
p_probit <- pnorm(beta0 + beta1*ctlgage)

beta0 <- -1.79;  beta1 <- 4.58
```

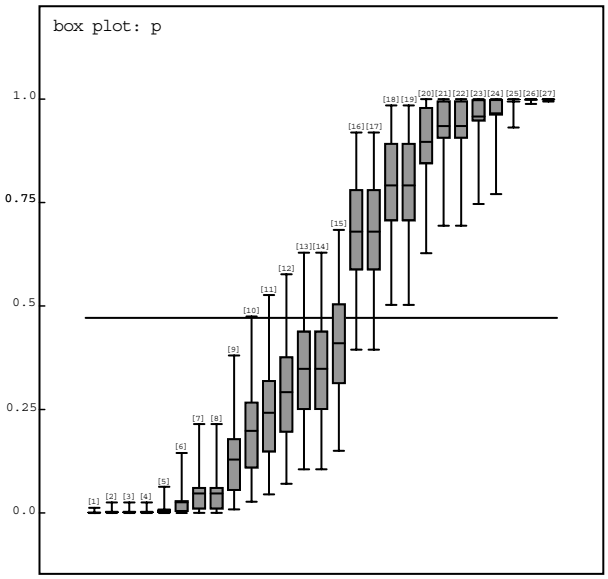


Figure 4.7 *Boxplots of p_i posterior distributions, binarized dugong data with complementary log-log link function.*

```
p_cloglog <- 1 - exp(-exp(beta0 + beta1*ctlgage))

plot(xgrid, p_logit, xlab="age", ylab="P(dugong is full grown)",
  pch=20,type="l")
lines(xgrid, p_probit, lty=2)
lines(xgrid, p_cloglog, lty=3)
points(age, z, pch=19)
legend(20, .6, legend=c("logit", "probit", "cloglog"), lty=1:3,
  ncol=1)
```

The resulting plot is shown in Figure 4.6. The logit and probit fits appear very similar, but the cloglog fitted curve is slightly different, increasing more slowly at first, but then accelerating and reaching its upper asymptote of 1 a bit sooner. Finally, boxplots of the posterior distributions of each of the p_i, induced by the link function and the posteriors for β_0 and β_1, are easily obtained using the `box plot` button on the comparison tool (`Inference` menu) in `WinBUGS`. Figure 4.7 shows the results under our slightly preferred cloglog link function. ■

We hope the collection of examples in this section (and in particular, the model choice example just completed) gives some idea about Bayesian modeling in practice, and sets up our more expansive discussions of model adequacy and comparison in the remainder of this chapter.

4.2 Bayesian robustness

As we have already mentioned, a commonly voiced concern with Bayesian methods is their dependence on various aspects of the modeling process. Possible sources of uncertainty include the prior distribution, the precise form of the likelihood, and the number of levels in the hierarchical model. Of course, these are concerns for the frequentist as well. For example, the statistics literature is filled with methods for investigating the effect of case deletion, which is a special type of likelihood sensitivity. Still, the problem appears more acute for the Bayesian due to the appearance of the prior distribution (even though it may have been deliberately chosen to be noninformative).

In the next subsection, we investigate the robustness of the conclusions of a Bayesian analysis by checking whether or not they remain essentially unchanged in the presence of perturbations in the prior, likelihood, or some other aspect of the model. Subsection 4.2.2 considers a reverse attack on the problem, wherein we attempt to characterize the class of model characteristics (say, prior distributions) that lead to a certain conclusion given the observed data. Both of these approaches differ slightly from what many authors call the "robust Bayesian viewpoint," which seeks model settings (particularly prior distributions) that are robust from the outset, rather than picking a (perhaps informative) baseline and checking it later, or cataloging assumptions based on a predetermined decision. The case studies in Chapter 8 feature a variety of robustness investigations, while further methodological tools are provided in Chapter 7.

4.2.1 Sensitivity analysis

The most basic tool for investigating model uncertainty is the *sensitivity analysis*. That is, we simply make reasonable modifications to the assumption in question, recompute the posterior quantities of interest, and see whether they have changed in a way that has practical impact on interpretations or decisions. If not, the data are strongly informative with respect to this assumption and we need not worry about it further. If so, then our results are sensitive to this assumption, and we may wish to communicate the sensitivity, think more carefully about it, collect more data, or all of these.

Example 4.5 Suppose our likelihood features a mean parameter θ and a variance parameter σ^2, and in our initial analysis we employed a $N(\mu, \tau^2)$

prior on θ and an $IG(a, b)$ prior on σ^2. To investigate possible sensitivity of our results to the prior for θ, we might first consider shifts in its mean, e.g., by increasing and decreasing μ by one prior standard deviation τ. We also might alter its precision, say, by doubling and halving τ. Alternatively, we could switch to a heavier-tailed t prior with the same moments.

Sensitivity to the prior for σ^2 could be investigated in the same way, although we would now need to solve for the hyperparameter values required to increase and decrease its mean or double and halve its standard deviation. ■

The increased ease of computing posterior summaries via Monte Carlo methods has led to a corresponding increased ease in performing sensitivity analyses. For example, given a converged MCMC sampler for a posterior distribution $p(\theta|\mathbf{y})$, it should take little time for the sampler to adjust to the new posterior distribution $p_{NEW}(\theta|\mathbf{y})$ arising from some modest change to the prior or likelihood. Thus, the additional sampling could be achieved with a substantial reduction in the burn-in period. Still, for a model having p hyperparameters, simply considering two alternatives (one larger, one smaller) to the baseline value for each results in a total of 3^p possible priors – possibly infeasible even using MCMC methods.

Fortunately, a little algebraic work eliminates the need for further sampling. Suppose we have a sample $\{\theta_1, \ldots, \theta_N\}$ from a posterior $p(\theta|\mathbf{y})$, which arises from a likelihood $f(\mathbf{y}|\theta) = \prod_{i=1}^n f(y_i|\theta)$ and a prior $p(\theta)$. To study the impact of deleting case k, we see that the new posterior is

$$p_{NEW}(\theta|\mathbf{y}) \propto \frac{f(\mathbf{y}|\theta)\pi(\theta)}{f(y_k|\theta)} \propto \frac{1}{f(y_k|\theta)} p(\theta|\mathbf{y}) \,.$$

In the notation of Subsection 3.3.2, we can use $g(\theta) = p(\theta|\mathbf{y})$ as an importance sampling density, so that the weight function is given by

$$w(\theta) = \frac{p_{NEW}(\theta|\mathbf{y})}{p(\theta|\mathbf{y})} = \frac{1}{f(y_k|\theta)} \,.$$

Thus for any posterior function of interest $h(\theta)$, we have $\hat{E}[h(\theta)|\mathbf{y}] = \sum_{j=1}^N h(\theta_j)/N$, and

$$\hat{E}_{NEW}[h(\theta)|\mathbf{y}] = \frac{\sum_{j=1}^N h(\theta_j) w(\theta_j)}{\sum_{j=1}^N w(\theta_j)} \,, \tag{4.9}$$

using equation (3.4). Because p and p_{NEW} should be reasonably similar, the former should be a good importance sampling density for the latter and hence the approximation in equation (4.9) should be good for moderate N. We could also obtain an approximate sample from p_{NEW} via the weighted bootstrap by resampling θ_i values with probability q_i, where $q_i = w(\theta_i)/\sum_{j=1}^N w(\theta_j)$.

In situations where large changes in the prior are investigated, or where outlying or influential cases are deleted, p_{NEW} may differ substantially from p. A substantial difference between the two posteriors can lead to highly variable importance sampling weights and, hence, poor estimation. Importance link function transformations of the Monte Carlo sample (MacEachern and Peruggia, 2000b) can be used to effectively alter the distribution from which the $\{\theta_1, \ldots, \theta_N\}$ were drawn, stablizing the importance sampling weights and improving estimation.

Sensitivity analysis via asymptotic approximation

Notice that importance sampling may be used for sensitivity analysis even if the original posterior sample $\{\theta_1, \ldots, \theta_N\}$ was obtained using some other method (e.g., the Gibbs sampler). The new posterior estimates in (4.9) are not likely to be as accurate as the original, but are probably sufficient for the purpose of a sensitivity analysis. In the same vein, the asymptotic methods of Section 3.2 can be used to obtain approximate sensitivity results without any further sampling *or* summation. For example, we have already seen in equation (3.1) a formula requiring only two maximizations (both with respect to the original posterior) that enables any number of sensitivity investigations. In the realm of model comparison, Kass and Vaidyanathan (1992) use such approximations to advantage in determining the sensitivity of Bayes factors to prior and likelihood input. Let $\pi_{NEW,k}(\theta_k)$ be new priors on θ_k where $k = 1, 2$ indexes the model. Writing BF for the Bayes factor under the original priors and BF_{NEW} for the Bayes factor under the new priors, Kass and Vaidyanathan (1992, equation (2.23)) show that

$$BF_{NEW} = BF \cdot \frac{r_1(\tilde{\theta}_1)}{r_2(\tilde{\theta}_2)} \cdot \left\{1 + O\left(\frac{1}{n}\right)\right\} \tag{4.10}$$

where $r_k(\tilde{\theta}_k) = \pi_{NEW,k}(\tilde{\theta}_k)/\pi_k(\tilde{\theta}_k)$ and $\tilde{\theta}_k$ is the posterior mode using the original prior $\pi_k(\theta_k)$. Thus, one only needs to evaluate r_k at $\tilde{\theta}_k$ for $k = 1, 2$ for each new pair of priors, and these computations are sufficiently easy and rapid that a very large number of new priors can be examined without much difficulty.

A further simplification comes from a decomposition of the parameter vector into two components, $\theta_k = (\theta_k^{(1)}, \theta_k^{(2)})$ with the first component containing all parameters that are common to both models. We define $\theta^{(1)} = \theta_1^{(1)}$ $(= \theta_2^{(1)})$. This parameter vector might be considered a nuisance parameter while the $\theta_k^{(2)}$, $k = 1, 2$ become the parameter vectors of interest. If we take the priors to be such that the two components are *a priori* independent under both models, we may write $\pi_k(\theta_k) = \pi_k^{(1)}(\theta^{(1)}) \cdot \pi_k^{(2)}(\theta_k^{(2)})$. Suppose this is the case, and furthermore $\pi_1^{(1)}(\theta^{(1)}) = \pi_2^{(1)}(\theta^{(1)})$, and that both these conditions hold for the new priors considered. In this situation, if

it turns out that the first-component posterior modes under the two models are approximately equal (formally, to order $O(n^{-1})$), then

$$\frac{\pi_{NEW,1}(\tilde{\theta}_1)}{\pi_{NEW,2}(\tilde{\theta}_2)} \doteq \frac{\pi^{(2)}_{NEW,1}(\tilde{\theta}^{(2)}_1)}{\pi^{(2)}_{NEW,2}(\tilde{\theta}^{(2)}_2)}$$

to the same order of accuracy as equation (4.10). The ratio $r_1(\tilde{\theta}_1)/r_2(\tilde{\theta}_2)$ in (4.10) thus involves the new priors only through their second components. In other words, if the modal components $\tilde{\theta}^{(1)}_k$ are nearly equal for $k = 1$ and 2, when we perform the sensitivity analysis we do not have to worry about the effect of modest modifications to the prior on the nuisance-parameter component of θ_k. Carlin, Kass, Lerch, and Huguenard (1992) use this approach for investigating sensitivity of a comparison of two competing models of human working memory load to outliers and changes in the prior distribution.

Sensitivity analysis via scale mixtures of normals

The conditioning feature of MCMC computational methods enables another approach to investigating the sensitivity of distributional specifications in either the likelihood or prior for a broad class of common hierarchical models. Consider the model $y_i = \mu_i + \epsilon_i,\ i = 1, \ldots, n$, where the μ_i are unknown mean structures and the ϵ_i are independent random errors having density f with mean 0. For convenience, one often assumes that the ϵ_i form a series of independent normal errors, i.e., $\epsilon_i|\sigma^2 \sim N(0, \sigma^2)$, $i = 1, \ldots, n$. However, the normal distribution's light tails may well make this an unrealistic assumption, and so we wish to investigate alternative forms that allow greater variability in the observed y_i values. Andrews and Mallows (1974) show that expanding our model to

$$\epsilon_i|\sigma^2, \lambda_i \sim N(0, \lambda_i\sigma^2),\ i = 1, \ldots, n\ ,$$

and subsequently placing a prior on λ_i enables a variety of familiar (and more widely dispersed) error densities to emerge. That is, we create f as the *scale mixture of normal distributions*,

$$f(\epsilon_i|\sigma^2) = \int_\Lambda p(\epsilon_i|\sigma^2, \lambda_i)p(\lambda_i)d\lambda_i\ ,\ i = 1, \ldots, n\ .$$

The following list identifies the necessary functional forms for $p(\lambda_i)$ to obtain some of the possible departures from normality:

- Student's t errors: If $\nu/\lambda_i \sim \chi^2_\nu$ (i.e., if $\lambda_i \sim IG(\frac{\nu}{2}, \frac{2}{\nu})$), then $\epsilon_i|\sigma \sim t_\nu(0, \sigma)$.
- Double exponential errors: If $\lambda_i \sim Expo(2)$, the exponential distribution having mean 2, then $\epsilon_i|\sigma \sim DE(0, \sigma)$, where DE denotes the double exponential distribution.

- Logistic errors: If $1/\sqrt{\lambda_i}$ has the asymptotic Kolmogorov distance distribution, then $\epsilon_i|\sigma$ is logistic (see Andrews and Mallows, 1974).

Generally one would not view the addition of n parameters to the model as a simplifying device, but would prefer instead to work directly with the nonnormal error density in question. However, Carlin and Polson (1991) point out that the conditioning feature of the Gibbs sampler makes this augmentation of the parameter space quite natural. To see this, note first that under an independence prior on the λ_i, we have $\lambda_i|\{\lambda_{j\neq i}\}, \{\mu_j\}, \sigma^2, \mathbf{y} \sim \lambda_i|\mu_i, \sigma^2, y_i$. But by Bayes theorem, $p(\lambda_i|\mu_i, \sigma^2, y_i) \propto p(y_i|\mu_i, \sigma^2, \lambda_i)p(\lambda_i)$, where the appropriate normalization constant, $p(y_i|\mu_i, \sigma^2)$, is known by construction as the desired nonnormal error density. Hence the complete conditional for λ_i will always be of known functional form. Generation of the required samples may be done directly if this form is a standard density; otherwise, a carefully selected rejection method may be employed. Finally, since the remaining complete conditionals (for σ^2 and the μ_j) are determined given $\boldsymbol{\lambda}$, convenient prior specifications may often be employed with the normal likelihood, again leading to direct sampling.

Example 4.6 Consider the model $y_i = \mu_i + \epsilon_{ij}$, where j indexes the model error distribution (i.e., the mixing distribution for the λ_i), and i indexes the observation, as before. Suppose that $\mu_i = f(\mathbf{x}_i, \theta)\boldsymbol{\beta}$, where $\boldsymbol{\beta}$ is a k-dimensional vector of linear nuisance parameters, and $f(\mathbf{x}_i, \theta) = (f_1(\mathbf{x}_i, \theta), \ldots, f_k(\mathbf{x}_i, \theta))$ is a collection of known functions, possibly nonlinear in θ. Creating the $n \times k$ matrix $F_\theta = (f(\mathbf{x}_i, \theta))$, the log-likelihood $\log p(\mathbf{y} \mid \theta, \boldsymbol{\beta}, \boldsymbol{\lambda}, \sigma^2)$ is

$$-\frac{1}{2\sigma^2}(\mathbf{y} - F_\theta\boldsymbol{\beta})^T\Sigma_i^{-1}(\mathbf{y} - F_\theta\boldsymbol{\beta}) - n\log\sigma - \frac{1}{2}\sum_{i=1}^{n}\log\lambda_i \ ,$$

where $\Sigma_i = Diag(\lambda_1, \ldots, \lambda_n)$. Assume that $\boldsymbol{\beta} \sim N(\boldsymbol{\beta}_0, \Sigma_0)$, where $\boldsymbol{\beta}_0$ and Σ_0 are known. In addition, let $\sigma^2 \sim IG(a_0, b_0)$, and let θ have prior distribution $p(\theta)$.

Suppose that, due to uncertainty about the error density and the impact of possible outliers, we wish to compare the three models

$$M = 1: \quad \epsilon_i \sim N(0, \sigma^2) \ ,$$

$$M = 2: \quad \epsilon_i \sim t(0, \sigma^2, \nu = 2) \text{ , and}$$

$$M = 3: \quad \epsilon_i \sim DE(0, \sigma) \ ,$$

where M indicates which error distribution (model) we have selected. For $M = 1$, clearly the λ_i are not needed, so their full conditional distributions are degenerate at 1. For $M = 2$, we have the full conditional distribution

$$IG\left(\frac{\nu+1}{2}, \left\{\frac{1}{2}\left[\frac{(y_i - f(\mathbf{x}_i, \theta)\boldsymbol{\beta})^2}{\sigma^2} + \nu\right]\right\}^{-1}\right), \ i = 1, \ldots, n \ ,$$

in a manner very similar to that of Andrews and Mallows (1974). Finally, for $M = 3$ we have a complete conditional proportional to

$$\lambda_i{}^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\left(\lambda_i + \frac{(y_i - f(\mathbf{x}_i, \theta)\boldsymbol{\beta})^2}{\lambda_i \sigma^2}\right)\right) ,$$

that is, $\lambda_i|\theta, \boldsymbol{\beta}, \sigma^2, \mathbf{y} \sim GIG\left(\frac{1}{2}, 1, (y_i - f(\mathbf{x}_i, \theta)\boldsymbol{\beta})^2/\sigma^2\right)$ for $i = 1, \ldots, n$, where GIG denotes the generalized inverse Gaussian distribution (see Devroye, 1986, p. 478). In order to sample from this density, we note that it is the reciprocal of an

$$\text{Inverse Gaussian}\left(\left|\frac{\sigma}{y_i - f(\mathbf{x}_i, \theta)\boldsymbol{\beta}}\right|, 1\right) ,$$

a density from which we may easily sample (see e.g. Devroye, 1986, p. 149). ■

Thus, the scale mixture of normals approach enables investigation of nonnormal error distributions as a component of the original model, rather than later as part of a sensitivity analysis, and with little additional computational complexity. If the λ_i all have posterior distributions centered tightly about 1, the additional modeling flexibility is unnecessary, and the assumption of normality is appropriate. In this same vein, the λ_i can be thought of as outlier diagnostics, because extreme observations will correspond to extreme fitted values of these scale parameters. Wakefield et al. (1994) use this idea to detect outlying individuals in a longitudinal study by employing the normal scale mixture at the *second* stage of their hierarchical model (i.e., on the prior for the random effect component of μ_i).

Example 4.7 Using the normal scale mixture idea, consider a reanalysis of Fisher's sleep data from Chapter 2, Problem 20. These are the increased hours of sleep for 10 patients treated with soporific B compared with soporific A, given as

$$1.2, 2.4, 1.3, 1.3, 0.0, 1.0, 1.8, 0.8, 4.6, 1.4 .$$

Suppose we use `WinBUGS` to compare models having mean θ and the three error distributions considered in Example 4.6, namely the normal, the t_2, and the DE. These are readily available as normal scale mixtures using mixing parameters λ_i as follows:

$M = 1:$ $\lambda_i = 1$

$M = 2:$ $\lambda_i \sim IG(1, 1)$

$M = 3:$ $\lambda_i \sim Expo(2)$

These three choices are easily coded in `WinBUGS`. In fact, in the code that follows we use a "data duplication" trick that enables us to fit all three models *simultaneously*, facilitating model comparison:

BUGS code

```
model
{
  for(i in 1:N) {
    # Duplicate the data
    Y1[i] <- Y[i]
    Y2[i] <- Y[i]
    Y3[i] <- Y[i]

    # Weighted precision parameters
    tau1[i] <- tau0[1]/lambda1[i];
    tau2[i] <- tau0[2]/lambda2[i];
    tau3[i] <- tau0[3]/lambda3[i];

    # Mean structures (all same)
    Y1[i] ~ dnorm(theta[1],tau1[i]);
    Y2[i] ~ dnorm(theta[2],tau2[i]);
    Y3[i] ~ dnorm(theta[3],tau3[i]);

    # Error distributions
    lambda1[i] <- 1;                # M1 = e_i ~ N(0,tau1)
    lambda2[i] <- 1/inv.lambda2[i]; # So that lambda ~ IG
    inv.lambda2[i] ~ dgamma(1,1);   # M2 = e_i ~ t_2(0, tau2)
   lambda3[i] ~ dexp(0.5);         # M3 = e_i ~ DE(0,tau3)
  }
# Priors
  for (k in 1:3) {
    theta[k] ~ dnorm(0,0.00001)     # Vague normal on mean
    sigma[k] ~ dunif(0.01,100)      # Uniform on sigma
    tau0[k] <- 1/(sigma[k]*sigma[k])
  }
}
```

Here, $k = 1, \ldots, 3$ indexes the model while $i = 1, \ldots, 10$ indexes the observation. `WinBUGS` will run all three models simultaneously, but their posteriors will be independent since there is no connection across k anywhere in the code. The `Compare, Stats,` and `DIC` tools may now produce convenient displays featuring all 3 models (see below).

Potential outliers can be readily identified by examining the boxplots of the λ_i posterior distributions for Models 2 and 3 in Figure 4.8. Subject 9 (who got 4.6 additional hours of sleep) is easily identified by the posterior of λ_9, which is centered near 10 and has 95th percentile well over 100 for Model 2 (note a log scale is being used on the vertical axis to adjust the plot for the extremely heavy upper tail). The large λ_9 provides the variance inflation needed to accommodate this outlying data point. Subjects 2 (2.4 additional hours) and 5 (0.0 additional hours) are also potential outliers, but to a far lesser degree.

To understand the benefit of nonnormal errors a bit better, consider the

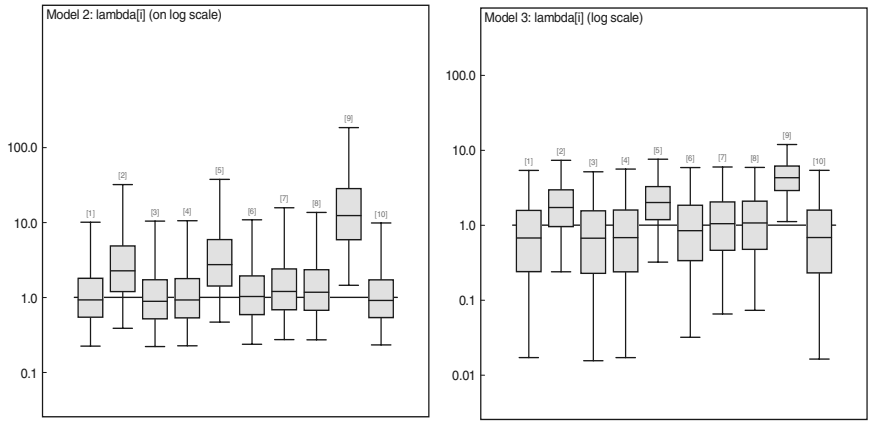


Figure 4.8 *Comparison of outlier diagnostics (λ_i posterior boxplots) for the two nonnormal error distributions, Fisher's sleep data.*

Model	mean	SD	2.5%	median	97.5%
1 (normal errors)	1.57	0.48	0.62	1.58	2.51
2 (t_2 errors)	1.31	0.28	0.78	1.29	1.91
3 (DE errors)	1.32	0.27	0.80	1.30	1.91

Table 4.2 *Posterior comparison for the grand mean θ across the three error distributions, Fisher's sleep data.*

fixed effect estimates in Table 4.2, as well as the boxplots of the three posterior distributions for the grand mean θ in Figure 4.9. Notice that the mean is shifted down in Models 2 and 3 (away from the large outlier and back toward the bulk of the data) relative to Model 1. In addition, θ is more precisely estimated in the t_2 and DE errors cases, as indicated by the narrower 95% central confidence intervals in Table 4.2 and the narrower boxplots in Figure 4.9. This is because the heavier tails of the t_2 and DE distributions allow them to accommodate the large outlier more readily, resulting in posterior estimates for the mean sleep increase that are both more accurate and more precise. ■

4.2.2 Prior partitioning

The method of sensitivity analysis described in the previous subsection is a direct and conceptually simple way of measuring the effect of the prior distributions and other assumptions made in a Bayes or empirical Bayes

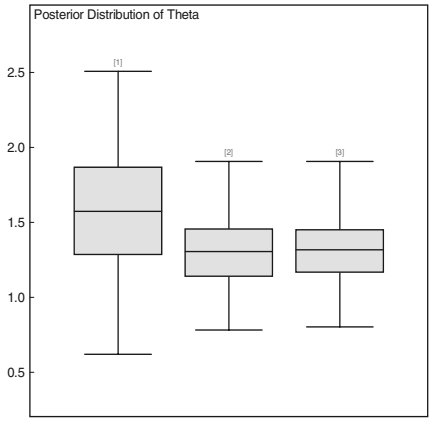


Figure 4.9 *Comparison of posterior boxplots for θ for three error distributions, Fisher's sleep data.*

analysis. However, it does not free us from careful development of the original prior, which must still be regarded as a reasonable baseline. This can be impractical because the prior beliefs and vested interests of the potential consumers of our analysis may be unimaginably broad. For example, if we are analyzing the outcome of a government-sponsored clinical trial to determine the effectiveness of a new drug relative to a standard treatment for a given disease, our results will likely be read by doctors in clinical practice, epidemiologists, regulatory workers (e.g., from the U.S. Food and Drug Administration), legislators, and, of course, patients suffering from or at risk of contracting the disease itself. These groups are likely to have widely divergent opinions as to what constitutes "reasonable" prior opinion; the clinician who developed the drug is likely to be optimistic about its value, while the regulatory worker (who has seen many similar drugs emerge as ineffective) may be more skeptical. What we need is a method for communicating the robustness of our conclusions to *any* prior input the reader deems appropriate.

A potential solution to this problem is to "work the problem backward," as follows. Suppose that, rather than fix the prior and compute the posterior distribution, we fix the posterior (or set of posteriors) that produce a given conclusion, and determine which prior inputs are consistent with this desired result, given the observed data. The reader would then be free to determine whether the outcome was reasonable according to whether the prior class that produced it was consistent with his or her own prior beliefs. We refer to this approach simply as *prior partitioning* since we are subdividing the prior class based on possible outcomes, though it is important to remember that such partitions also depend on the data and the decision to be reached. As such, the approach is not strictly Bayesian (the

data are playing a role in determining the prior), but it does provide valuable robustness information while retaining the framework's philosophical, structural, documentary, and communicational advantages.

To illustrate the basic idea, consider the point null testing scenario $H_0 : \theta = \theta_0$ versus $H_1 : \theta \neq \theta_0$. Without loss of generality, set $\theta_0 = 0$. Suppose our data x has density $f(x|\theta)$, where θ is an unknown scalar parameter. Let π represent the prior probability of H_0, and $G(\theta)$ the prior cumulative distribution function (cdf) of θ conditional on $\{\theta \neq 0\}$. The complete prior cdf for θ is then $F(\theta) = \pi I_{[0,\infty)}(\theta) + (1-\pi)G(\theta)$, where I_S is the indicator function of the set S. Hence, the posterior probability of the null hypothesis is

$$P_G(\theta = 0|x) = \frac{\pi f(x|0)}{\pi f(x|0) + (1-\pi)\int f(x|\theta)dG(\theta)} \; . \tag{4.11}$$

Prior partitioning seeks to characterize the G for which this probability is less than or equal to some small probability $p \in (0,1)$, in which case we reject the null hypothesis. (Similarly, we could also seek the G leading to $P_G(\theta \neq 0|x) \leq p$, in which we would reject H_1.) Elementary calculations show that characterizing this class of priors $\{G\}$ is equivalent to characterizing the set $\mathcal{H}_c$, defined as

$$\mathcal{H}_c = \left\{ G : \int f(x|\theta)dG(\theta) \geq c = \frac{1-p}{p}\frac{\pi}{1-\pi}f(x|0) \right\} \; . \tag{4.12}$$

Carlin and Louis (1996) establish results regarding the features of $\mathcal{H}_c$, and then use these results to obtain sufficient conditions for $\mathcal{H}_c$ to be nonempty for classes of priors that satisfy various moment and percentile restrictions. The latter are somewhat more useful, since percentiles and tail areas of the conditional prior G are transform-equivariant, and Chaloner et al. (1993) have found that elicitees are most comfortable describing their opinions through a "best guess" (mean, median, or mode) and a few relatively extreme percentiles (say, the 5^{th} and the 95^{th}).

To lay out the technical detail of this percentile restriction approach, let θ_L and θ_U be such that $P_G(\theta \leq \theta_L) = a_L$ and $P_G(\theta > \theta_U) = a_U$, where a_L and a_U lie in the unit interval and sum to less than 1. For fixed values for θ_L and θ_U, we seek the region of (a_L, a_U) values (i.e., the class of priors) that lead to a given decision. To accomplish this we can take a general G, alter it to pass through (θ_L, a_L) and $(\theta_U, 1-a_U)$, and then search over all G subject only to the constraint that the intervals $(-\infty, a_L]$, $(a_L, a_U]$, and (a_U, ∞) all have positive probability.

Assume that $f(x|\theta)$ is a unimodal function of θ for fixed x that vanishes in both tails, an assumption that will be at least approximately true for large datasets due to the asymptotic normality of the observed likelihood function. Keeping θ_L, θ_U, and $\mathbf{a}$ fixed, we seek the supremum and infimum

of $\int f(x|\theta)dG(\theta)$. The infimum will always be given by

$$(1 - a_L - a_U) \min\{f(x|\theta_L), f(x|\theta_U)\}\ , \tag{4.13}$$

since a unimodal f must take its minimum over the central support interval at one of its edges. However, depending on the location of the maximum likelihood estimator $\hat{\theta}$, $\sup_G \int f(x|\theta)dG(\theta)$ equals

$$\begin{array}{ll} a_L f(x|\hat{\theta}) + (1 - a_L - a_U) f(x|\theta_L) + a_U f(x|\theta_U), & \hat{\theta} \leq \theta_L \\ a_L f(x|\theta_L) + (1 - a_L - a_U) f(x|\hat{\theta}) + a_U f(x|\theta_U), & \theta_L < \hat{\theta} \leq \theta_U \\ a_L f(x|\theta_L) + (1 - a_L - a_U) f(x|\theta_U) + a_U f(x|\hat{\theta}), & \hat{\theta} > \theta_U \end{array} \quad . \tag{4.14}$$

Notice that the infimum is obtained by pushing mass as far away from the MLE as allowed by the constraints, while the supremum is obtained by pushing the mass as close to the MLE as possible. In conjunction with π, the prior probability of H_0, the supremum and infimum may be used to determine the prior percentiles compatible with $P(\theta = 0|x) \leq p$ and $P(\theta \neq 0|x) \leq p$, respectively. Because $\mathcal{H}_c$ is empty if the supremum does not exceed c, we can use the supremum expression to determine whether there are any G that satisfy the inequality in (4.12), i.e., whether any priors G exist that enable stopping to reject the null hypothesis. Similarly, the infimum expression may be useful in determining whether any G enable stopping to reject the alternative hypothesis, H_1.

We may view as fixed either the (θ_L, θ_U) pair, the (a_L, a_U) pair, or both. As an example of the first case, suppose we seek the (a_L, a_U) compatible with a fixed (θ_L, θ_U) pair (an *indifference zone*) for which $\int f(x|\theta)dG(\theta) \geq c$. Then given the location of $\hat{\theta}$ with respect to the indifference zone, equation (4.14) may be easily solved to obtain the half-plane in which acceptable $\mathbf{a}$s must lie. When combined with the necessary additional constraints $a_L \geq 0, a_U \geq 0$, and $a_L + a_U \leq 1$, the result is a polygonal region that is easy to graph and interpret. Graphs of acceptable (θ_L, θ_U) pairs for fixed (a_L, a_U) may be obtained similarly, although the solution of equation (4.14) is now more involved and the resulting regions may no longer be compact. We will return to these ideas in the context of our Section 8.2 data example.

Since the ideas behind prior partitioning are somewhat more theoretical and less computational than those driving sensitivity analysis, the approach has been well developed in the literature. Two often-quoted early references are the paper by Edwards, Lindman, and Savage (1963), who explore robust Bayesian methods in the context of psychological models, and the book by Mosteller and Wallace (1964), who discuss bounds on the prior probabilities necessary to choose between two simple hypotheses (authorship of a given disputed *Federalist* paper by either Alexander Hamilton or James Madison).

The subsequent literature in the area is vast; see Berger (1985, Section 4.7) or, more recently, Berger (1994) for a comprehensive review. Here we mention only a few particularly important papers. In the point null setting,

Berger and Sellke (1987) and Berger and Delampady (1987) show that we attain the minimum of $P(\theta = \theta_0|x)$ over all conditional priors G for $\theta \neq \theta_0$ when G places all of its mass at $\hat{\theta}$, the maximum likelihood estimate of θ. Even in this case, where G is working with the data against H_0, these authors showed that the resulting $P(\theta = \theta_0|x)$ values are typically still larger than the corresponding two-sided p-value, suggesting that the standard frequentist approach is biased against H_0 in this case. In the interval null hypothesis setting, prior partitioning is reminiscent of the work of O'Hagan and Berger (1988), who obtain bounds on the posterior probability content of each of a collection of intervals that form the support of a univariate parameter, under the restriction that the prior probability assignment to these intervals is in a certain sense unimodal. In the specific context of clinical trial monitoring, Greenhouse and Wasserman (1995) compute bounds on posterior expectations and tail areas (stopping probabilities) over an ϵ-contaminated class of prior distributions (Berger and Berliner, 1986).

Further restricting the prior class

Sargent and Carlin (1996) extend the above approach to the case of an interval null hypothesis, i.e., $H_0 : \theta \in [\theta_L, \theta_U]$ versus $H_1 : \theta \notin [\theta_L, \theta_U]$. This formulation is useful in the context of clinical trial monitoring, where $[\theta_L, \theta_U]$ is thought of as an *indifference zone*, within which we are indifferent as to the use of treatment or placebo. For example, we might take $\theta_U > 0$ if there were increased costs or toxicities associated with the treatment. Let π again denote the prior probability of H_0, and let $G(\theta)$ now correspond to the prior cdf of θ given $\theta \notin [\theta_L, \theta_U]$. Making the simplifying assumption of a uniform prior over the indifference zone, the complete prior density for θ may be written as

$$p(\theta) = \frac{\pi}{\theta_U - \theta_L} I_{[\theta_L, \theta_U]}(\theta) + (1 - \pi) g(\theta) \; . \tag{4.15}$$

Sargent and Carlin (1996) derive expressions similar to (4.11) and (4.12) under the percentile restrictions of the previous subsection. However, these rather weak restrictions lead to prior classes that, while plausible, are often too broad for practical use. As such, we might consider a sequence of increasingly tight restrictions on the shape and smoothness of permissible priors, which in turn enable increasingly informative results. For example, we might retain the mixture form (4.15), but now restrict $g(\theta)$ to some particular parametric family. Carlin and Sargent (1996) refer to such a prior as "semiparametric" because the parametric form for g does not cover the indifference zone $[\theta_L, \theta_U]$, although since we have adopted another parametric form over this range (the uniform) one might argue that "biparametric" or simply "mixture" would be better names.

We leave it as an exercise to show that requiring $G \in \mathcal{H}_c$ is equivalent

to requiring $BF \leq \left(\frac{p}{1-p}\right)\left(\frac{1-\pi}{\pi}\right)$, where

$$BF = \frac{\frac{1}{\theta_U - \theta_L}\int_{\theta_L}^{\theta_U} f(x|\theta)d\theta}{\int f(x|\theta)g(\theta)d\theta}\ , \tag{4.16}$$

the Bayes factor in favor of the null hypothesis. Equation (4.16) expresses the Bayes factor as the ratio of the marginal densities under the competing hypotheses; it is also expressible as the ratio of posterior to prior odds in favor of the null. As such, BF gives the extent to which the data have revised our prior beliefs concerning the two hypotheses. Note that if we take $\pi = 1/2$ (equal prior weighting of null and alternative), then a Bayes factor of 1 suggests equal posterior support for the two hypotheses. In this case, we require a Bayes factor of 1/19 or smaller to insure that $P(H_0|x)$ does not exceed 0.05.

In practice, familiar models from the exponential family are often appropriate (either exactly or asymptotically) for the likelihood $f(x|\theta)$. This naturally leads to consideration of the restricted class of *conjugate* priors $g(\theta)$, to obtain a closed form for the integral in the denominator of (4.16). Since a normal approximation to the likelihood for θ is often suitable for even moderate sample sizes, we illustrate in the case of a conjugate normal prior. The fact that g is defined only on the complement of the indifference zone presents a slight complication, but, fortunately, the calculations remain tractable under a renormalized prior with the proper support. That is, we take

$$g(\theta) = \frac{N(\theta|\mu,\tau^2)}{1 - \left[\Phi\left(\frac{\theta_U-\mu}{\tau}\right) - \Phi\left(\frac{\theta_L-\mu}{\tau}\right)\right]}\ ,\quad \theta \notin [\theta_L, \theta_U]\ ,$$

where the numerator denotes the density of a normal distribution with mean μ and variance τ^2, and Φ denotes the cdf of a standard normal distribution.

To obtain a computational form for equation (4.16), suppose we can approximate the likelihood satisfactorily with a $N(\theta|\hat{\theta},\hat{\sigma}^2)$ density, where $\hat{\theta}$ is the maximum likelihood estimate (MLE) of θ and $\hat{\sigma}^2$ is a corresponding standard error estimate. Probability calculus then shows that

$$\begin{aligned}\int f(x|\theta)g(\theta)d\theta = \frac{1}{\sqrt{2\pi(\hat{\sigma}^2+\tau^2)}}\exp\left[-\frac{(\mu-\hat{\theta})^2}{2(\hat{\sigma}^2+\tau^2)}\right] \\ \times\left\{1 - \left[\Phi\left(\frac{\theta_U-\eta}{\nu}\right) - \Phi\left(\frac{\theta_L-\eta}{\nu}\right)\right]\right\}\ ,\end{aligned} \tag{4.17}$$

where $\eta = (\hat{\sigma}^2\mu + \tau^2\hat{\theta})/(\hat{\sigma}^2+\tau^2)$ and $\nu^2 = \hat{\sigma}^2\tau^2/(\hat{\sigma}^2+\tau^2)$. (Note that η and ν^2 are respectively the posterior mean and variance under the fully parametric normal/normal model, described in the next subsection.) Since $\int_{\theta_L}^{\theta_U} f(x|\theta)d\theta = \Phi\left(\frac{\theta_U-\hat{\theta}}{\hat{\sigma}}\right) - \Phi\left(\frac{\theta_L-\hat{\theta}}{\hat{\sigma}}\right)$, we can now obtain the Bayes factor

(4.16) without numerical integration, provided that there are subroutines available to evaluate the normal density and cdf.

As a final approach, we might abandon the mixture prior form (4.15) in favor of a single parametric family $h(\theta)$, preferably chosen as conjugate with the likelihood $f(x|\theta)$. If such a choice is possible, we obtain simple closed form expressions for Bayes factors and tail probabilities whose sensitivity to changes in the prior parameters can be easily examined. For example, for our $N(\theta|\hat{\theta}, \hat{\sigma}^2)$ likelihood under a $N(\theta|\mu, \tau^2)$ prior, the posterior probability of H_0 is nothing but

$$P(\theta \in [\theta_L, \theta_U]|x) = \Phi\left(\frac{\theta_U - \eta}{\nu}\right) - \Phi\left(\frac{\theta_L - \eta}{\nu}\right) ,$$

where η and ν^2 are again as defined beneath equation (4.17). Posterior probabilities that correspond to stopping to reject the hypotheses $H_L : \theta < \theta_L$ and $H_U : \theta > \theta_U$ arise similarly.

4.3 Model assessment

We have already presented several tools for Bayesian model assessment in Subsection 2.5.1. It is not our intention to review all of these tools, but rather to point out how their computation is greatly facilitated by modern Monte Carlo computational methods. Because most Bayesian models encountered in practice require some form of sampling to evaluate the posterior distributions of interest, this means that common model checks will be available at little extra cost, both in terms of programming and runtime.

Consider for example the simple Bayesian residual,

$$r_i = y_i - E(y_i|\mathbf{z}),\ i = 1, \ldots, n,$$

where $\mathbf{z}$ is the sample of data used to fit the model and $\mathbf{y} = (y_1, \ldots, y_n)'$ is an independent validation sample. Clearly, calculation of r_i requires an expectation with respect to the posterior predictive distribution $p(y_i|\mathbf{z})$, which will rarely be available in closed form. However, notice that we can write

$$\begin{aligned} E(y_i|\mathbf{z}) &= \int y_i p(y_i|\mathbf{z}) dy_i \\ &= \int\int y_i f(y_i|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{z}) d\boldsymbol{\theta} dy_i \\ &= \int E(y_i|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{z}) d\boldsymbol{\theta} \\ &\approx \frac{1}{G}\sum_{g=1}^{G} E(y_i|\boldsymbol{\theta}^{(g)}) , \end{aligned}$$

where the $\boldsymbol{\theta}^{(g)}$ are samples from the posterior distribution $p(\boldsymbol{\theta}|\mathbf{z})$ (for

MCMC algorithms, we would of course include only post-convergence samples). The equality in the second line holds due to the conditional independence of $\mathbf{y}$ and $\mathbf{z}$ given $\boldsymbol{\theta}$, while that in the third line arises from reversing the order of integration. But $E(y_i|\boldsymbol{\theta})$ typically *will* be available in closed form, since this is nothing but the mean structure of the likelihood. The fourth line thus arises as a Monte Carlo integration.

Even if $E(y_i|\boldsymbol{\theta})$ is not available in closed form, we can still estimate $E(y_i|\mathbf{z})$ provided we can draw samples $y_i^{(g)}$ from $f(y_i|\boldsymbol{\theta}^{(g)})$. Such sampling is naturally appended onto the algorithm generating the $\boldsymbol{\theta}^{(g)}$ samples themselves. In this case we have

$$\begin{aligned} E(y_i|\mathbf{z}) &= \int\int y_i f(y_i|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{z}) d\boldsymbol{\theta} dy_i \\ &\approx \frac{1}{G}\sum_{g=1}^{G} y_i^{(g)} , \end{aligned}$$

since $(y_i^{(g)}, \boldsymbol{\theta}^{(g)})$ constitute a sample from $p(y_i, \boldsymbol{\theta}|\mathbf{z})$. This estimator will not be as accurate as the first (because both integrals are now being done via Monte Carlo), but will still be simulation consistent (i.e., converge to the true value with probability 1 as $G \to \infty$).

Next, consider the cross-validation residual given in (2.27),

$$r_i = y_i - E(y_i|\mathbf{y}_{(i)}),\ i = 1, \ldots, n,$$

where we recall that $\mathbf{y}_{(i)}$ denotes the vector of all the data except the i^{th} value. Now we have

$$\begin{aligned} E(y_i|\mathbf{y}_{(i)}) &= \int E(y_i|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{y}_{(i)}) d\boldsymbol{\theta} \\ &\approx \int E(y_i|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{y}) d\boldsymbol{\theta} \\ &\approx \frac{1}{G}\sum_{g=1}^{G} E(y_i|\boldsymbol{\theta}^{(g)}) , \end{aligned}$$

where the $\boldsymbol{\theta}^{(g)}$ are samples from the complete data posterior $p(\boldsymbol{\theta}|\mathbf{y})$. The approximation in the second line should be adequate unless the dataset is small and y_i is an extreme outlier. It enables the use of the same $\boldsymbol{\theta}^{(g)}$ samples (already produced by the Monte Carlo algorithm) for estimating each conditional mean $E(y_i|\mathbf{y}_{(i)})$, hence each residual r_i, for $i = 1, \ldots, n$.

Finally, to obtain the cross-validation standardized residual in (2.28), we require not only $E(y_i|\mathbf{y}_{(i)})$, but also $Var(y_i|\mathbf{y}_{(i)})$. This quantity is estimable by first rewriting $Var(y_i|\mathbf{y}_{(i)}) = E(y_i^2|\mathbf{y}_{(i)}) - [E(y_i|\mathbf{y}_{(i)})]^2$, and then

observing

$$\begin{aligned} E(y_i^2|\mathbf{y}_{(i)}) &= \int E(y_i^2|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y}_{(i)})d\boldsymbol{\theta} \\ &= \int \{Var(y_i|\boldsymbol{\theta}) + [E(y_i|\boldsymbol{\theta})]^2\}p(\boldsymbol{\theta}|\mathbf{y}_{(i)})d\boldsymbol{\theta} \\ &\approx \frac{1}{G}\sum_{g=1}^{G}\left\{Var(y_i|\boldsymbol{\theta}^{(g)}) + [E(y_i|\boldsymbol{\theta}^{(g)})]^2\right\} , \end{aligned}$$

again approximating the reduced data posterior with the full data posterior and performing a Monte Carlo integration.

Other posterior quantities from Subsection 2.5.1, such as conditional predictive ordinates $f(y_i|\mathbf{y}_{(i)})$ and posterior predictive model checks (e.g., Bayesian p-values) can be derived similarly. The details of some of these calculations are left as exercises.

4.4 Bayes factors via marginal density estimation

As in the previous section, our goal in this section is not to review the methods for model choice and averaging that have already been presented in Subsections 2.3.3 and 2.5.2, respectively. Rather, we provide several computational approaches for obtaining these quantities, especially the Bayes factor. In Section 4.6, we will present some more advanced approaches that are not only implemented via MCMC methods, but also motivated by these methods, in the sense that they are applicable in very highly-parametrized model settings where the traditional model selection method offered by Bayes factors is unavailable or infeasible.

Recall the use of the Bayes factor as a method for choosing between two competing models M_1 and M_2, given in equation (2.19) as

$$BF = \frac{p(\mathbf{y} \mid M_1)}{p(\mathbf{y} \mid M_2)} , \tag{4.18}$$

the ratio of the observed marginal densities for the two models. For large sample sizes n, the necessary integrals over $\boldsymbol{\theta}_i$ may be available conveniently and accurately via asymptotic approximation, producing estimated Bayes factors accurate to order $O(1/n)$, where n is the number of factors contributing to the likelihood function. In any case, equation (4.10) reveals that asymptotic approximations are helpful in discovering the sensitivity of Bayes factors to prior specification, whether they played a role in the original calculation or not.

For moderate sample sizes n or reasonably challenging models, however, such approximations are not appropriate, and sampling-based methods must be used to obtain estimates of the marginal likelihoods needed to evaluate BF. This turns out to be a surprisingly difficult problem; unlike

posterior and predictive distributions, marginal distributions are not easily estimated from the output of an MCMC algorithm. As a result, many different approaches have been suggested in the literature, all of which involve augmenting or otherwise "tricking" a sampler into producing the required marginal density estimates. As with our Subsection 3.4.6 discussion of convergence diagnostics, we do not attempt a technical review of every method, but we do comment on their strengths and weaknesses so that the reader can judge which will likely be the most appropriate in a given problem setting. Kass and Raftery (1995) provide a comprehensive review of Bayes factors, including many of the computational methods described below.

4.4.1 Direct methods

In what follows, we suppress the dependence on the model indicator M in our notation because all our calculations must be repeated for both models appearing in expression (4.18). Observe that since

$$p(\mathbf{y}) = \int f(\mathbf{y} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta}) d\boldsymbol{\theta} \ ,$$

we could generate observations $\{\boldsymbol{\theta}^{(g)}\}_{g=1}^{G}$ from the prior and compute the estimate

$$\hat{p}(\mathbf{y}) = \frac{1}{G} \sum_{g=1}^{G} f(\mathbf{y} \mid \boldsymbol{\theta}^{(g)}) \ , \tag{4.19}$$

a simple Monte Carlo integration. Unfortunately, the conditional likelihood $f(\mathbf{y} \mid \boldsymbol{\theta})$ will typically be very peaked compared to the prior $p(\boldsymbol{\theta})$, so that (4.19) will be a very inefficient estimator (most of the terms in the sum will be near 0). A better approach would be to use samples from the posterior distribution, as suggested by Newton and Raftery (1994). These authors develop the estimator

$$\hat{p}(\mathbf{y}) = \left[\frac{1}{G} \sum_{g=1}^{G} \frac{1}{f(\mathbf{y} \mid \boldsymbol{\theta}^{(g)})} \right]^{-1} , \tag{4.20}$$

the harmonic mean of the posterior sample conditional likelihoods. This approach, while efficient, can be quite unstable since a few of the conditional likelihood terms in the sum will still be near 0. Theoretically, this difficulty corresponds to this estimator's failure to obey a Gaussian central limit theorem as $G \to \infty$. To correct this, Newton and Raftery suggest a compromise between methods (4.19) and (4.20) wherein we use a mixture of the prior and posterior densities for each model, $\tilde{p}(\boldsymbol{\theta}) = \delta p(\boldsymbol{\theta}) + (1 - \delta) p(\boldsymbol{\theta} | \mathbf{y})$, as an importance sampling density. Defining $w(\boldsymbol{\theta}) = p(\boldsymbol{\theta}) / \tilde{p}(\boldsymbol{\theta})$, we then have

$$\hat{p}(\mathbf{y}) = \frac{\sum_{g=1}^{G} f(\mathbf{y} \mid \boldsymbol{\theta}^{(g)}) \, w(\boldsymbol{\theta}^{(g)})}{\sum_{g=1}^{G} w(\boldsymbol{\theta}^{(g)})} \ .$$

This estimator is more stable and does satisfy a central limit theorem, although in this form it does require sampling from both the posterior and the prior.

A useful generalization of (4.20) was provided by Gelfand and Dey (1994), who began with the identity

$$[p(\mathbf{y})]^{-1} = \int \frac{h(\boldsymbol{\theta})}{f(\mathbf{y}|\boldsymbol{\theta})\, p(\boldsymbol{\theta})}\, p(\boldsymbol{\theta}|\mathbf{y}) d\boldsymbol{\theta}\ ,$$

which holds for any proper density h. Given samples $\boldsymbol{\theta}^{(g)}$ from the posterior distribution, this suggests the estimator

$$\hat{p}(\mathbf{y}) = \left[\frac{1}{G}\sum_{g=1}^{G} \frac{h(\boldsymbol{\theta}^{(g)})}{f(\mathbf{y}|\boldsymbol{\theta}^{(g)})\, p(\boldsymbol{\theta}^{(g)})}\right]^{-1} . \tag{4.21}$$

Notice that taking $h(\boldsymbol{\theta}) = p(\boldsymbol{\theta})$, the prior density, produces (4.20). However, this is a poor choice from the standpoint of importance sampling theory, which instead suggests choosing h to roughly match the posterior density. Gelfand and Dey suggest a multivariate normal or t density with mean and covariance matrix estimated from the $\boldsymbol{\theta}^{(g)}$ samples. Kass and Raftery (1995) observe that this estimator does satisfy a central limit theorem provided that

$$\int \frac{h^2(\boldsymbol{\theta})}{f(\mathbf{y}|\boldsymbol{\theta})\, p(\boldsymbol{\theta})}\, d\boldsymbol{\theta} < \infty\ ,$$

i.e., the tails of h are relatively thin. The estimator does seem to perform well in practice unless $\boldsymbol{\theta}$ is of too high a dimension, complicating the selection of a good h density.

The material in this section is closely related to what Tan et al. (2008) refer to as *inverse Bayes formulae*, which use a reexpression of Bayes' Rule to express a particular marginal distribution in terms of available conditional distributions. Such formulae can allow one to avoid MCMC computation (for which we have seen convergence assessment is often problematic) for both posterior estimation and model comparison in a fairly broad class of missing data problems. In particular, these authors stress the use of noniterative Monte Carlo and EM computational methods. Still, even with data augmentation there is a limit to the generality of the hierarchical models one can analyze with this technology.

4.4.2 Using Gibbs sampler output

In the case of models fit via Gibbs sampling with closed-form full conditional distributions, Chib (1995) offers an alternative that avoids the specification of the h function above. The method begins by simply rewriting

Bayes' rule as

$$p(\mathbf{y}) = \frac{f(\mathbf{y}|\boldsymbol{\theta})\, p(\boldsymbol{\theta})}{p(\boldsymbol{\theta}|\mathbf{y})} \ .$$

Only the denominator on the right-hand side is unknown, so an estimate of the posterior would produce an estimate of the marginal density, as we desire. But because this identity holds for *any* $\boldsymbol{\theta}$ value, we require only a posterior density estimate at a single point – say, $\boldsymbol{\theta}'$. Therefore, we have

$$\log \hat{p}(\mathbf{y}) = \log f(\mathbf{y}|\boldsymbol{\theta}') + \log p(\boldsymbol{\theta}') - \log \hat{p}(\boldsymbol{\theta}'|\mathbf{y}) \ , \tag{4.22}$$

where we have switched to the log scale to improve computational accuracy. While in theory $\boldsymbol{\theta}'$ is arbitrary, Chib (1995) suggests choosing it as a point of high posterior density, again to maximize accuracy in (4.22).

It remains to show how to obtain the estimate $\hat{p}(\boldsymbol{\theta}'|\mathbf{y})$. We first describe the technique in the case where the parameter vector can be decomposed into two blocks (similar to the data augmentation scenario of Tanner and Wong, 1987). That is, we suppose that $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$, where $p(\boldsymbol{\theta}_1|\mathbf{y}, \boldsymbol{\theta}_2)$ and $p(\boldsymbol{\theta}_2|\mathbf{y}, \boldsymbol{\theta}_1)$ are both available in closed form. Writing

$$p(\boldsymbol{\theta}'_1, \boldsymbol{\theta}'_2|\mathbf{y}) = p(\boldsymbol{\theta}'_2|\mathbf{y}, \boldsymbol{\theta}'_1) p(\boldsymbol{\theta}'_1|\mathbf{y}) \ , \tag{4.23}$$

we observe that the first term on the right-hand side is available explicitly at $\boldsymbol{\theta}'$, while the second can be estimated via the "Rao-Blackwellized" mixture estimate (3.9), namely,

$$\hat{p}(\boldsymbol{\theta}'_1|\mathbf{y}) = \frac{1}{G} \sum_{g=1}^{G} p(\boldsymbol{\theta}'_1|\mathbf{y}, \boldsymbol{\theta}_2^{(g)}) \ , \tag{4.24}$$

since $\boldsymbol{\theta}_2^{(g)} \sim p(\boldsymbol{\theta}_2|\mathbf{y})$, $g = 1, \ldots, G$. Thus our marginal density estimate in (4.22) becomes

$$\log \hat{p}(\mathbf{y}) = \log f(\mathbf{y}|\boldsymbol{\theta}'_1, \boldsymbol{\theta}'_2) + \log p(\boldsymbol{\theta}'_1, \boldsymbol{\theta}'_2) - \log p(\boldsymbol{\theta}'_2|\mathbf{y}, \boldsymbol{\theta}'_1) - \log \hat{p}(\boldsymbol{\theta}'_1|\mathbf{y}) \ .$$

Exponentiating produces the final marginal density estimate.

Next, suppose there are three parameter blocks, $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3)$. The decomposition of the joint posterior density (4.23) now becomes

$$p(\boldsymbol{\theta}'_1, \boldsymbol{\theta}'_2, \boldsymbol{\theta}'_3|\mathbf{y}) = p(\boldsymbol{\theta}'_3|\mathbf{y}, \boldsymbol{\theta}'_1, \boldsymbol{\theta}'_2) p(\boldsymbol{\theta}'_2|\mathbf{y}, \boldsymbol{\theta}'_1) p(\boldsymbol{\theta}'_1|\mathbf{y}) \ .$$

Again the first term is available explicitly, while the third term may be estimated as a mixture of $p(\boldsymbol{\theta}'_1|\mathbf{y}, \boldsymbol{\theta}_2^{(g)}, \boldsymbol{\theta}_3^{(g)})$, $g = 1, \ldots, G$, similar to (4.24) above. For the second term, we may write

$$p(\boldsymbol{\theta}'_2|\mathbf{y}, \boldsymbol{\theta}'_1) = \int p(\boldsymbol{\theta}'_2|\mathbf{y}, \boldsymbol{\theta}'_1, \boldsymbol{\theta}_3) p(\boldsymbol{\theta}_3|\mathbf{y}, \boldsymbol{\theta}'_1) d\boldsymbol{\theta}_3 \ ,$$

suggesting the estimator

$$\hat{p}(\boldsymbol{\theta}'_2|\mathbf{y}, \boldsymbol{\theta}'_1) = \frac{1}{G} \sum_{g=1}^{G} p(\boldsymbol{\theta}'_2|\mathbf{y}, \boldsymbol{\theta}'_1, \boldsymbol{\theta}_3^{*(g)}) \ ,$$

where $\boldsymbol{\theta}_3^{*(g)} \sim p(\boldsymbol{\theta}_3|\mathbf{y}, \boldsymbol{\theta}'_1)$. Such draws are *not* available from the original posterior sample, which instead contains $\boldsymbol{\theta}_3^{(g)} \sim p(\boldsymbol{\theta}_3|\mathbf{y})$. However, we may produce them simply by continuing the Gibbs sampler for an additional G iterations with only two full conditional distributions, namely

$$p(\boldsymbol{\theta}_2|\mathbf{y}, \boldsymbol{\theta}'_1, \boldsymbol{\theta}_3) \text{ and } p(\boldsymbol{\theta}_3|\mathbf{y}, \boldsymbol{\theta}'_1, \boldsymbol{\theta}_2) \, .$$

Thus, while additional sampling is required, new computer code is not; we need only continue with a portion of the old code. The final marginal density estimate then arises from

$$\begin{aligned} \log \hat{p}(\mathbf{y}) = &\log f(\mathbf{y}|\boldsymbol{\theta}'_1, \boldsymbol{\theta}'_2, \boldsymbol{\theta}'_3) + \log p(\boldsymbol{\theta}'_1, \boldsymbol{\theta}'_2, \boldsymbol{\theta}'_3) \\ &- \log p(\boldsymbol{\theta}'_3|\mathbf{y}, \boldsymbol{\theta}'_1, \boldsymbol{\theta}'_2) - \log \hat{p}(\boldsymbol{\theta}'_2|\mathbf{y}, \boldsymbol{\theta}'_1) - \log \hat{p}(\boldsymbol{\theta}'_1|\mathbf{y}) \, . \end{aligned}$$

The extension to B parameter blocks requires a similar factoring of the joint posterior into B components, with $(B-1)$ Gibbs sampling runs of G samples each to estimate the various factors. We note that clever partitioning of the parameter vector into only a few blocks (each still having a closed form full conditional) can increase computational accuracy and reduce programming and sampling time as well.

Extension of this basic approach to more difficult model settings is possible. For instance, Basu and Chib (2003) apply it to the Dirichlet process (DP) mixture model setting of Subsection 2.6. This paper resolves the issue of calculation of the likelihood ordinate using a collapsed sequential importance sampling (SIS) algorithm.

4.4.3 Using Metropolis-Hastings output

Equations like (4.24) require us to know the normalizing constant for the full conditional distribution $p(\boldsymbol{\theta}_1|\mathbf{y}, \boldsymbol{\theta}_2)$, thus precluding their use with full conditionals updated using Metropolis-Hastings (rather than Gibbs) steps. To remedy this, Chib and Jeliazkov (2001) extend the approach, which takes a particularly simple form in the case where the parameter vector $\boldsymbol{\theta}$ can be updated in a single block. Let

$$\alpha(\boldsymbol{\theta}, \boldsymbol{\theta}^*|\mathbf{y}) = \min \left\{ 1, \frac{p(\boldsymbol{\theta}^*|\mathbf{y}) q(\boldsymbol{\theta}^*, \boldsymbol{\theta}|\mathbf{y})}{p(\boldsymbol{\theta}|\mathbf{y}) q(\boldsymbol{\theta}, \boldsymbol{\theta}^*|\mathbf{y})} \right\} \, ,$$

the probability of accepting a Metropolis-Hastings candidate $\boldsymbol{\theta}^*$ generated from a candidate density $q(\boldsymbol{\theta}, \boldsymbol{\theta}^*|\mathbf{y})$ (note that this density is allowed to depend on the data $\mathbf{y}$). Chib and Jeliazkov (2001) then show

$$p(\boldsymbol{\theta}'|\mathbf{y}) = \frac{E_1 \left\{ \alpha(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{y}) q(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{y}) \right\}}{E_2 \left\{ \alpha(\boldsymbol{\theta}', \boldsymbol{\theta}|\mathbf{y}) \right\}} \, , \tag{4.25}$$

where E_1 is the expectation with respect to the posterior $p(\boldsymbol{\theta}|\mathbf{y})$ and E_2 is the expectation with respect to the candidate density $q(\boldsymbol{\theta}', \boldsymbol{\theta}|\mathbf{y})$. The numerator is then estimated by averaging the product in braces with respect

to draws from the posterior, while the denominator is estimated by averaging the acceptance probability with respect to draws from $q(\boldsymbol{\theta}', \boldsymbol{\theta}|\mathbf{y})$, given the fixed value $\boldsymbol{\theta}'$. Note that this calculation does not require knowledge of the normalizing constant for $p(\boldsymbol{\theta}|\mathbf{y})$. Plugging this estimate of (4.25) into (4.22) completes the estimation of the marginal likelihood. When there are two more more blocks, Chib and Jeliazkov (2001) illustrate an extended version of this algorithm using multiple MCMC runs, similar to the Chib (1995) approach for the Gibbs sampler outlined in the previous subsection.

4.5 Bayes factors via sampling over the model space

The methods of the previous section all seek to estimate the marginal density $p(\mathbf{y})$ for each model, and subsequently calculate the Bayes factor via equation (4.18). They also operate on a posterior sample that has already been produced by some noniterative or MCMC method, though the methods of Chib (1995) and Chib and Jeliazkov (2001) will often require multiple runs of slightly different versions of the MCMC algorithm to produce the necessary output. But for some complicated or high-dimensional model settings, such as spatial models using Markov random field priors (which involve large numbers of random effect parameters that cannot be analytically integrated out of the likelihood nor readily updated in blocks), these methods may be infeasible.

An alternative approach favored by many authors is to include the model indicator M as a parameter in the sampling algorithm itself. This of course complicates the initial sampling process, but has the important benefit of producing a stream of samples $\{M^{(g)}\}_{g=1}^{G}$ from $p(M|\mathbf{y})$, the marginal posterior distribution of the model indicator. Hence, the ratio

$$\hat{p}(M = j|\mathbf{y}) = \frac{\text{number of } M^{(g)} = j}{\text{total number of } M^{(g)}}, \quad j = 1, \ldots, K\,, \tag{4.26}$$

provides a simple estimate of each posterior model probability, which may then be used to compute the Bayes factor between any two of the models, say j and j', via

$$\widehat{BF}_{jj'} = \frac{\hat{p}(M = j|\mathbf{y})/\hat{p}(M = j'|\mathbf{y})}{p(M = j)/p(M = j')}\,, \tag{4.27}$$

the original formula used to define the Bayes factor in (2.18). Estimated variances of the estimates in (4.26) are easy to obtain even if the $M^{(g)}$ output stream exhibits autocorrelation through the batching formula (3.30) or other methods mentioned in Subsection 3.4.6.

Sampling over the model space alone

Most of the methods we will discuss involve algorithmic searches over both the model and the parameter spaces simultaneously. This is a natural way

to think about the problem, but like any other augmented sampler such an approach risks a less well-identified parameter space, increased correlations, and hence slower convergence than a sampler operating on any one of the models alone. Moreover, if our interest truly lies only in computing posterior model probabilities $p(M = j|\mathbf{y})$ or a Bayes factor, the parameter samples are not needed, relegating the $\boldsymbol{\theta}_j$ to nuisance parameter status.

These thoughts motivate the creation of samplers that operate on the model space alone. This in turn requires us to integrate the $\boldsymbol{\theta}_j$ out of the model before sampling begins. To obtain such marginalized expressions in closed form requires fairly specialized likelihood and prior settings, but several authors have made headway in this area for surprisingly broad classes of models. For example, Madigan and York (1995) offer an algorithm for searching over a space of graphical models for discrete data, an approach they refer to as Markov chain Monte Carlo model composition, or $(MC)^3$. Raftery, Madigan, and Hoeting (1997) work instead in the multiple regression setting with conjugate priors, again enabling a model-space-only search. They compare the $(MC)^3$ approach with the "Occam's Window" method of Madigan and Raftery (1994). Finally, Clyde, DeSimone, and Parmigiani (1996) use importance sampling (not MCMC) to search for the most promising models in a hierarchical normal linear model setting. They employ an orthogonalization of the design matrix, which enables impressive gains in efficiency over the Metropolis-based $(MC)^3$ method.

Sampling over model and parameter space

Unfortunately, most model settings are too complicated to allow the entire parameter vector $\boldsymbol{\theta}$ to be integrated out of the joint posterior in closed form, and thus require that any MCMC model search be over the model and parameter space jointly. Such searches date at least to the work of Carlin and Polson (1991), who included M as a parameter in the Gibbs sampler in concert with the scale mixture of normals idea mentioned in Subsection 4.2.1. In this way, they computed Bayes factors and compared marginal posterior densities for a parameter of interest under changing specification of the model error densities and related prior densities. Their algorithm required that the models share the same parametrization, however, and so would not be appropriate for comparing two different mean structures (say, a linear and a quadratic). George and McCulloch (1993) circumvented this problem for multiple regression models by introducing latent indicator variables that determine whether or not a particular regression coefficient may be safely estimated by 0, a process they referred to as *stochastic search variable selection* (SSVS). Unfortunately, in order to satisfy the Markov convergence requirement of the Gibbs sampler, a regressor can never completely "disappear" from the model, so the Bayes factors obtained necessarily depend on values of user-chosen tuning constants.

4.5.1 Product space search

Carlin and Chib (1995), whom we abbreviate as "CC", present a Gibbs sampling method that avoids these theoretical convergence difficulties and accommodates completely general model settings. Suppose there are K candidate models, and corresponding to each, there is a distinct parameter vector $\boldsymbol{\theta}_j$ having dimension n_j, $j = 1, \ldots, K$. Our interest lies in $p(M = j|\mathbf{y})$, $j = 1, \ldots, K$, the posterior probabilities of each of the K models, and possibly the model-specific posterior distributions $p(\boldsymbol{\theta}_j|M = j, \mathbf{y})$ as well.

Supposing that $\boldsymbol{\theta}_j \in \Re^{n_j}$, the approach is essentially to sample over the model indicator and the *product* space $\prod_{j=1}^{K} \Re^{n_j}$. This entails viewing the prior distributions as model specific and part of the Bayesian model specification. That is, corresponding to model j we write the likelihood as $f(\mathbf{y}|\boldsymbol{\theta}_j, M = j)$ and the prior as $p(\boldsymbol{\theta}_j|M = j)$. Since we are assuming that M merely provides an indicator as to which particular $\boldsymbol{\theta}_j$ is relevant to $\mathbf{y}$, we have that $\mathbf{y}$ is independent of $\{\boldsymbol{\theta}_{i\neq j}\}$ given that $M = j$. In addition, since our primary goal is the computation of Bayes factors, we assume that each prior $p(\boldsymbol{\theta}_j|M = j)$ is proper (though possibly quite vague). For simplicity we assume complete independence among the various $\boldsymbol{\theta}_j$ given the model indicator M, and thus may complete the Bayesian model specification by choosing proper "pseudopriors," $p(\boldsymbol{\theta}_j|M \neq j)$. Writing $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\}$, our conditional independence assumptions imply that

$$\begin{aligned} p(\mathbf{y}|M = j) &= \int f(\mathbf{y}|\boldsymbol{\theta}, M = j)p(\boldsymbol{\theta}|M = j)d\boldsymbol{\theta} \\ &= \int f(\mathbf{y}|\boldsymbol{\theta}_j, M = j)p(\boldsymbol{\theta}_j|M = j)d\boldsymbol{\theta}_j \,, \end{aligned} \tag{4.28}$$

and so the form given to $p(\boldsymbol{\theta}_j|M \neq j)$ is irrelevant. Thus, as the name suggests, a pseudoprior is not really a prior at all, but only a conveniently chosen linking density, used to completely define the joint model specification. Hence the joint distribution of $\mathbf{y}$ and $\boldsymbol{\theta}$ given $M = j$ is

$$p(\mathbf{y}, \boldsymbol{\theta}, M = j) \;=\; f(\mathbf{y}|\boldsymbol{\theta}_j, M = j) \left\{ \prod_{i=1}^{K} p(\boldsymbol{\theta}_i|M = j) \right\} \pi_j \,,$$

where $\pi_j \equiv P(M = j)$ is the prior probability assigned to model j, satisfying $\sum_{j=1}^{K} \pi_j = 1$.

In order to implement the Gibbs sampler, we need the full conditional distributions of each $\boldsymbol{\theta}_j$ and M. The former is given by

$$p(\boldsymbol{\theta}_j|\boldsymbol{\theta}_{i\neq j}, M, \mathbf{y}) \propto \begin{cases} f(\mathbf{y}|\boldsymbol{\theta}_j, M = j)p(\boldsymbol{\theta}_j|M = j) \,, & M = j \\ p(\boldsymbol{\theta}_j|M \neq j) \,, & M \neq j \end{cases} .$$

That is, when $M = j$ we generate from the usual model j full conditional; when $M \neq j$ we generate from the pseudoprior. Both of these generations are straightforward provided $p(\boldsymbol{\theta}_j|M = j)$ is taken to be conjugate with its

likelihood. The full conditional for M is

$$p(M=j|\boldsymbol{\theta},\mathbf{y}) = \frac{f(\mathbf{y}|\boldsymbol{\theta}_j, M=j)\left\{\prod_{i=1}^{K} p(\boldsymbol{\theta}_i|M=j)\right\}\pi_j}{\sum_{k=1}^{K} f(\mathbf{y}|\boldsymbol{\theta}_k, M=k)\left\{\prod_{i=1}^{K} p(\boldsymbol{\theta}_i|M=k)\right\}\pi_k} \ . \quad (4.29)$$

Since M is a discrete finite parameter, its generation is routine as well. Therefore, all the required full conditional distributions are well defined and, under the usual regularity conditions (Roberts and Smith, 1993), the algorithm will produce samples from the correct joint posterior distribution. In particular, equations (4.26) and (4.27) can now be used to estimate the posterior model probabilities and the Bayes factor between any two models, respectively.

Notes on implementation

Notice that, in contrast to equation (4.26), summarization of the totality of $\boldsymbol{\theta}_j^{(g)}$ samples is not appropriate. This is because what is of interest in our setting is not the marginal posterior densities $p(\boldsymbol{\theta}_j|\mathbf{y})$, but rather the *conditional* posterior densities $p(\boldsymbol{\theta}_j|M=j,\mathbf{y})$. However, suppose that in addition to $\boldsymbol{\theta}$ we have a vector of nuisance parameters η, common to all models. Then the fully marginal posterior density of η, $p(\eta|\mathbf{y})$, may be of some interest. Because the data are informative about η regardless of the value of M, a pseudoprior for η is not required. Still, care must be taken to ensure that η has the same interpretation in both models. For example, suppose we wish to choose between the two nested regression models

$$M=1: \ \ y_i = \alpha + \epsilon_i, \qquad \epsilon_i \overset{iid}{\sim} \text{Normal}(0,\sigma^2) \ ,$$

and

$$M=2: \ \ y_i = \alpha + \beta x_i + \epsilon_i, \quad \epsilon_i \overset{iid}{\sim} \text{Normal}(0,\tau^2) \ ,$$

both for $i=1,\ldots,n$. In the above notation, $\boldsymbol{\theta}_1=\sigma$, $\boldsymbol{\theta}_2=(\beta,\tau)$, and $\eta=\alpha$. Notice that α is playing the role of a grand mean in model 1, but is merely an intercept in model 2. The corresponding posteriors could be quite different if, for example, the observed y_i values were centered near 0, while those for x_i were centered far from 0. Besides being unmeaningful from a practical point of view, the resulting $p(\alpha|\mathbf{y})$ would likely be bimodal, a shape that could wreak great havoc with convergence of the MCMC algorithm, since jumps between $M=1$ and 2 would be extremely unlikely.

Poor choices of the pseudopriors $p(\boldsymbol{\theta}_j|M\neq j)$ can have a similar deleterious effect on convergence. Good choices will produce $\boldsymbol{\theta}_j^{(g)}$ values that are consistent with the data, so that $p(M=j|\boldsymbol{\theta},\mathbf{y})$ will still be reasonably large at the next M update step. Failure to generate competitive pseudoprior values will again result in intolerably high autocorrelations in the $M^{(g)}$ chain, hence slow convergence. As such, matching the pseudopriors as nearly as possible to the true model-specific posteriors is recommended.

This can be done using first-order (normal) approximations or other parametric forms designed to mimic the output from K individual preliminary MCMC runs. Note that we are *not* using the data to help select the prior, but only the pseudoprior.

If for a particular dataset one of the $p(M = j|\mathbf{y})$ is extremely large, the realized chain will exhibit slow convergence due to the resulting near-absorbing state in the algorithm. In this case, the π_j may be adjusted to correct the imbalance; the final Bayes factor computed from (4.18) will still reflect the true odds in favor of $M = j$ suggested by the data.

Finally, one might plausibly consider the modified version of the above algorithm that skips the generation of actual pseudoprior values, and simply keeps $\boldsymbol{\theta}_j^{(g)}$ at its current value when $M^{(g)} \neq j$. While this MCMC algorithm is no longer a Gibbs sampler, Green and O'Hagan (1998) show that it does still converge to the correct stationary distribution (i.e., its transition kernel does satisfy detailed balance). Under this alternative we would be free to choose $p(\boldsymbol{\theta}_j|M=i) = p(\boldsymbol{\theta}_j|M=j)$ for all i, meaning that all of these terms cancel from the M update step in equation (4.29). This pseudoprior-free version of the algorithm thus avoids the need for preliminary runs, and greatly reduces the generation and storage burdens in the final model choice run. Unfortunately, Green and O'Hagan also find it to perform quite poorly, since without the variability in candidate $\boldsymbol{\theta}_j^{(g)}$ values created by the pseudopriors, switches between models are rare. For more recent developments and refinements along these same lines, see Congdon (2007b).

Scott (2002) and Congdon (2006) suggest skipping the pseudoprior generation and simply estimating the marginal model probabilities (4.28) using independent, model-specific MCMC runs. However, Robert and Marin (2008) show that such algorithms are biased, because they do not sample over the correct *joint* posterior $p(\boldsymbol{\theta}, M|\mathbf{y})$, required for validity of the product space approach.

4.5.2 "Metropolized" product space search

Dellaportas, Forster, and Ntzoufras (2002) propose a hybrid Gibbs-Metropolis strategy. In their strategy, the model selection step is based on a proposal for a move from model j to j', followed by acceptance or rejection of this proposal. That is, the method is a "Metropolized Carlin and Chib" (MCC) approach, which proceeds as follows:

1. Let the current state be $(j, \boldsymbol{\theta}_j)$, where $\boldsymbol{\theta}_j$ is of dimension n_j.
2. Propose a new model j' with probability $h(j, j')$.
3. Generate $\boldsymbol{\theta}_{j'}$ from a pseudoprior $p(\boldsymbol{\theta}_{j'}|M \neq j')$ as in Carlin and Chib's product space search method.

4. Accept the proposed move (from j to j') with probability

$$\alpha_{j\to j'} = \min\left\{1,\ \frac{f(\mathbf{y}|\boldsymbol{\theta}_{j'}, M=j')p(\boldsymbol{\theta}_{j'}|M=j')p(\boldsymbol{\theta}_j|M=j')\pi_{j'}h(j',j)}{f(\mathbf{y}|\boldsymbol{\theta}_j, M=j)p(\boldsymbol{\theta}_j|M=j)p(\boldsymbol{\theta}_{j'}|M=j)\pi_j h(j,j')}\right\}.$$

Thus by "Metropolizing" the model selection step, the MCC method needs to sample only from the pseudoprior for the proposed model j'. In this method, the move is a Gibbs step or a sequence of Gibbs steps when $j' = j$. Posterior model probabilities and Bayes factors can be estimated as before.

4.5.3 Reversible jump MCMC

This method, originally due to Green (1995), is another strategy that samples over the model and parameter space, but which avoids the full product space search of the Carlin and Chib (1995) method (and the associated pseudoprior specification and sampling), at the cost of a less straightforward algorithm operating on the *union* space, $\mathcal{M}\times\bigcup_{j\in\mathcal{M}}\Theta_j$. It generates a Markov chain that can "jump" between models with parameter spaces of different dimensions, while retaining the aperiodicity, irreducibility, and detailed balance conditions necessary for MCMC convergence.

A typical reversible jump algorithm proceeds as follows.

1. Let the current state of the Markov chain be $(j, \boldsymbol{\theta}_j)$, where $\boldsymbol{\theta}_j$ is of dimension n_j.
2. Propose a new model j' with probability $h(j, j')$.
3. Generate $\mathbf{u}$ from a proposal density $q(\mathbf{u}|\boldsymbol{\theta}_j, j, j')$.
4. Set $(\boldsymbol{\theta}'_{j'}, \mathbf{u}') = \mathbf{g}_{j,j'}(\boldsymbol{\theta}_j, \mathbf{u})$, where $\mathbf{g}_{j,j'}$ is a deterministic function that is 1-1 and onto. This is a "dimension matching" function, specified so that $n_j + dim(\mathbf{u}) = n_{j'} + dim(\mathbf{u}')$.
5. Accept the proposed move (from j to j') with probability $\alpha_{j\to j'}$, which is the minimum of 1 and

$$\frac{f(\mathbf{y}|\boldsymbol{\theta}'_{j'}, M=j')p(\boldsymbol{\theta}'_{j'}|M=j')\pi_{j'}h(j',j)q(\mathbf{u}'|\boldsymbol{\theta}_{j'}, j', j)}{f(\mathbf{y}|\boldsymbol{\theta}_j, M=j)p(\boldsymbol{\theta}_j|M=j)\pi_j h(j,j')q(\mathbf{u}|\boldsymbol{\theta}_j, j, j')}\left|\frac{\partial\mathbf{g}(\boldsymbol{\theta}_j, \mathbf{u})}{\partial(\boldsymbol{\theta}_j, \mathbf{u})}\right|.$$

When $j' = j$, the move can be either a standard Metropolis-Hastings or Gibbs step. Posterior model probabilities and Bayes factors may be estimated from the output of this algorithm as in the previous subsection.

The "dimension matching" aspect of this algorithm (step 4 above) is a bit obscure and merits further discussion. Suppose we are comparing two models, for which $\theta_1 \in \Re$ and $\theta_2 \in \Re^2$. If θ_1 is a subvector of θ_2, then when moving from $j = 1$ to $j' = 2$, we might simply draw $u \sim q(u)$ and set

$$\theta'_2 = (\theta_1, u)\ .$$

That is, the dimension matching g is the identity function, and so the Jacobian in step 5 is equal to 1. Thus, if we set $h(1,1) = h(1,2) = h(2,1) = h(2,2) = 1/2$, we have

$$\alpha_{1\to 2} = \min\left\{1, \frac{f(\mathbf{y}|\boldsymbol{\theta}_2', M=2)p(\boldsymbol{\theta}_2'|M=2)\pi_2}{f(\mathbf{y}|\boldsymbol{\theta}_1, M=1)p(\boldsymbol{\theta}_1|M=1)\pi_1 q(u)}\right\} ,$$

with a corresponding expression for $\alpha_{2\to 1}$.

In many cases, however, θ_1 will not naturally be thought of as a subvector of θ_2. Green (1995) considers the case of a *changepoint model* (see e.g. Table 3.5 and the associated exercises in Chapter 3), in which the choice is between a time series model having a single, constant mean level θ_1, and one having two levels – say, $\theta_{2,1}$ before the changepoint and $\theta_{2,2}$ afterward. In this setting, when moving from Model 2 to 1, we would not likely want to use *either* $\theta_{2,1}$ or $\theta_{2,2}$ as the proposal value θ_1'. A more plausible choice might be

$$\theta_1' = \frac{\theta_{2,1} + \theta_{2,2}}{2} , \tag{4.30}$$

since the average of the pre- and post-changepoint levels should provide a competitive value for the single level in Model 1. To ensure reversibility of this move, when going from Model 1 to 2 we might sample $u \sim q(u)$ and set

$$\theta_{2,1}' = \theta_1 - u \quad \text{and} \quad \theta_{2,2}' = \theta_1 + u ,$$

since this is a 1-1 and onto function corresponding to the deterministic down move (4.30).

Several variations or simplifications of reversible jump MCMC have been proposed for various model classes; see e.g. Richardson and Green (1997) in the context of mixture modeling, and Knorr-Held and Rasser (2000) for a spatial disease mapping application. Also, the "jump diffusion" approach of Phillips and Smith (1996) can be thought of as a variant on the reversible jump idea.

As with other Metropolis-Hastings algorithms, transformations to various parameters are often helpful in specifying proposal densities in reversible jump algorithms (say, taking the log of a variance parameter). It may also be helpful to apply reversible jump to a somewhat reduced model, where we analytically integrate certain parameters out of the model and use (lower-dimensional) proposal densities for the parameters that remain. We will generally not have closed forms for the full conditional distributions of these "leftover" parameters, but this is not an issue because, unlike the Gibbs-based CC method, reversible jump does not require them. Here an example might be hierarchical normal random effects models with conjugate priors: the random effects (and perhaps even the fixed effects) may be integrated out, permitting the algorithm to sample only the model indicator and the few remaining (variance) parameters.

4.5.4 Using partial analytic structure

Godsill (2001) proposes use of a "composite model space," which is essentially the setting of Carlin and Chib (1995) except that parameters are allowed to be "shared" between different models. A standard Gibbs sampler applied to this composite model produces the CC method, while a more sophisticated Metropolis-Hastings approach produces a version of the reversible jump algorithm that avoids the "dimension matching" step present in its original formulation (see step 4 in Subsection 4.5.3 above). This step is often helpful for challenging problems (e.g., when moving to a model containing many parameters whose values would not plausibly equal any of those in the current model), but may be unnecessary for simpler problems.

Along these lines, Godsill (2001) outlines a reversible jump method that takes advantage of *partial analytic structure* (PAS) in the Bayesian model. This procedure is applicable when there exists a subvector $(\boldsymbol{\theta}_{j'})_{\mathcal{U}}$ of the parameter vector $\boldsymbol{\theta}_{j'}$ for model j' such that $p((\boldsymbol{\theta}_{j'})_{\mathcal{U}}|(\boldsymbol{\theta}_{j'})_{-\mathcal{U}}, M = j', \mathbf{y})$ is available in closed form, and in the current model j, there exists an equivalent subvector $(\boldsymbol{\theta}_j)_{-\mathcal{U}}$ (the elements of $\boldsymbol{\theta}_j$ *not* in subvector $\mathcal{U}$) of the same dimension as $(\boldsymbol{\theta}_{j'})_{-\mathcal{U}}$. Operationally,

1. Let the current state be $(j, \boldsymbol{\theta}_j)$, where $\boldsymbol{\theta}_j$ is of dimension n_j.
2. Propose a new model j' with probability $h(j, j')$.
3. Set $(\boldsymbol{\theta}_{j'})_{-\mathcal{U}} = (\boldsymbol{\theta}_j)_{-\mathcal{U}}$.
4. Accept the proposed move with probability
$$\alpha_{j \to j'} = \min\left\{1,\ \frac{p(j'|(\boldsymbol{\theta}_{j'})_{-\mathcal{U}}, \mathbf{y})h(j', j)}{p(j|(\boldsymbol{\theta}_j)_{-\mathcal{U}}, \mathbf{y})h(j, j')}\right\}, \tag{4.31}$$
where $p(j|(\boldsymbol{\theta}_j)_{-\mathcal{U}}, \mathbf{y}) = \int p(j, (\boldsymbol{\theta}_j)_{\mathcal{U}}|(\boldsymbol{\theta}_j)_{-\mathcal{U}}, \mathbf{y})\ d(\boldsymbol{\theta}_j)_{\mathcal{U}}$.
5. If the model move is accepted, update the parameters of the new model $(\boldsymbol{\theta}_{j'})_{\mathcal{U}}$ and $(\boldsymbol{\theta}_{j'})_{-\mathcal{U}}$ using standard Gibbs or Metropolis-Hastings steps; otherwise, update the parameters of the old model $(\boldsymbol{\theta}_j)_{\mathcal{U}}$ and $(\boldsymbol{\theta}_j)_{-\mathcal{U}}$ using standard Gibbs or Metropolis-Hastings steps.

Note that model move proposals of the form $j \to j$ always have acceptance probability 1, and therefore when the current model is proposed, this algorithm simplifies to standard Gibbs or Metropolis-Hastings steps. Note that multiple proposal densities may be needed for $(\boldsymbol{\theta}_j)_{\mathcal{U}}$ across models since, while this parameter is common to all of them, its interpretation and posterior support may differ. Troughton and Godsill (1998) give an example of this algorithm, in which the update step of $(\boldsymbol{\theta}_j)_{\mathcal{U}}$ is skipped when a proposed model move is rejected.

Summary and recommendations

Han and Carlin (2001) review and compare many of the methods described in this and the previous subsections in the context of two examples, the first

a simple regression example, and the second a more challenging hierarchical longitudinal model (see Section 7.3). The methods described in this section that sample jointly over model and parameter space (such as product space search and reversible jump) often converge very slowly, due to the difficulty in finding suitable pseudoprior (or proposal) densities. Marginalizing random effects out of the model can be helpful in this regard, but this tends to create new problems: the marginalization often leads to a very complicated form for the model switch acceptance ratio (step 5 in Subsection 4.5.3), thus increasing the chance of an algebraic or computational error. Even with the marginalization, such methods remain difficult to tune. The user will often need a rough idea of the posterior model probabilities $p(M = j|\mathbf{y})$ in order to set the prior model probabilities π_j in such a way that the sampler spends roughly equal time visiting each of the candidate models. Preliminary model-specific runs are also typically required to specify proposal (or pseudoprior) densities for each model.

By contrast, the marginal likelihood methods of Section 4.4 appear relatively easy to program and tune. These methods do not require preliminary runs (only a point of high posterior density, $\boldsymbol{\theta}'$), and in the case of the Gibbs sampler, only a rearrangement of existing computer code. Estimating standard errors is more problematic (the authors' suggested approach involves a spectral density estimate and the delta method), but simply replicating the entire procedure a few times with different random number seeds generally provides an acceptable idea of the procedure's order of accuracy.

In their numerical illustrations, Han and Carlin (2001) found that the RJ and PAS methods ran more quickly than the other model space search methods, but the marginal likelihood methods seemed to produce the highest degree of accuracy for roughly comparable runtimes. This is in keeping with the intuition that some gain in precision should accrue to MCMC methods that avoid a model space search.

As such, we recommend the marginal likelihood methods as relatively easy and safe approaches when choosing among a collection of standard (e.g., hierarchical linear) models. We hasten to add, however, that the blocking required by these methods may preclude their use in some settings, such as spatial models using Markov random field priors (which involve large numbers of random effect parameters that cannot be analytically integrated out of the likelihood nor readily updated in blocks; see Subsection 7.7.2). In such cases, reversible jump may offer the only feasible alternative for estimating a Bayes factor. The marginal likelihood methods would also seem impractical if the number of candidate models were very large (e.g., in variable selection problems having 2^p possible models, corresponding to each of p predictors being either included or excluded). But, as alluded to earlier, we caution that the ability of joint model and parameter space search methods to sample effectively over such large spaces is very much in doubt; see for example Clyde et al. (1996).

4.6 Other model selection methods

The Bayes factor estimation methods discussed in the previous two sections require substantial time and effort (both human and computer) for a rather modest payoff, namely, a collection of posterior model probability estimates (possibly augmented with associated standard error estimates). Besides being mere single number summaries of relative model worth, Bayes factors are not interpretable with improper priors on any components of the parameter vector, because if the prior distribution is improper, then the marginal distribution of the data necessarily is as well. Even for proper priors, the Bayes factor has been criticized on theoretical grounds; see for example Gelfand and Dey (1994) and Draper (1995). One might conclude that none of the methods considered thus far is appropriate for everyday, "rough and ready" model comparison, and instead search for more computationally realistic alternatives.

One such alternative might be more informal, perhaps graphical methods for model selection. These could be based on the marginal distributions $m(y_i)$, $i = 1, \ldots, n$ (as in Berger 1985, p. 199), provided they exist. Alternatively, we could use the conditional predictive distributions $f(y_i|\mathbf{y}_{(i)})$, $i = 1, \ldots, n$ (as in Gelfand et al., 1992), since they will be proper when $p(\boldsymbol{\theta}|\mathbf{y}_{(i)})$ is, and they indicate the likelihood of each datapoint in the presence of all the rest. For example, the product (or the sum of the logs) of the observed conditional predictive ordinate (CPO) values $f(y_i^{obs}|\mathbf{y}_{(i)})$ given in (2.29) could be compared across models, with the larger result indicating the preferred model. Alternatively, sums of the squares or absolute values of the standardized residuals d_i given in (2.28) could be compared across models, with the model having the smaller value now being preferred. We could improve this procedure by accounting for differing model size – say, by plotting the numerator of (2.28) versus its denominator for a variety of candidate models on the same set of axes. In this way we could judge the accuracy of each model relative to its precision. For example, an overfitted model (i.e., one with redundant predictor variables) would tend to have residuals of roughly the same size as an adequate one, but with higher variances (due to the "collinearity" of the predictors). We exemplify this conditional predictive approach in our Section 8.1 case study.

4.6.1 Penalized likelihood criteria: AIC, BIC, and DIC

For some advanced models, even cross-validatory predictive selection methods may be unavailable. For example, in the spatial models of Subsection 7.7.2, the presence of certain model parameters identified only by the prior leads to an information deficit that causes the conditional predictive distributions (2.29) to be improper. In many cases, informal likelihood or penalized likelihood criteria may offer a feasible alternative. Log-

likelihood summaries are easy to estimate using posterior samples $\{\boldsymbol{\theta}^{(g)}, g = 1, \ldots, G\}$, since we may think of $\ell \equiv \log L(\boldsymbol{\theta})$ as a parametric function of interest, and subsequently compute

$$\hat{\ell} \equiv E[\log L(\boldsymbol{\theta})|\mathbf{y}] \approx \frac{1}{G}\sum_{g=1}^{G} \log L(\boldsymbol{\theta}^{(g)}) \tag{4.32}$$

as an overall measure of model fit to be compared across models. To account for differing model size, we could penalize $\hat{\ell}$ using the same sort of penalties as the Bayesian information (Schwarz) criterion (2.20) or the Akaike information criterion (2.21). In the case of the former, for example, we would have

$$\widehat{BIC} = 2\hat{\ell} - p \log n \ ,$$

where as usual p is the number of parameters in the model, and n is the number of datapoints.

Unfortunately, a problem arises here for the case of hierarchical models: what exactly are p and n? For example, in a longitudinal setting (see Section 7.3) in which we have s_i observations on patient i, $i = 1, \ldots, m$, shall we set $n = \sum_i s_i$ (the total number of observations), or $n = m$ (the number of patients)? If the observations on every patient were independent, the former choice would seem most appropriate, while if they were perfectly correlated within each patient, we might instead choose the latter. But of course the true state of nature is likely somewhere in between. Similarly, if we had a collection of m random effects, one for each patient, what does this contribute to p? If the random effects had nothing in common (i.e., they were essentially like fixed effects), they would contribute the full m parameters to p, but if the data (or prior) indicated they were all essentially identical, they would contribute little more than one "effective parameter" to the total model size p. Pauler (1998) obtained results in the case of hierarchical normal linear models, but results for general models remain elusive. In particular, Volinsky and Raftery (2000) showed that *either* definition above (m or $\sum_i s_i$) can be justified asymptotically in the case of survival models with censored data.

In part to help address this problem, Spiegelhalter et al. (2002) suggested the *Deviance Information Criterion* (DIC), a generalization of the Akaike information criterion (AIC) that is based on the posterior distribution of the deviance statistic (2.24). As introduced in Subsection 2.4.2, the DIC approach captures the *fit* of a model by the posterior expectation of the deviance, $\overline{D} = E_{\theta|y}[D]$, and the *complexity* of a model by the effective number of parameters p_D, defined in equation (2.25) as

$$p_D = \overline{D} - D(\overline{\boldsymbol{\theta}}) \ .$$

The DIC is then defined as in equation (2.26) analogous to AIC as

$$DIC = \overline{D} + p_D \ .$$

Because we desire models that exhibit good fit but also a reasonable degree of parsimony, smaller values of DIC indicate preferred models. As with other penalized likelihood criteria, DIC is not intended for identification of the "correct" model, but rather merely as a method of comparing a collection of alternative formulations (all of which may be incorrect).

An asymptotic justification of DIC is straightforward in cases where the number of observations n grows with respect to the number of parameters p, and where the prior $p(\theta)$ is non-hierarchical and completely specified (i.e., having no unknown parameters). Here we may expand $D(\boldsymbol{\theta})$ around $\bar{\boldsymbol{\theta}}$ to give, to second order,

$$D(\boldsymbol{\theta}) \approx D(\bar{\boldsymbol{\theta}}) - 2(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})^T L' - (\boldsymbol{\theta} - \bar{\boldsymbol{\theta}})^T L''(\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}) \tag{4.33}$$

where $L = \log p(\mathbf{y}|\boldsymbol{\theta}) = -D(\boldsymbol{\theta})/2$, and L' and L'' are the first derivative vector and second derivative matrix with respect to $\boldsymbol{\theta}$. However, from the Bayesian Central Limit Theorem (Theorem 3.1) we have that $\boldsymbol{\theta} \mid \mathbf{y}$ is approximately distributed as $N\left(\hat{\boldsymbol{\theta}}, [-L'']^{-1}\right)$, where $\bar{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}}$ are the maximum likelihood estimates such that $L' = 0$. This in turn implies that $(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T(-L'')(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})$ has an approximate chi-squared distribution with p degrees of freedom. Thus, writing $D_{\text{non}}(\boldsymbol{\theta})$ to represent the deviance for a non-hierarchical model, from (4.33) we have that

$$D_{\text{non}}(\boldsymbol{\theta}) \;\approx\; D(\hat{\boldsymbol{\theta}}) - (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T L''(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) \;=\; D(\hat{\boldsymbol{\theta}}) + \chi^2_p \,.$$

Rearranging this expression and taking expectations with respect to the posterior distribution of $\boldsymbol{\theta}$, we have

$$p \;\approx\; E_{\theta|y}[D_{\text{non}}(\boldsymbol{\theta})] - D(\hat{\boldsymbol{\theta}}) \,, \tag{4.34}$$

so that the number of parameters is approximately the expected deviance $\overline{D} = E_{\theta|y}[D_{\text{non}}(\boldsymbol{\theta})]$ minus the fitted deviance. But since $AIC = D(\hat{\boldsymbol{\theta}}) + 2p$, from (4.34) we obtain $AIC \approx \overline{D} + p$, the expected deviance plus the number of parameters. The DIC approach for hierarchical models thus follows this equation and equation (4.34), but substituting the posterior mean $\overline{\theta}$ for the maximum likelihood estimate $\hat{\theta}$. It is a generalization of Akaike's criterion, because for non-hierarchical models, $\overline{\boldsymbol{\theta}} \approx \hat{\boldsymbol{\theta}}$, $p_D \approx p$, and DIC $\approx$ AIC. DIC can also be shown to have much in common with the hierarchical model selection tools previously suggested by Ye (1998) and Hodges and Sargent (2001), though the DIC idea applies much more generally.

As with all penalized likelihood criteria, DIC consists of two terms, one representing "goodness of fit" and the other a penalty for increasing model complexity. As mentioned in Subsection 2.4.2, DIC is scale-free, so as with AIC and BIC, only *differences* in DIC across models are meaningful. (Of course, p_D *does* have a scale, namely, the size of the effective parameter space.) DIC is also a very general tool, and can be readily calculated for each model being considered without analytic adaptation, complicated loss

functions, additional MCMC sampling (say, of predictive values), or any matrix inversion.

The general applicability, attractive interpretations, and, perhaps most importantly, ready availability of p_D and DIC within the `WinBUGS` package led to their widespread use by data analysts even before the publication of the original paper by Spiegelhalter et al. (2002); see e.g. Erkanli et al. (1999). Still, many practical issues have led some to question the appropriateness of DIC for arbitrarily general Bayesian models. For example, DIC is not invariant to parametrization, so (as with prior elicitation) the most plausible parametrization must be carefully chosen beforehand. Unknown scale parameters and other innocuous restructuring of the model can also lead to small changes in the computed DIC value. Determining an appropriate variance estimate for DIC also remains a vexing practical problem. Zhu and Carlin (2000) experiment with various delta method approaches to this problem in the context of spatio-temporal models of the sort considered in Section 7.7.2, but conclude that a "brute force" replication approach may be the only suitably accurate method in such complicated settings. That is, we would independently replicate the calculation of DIC a large number of times N, obtaining a sequence of DIC estimates $\{DIC_l,\ l = 1, \ldots, N\}$, and estimate $Var(DIC)$ by the sample variance,

$$\widehat{Var}(DIC) = \frac{1}{N-1}\sum_{l=1}^{N}(DIC_l - \overline{DIC})^2 \ .$$

Finally, and perhaps most embarrassingly, p_D can occasionally emerge as *negative*, even though clearly an "effective model size" should be between 0 and the sheer number of parameters in the model. This situation is rare in standard model classes, but does arise in certain situations where the joint posterior departs markedly from normality, thus violating the terms of the asymptotic argument presented above. For example, negative p_Ds have been observed with non-log-concave likelihoods in the presence of substantial conflict between prior and data, when the posterior distribution for a parameter is extremely asymmetric (or symmetric but bimodal), and in situations where the posterior mean is a very poor summary statistic and thus leads to an unreasonably large deviance estimate $D(\bar{\boldsymbol{\theta}})$. The following example provides an illustration of this problem.

Example 4.8 Consider a DIC analysis of the three error models for Fisher's sleep data compared in Example 4.7, namely the normal, the t_2, and the DE. Recall in that example we fit the latter two distributions as normal scale mixtures, but `WinBUGS` will also allow them to be fit directly by making two more copies of the data vector (say, `Y4` and `Y5`) and augmenting our previous code with

BUGS code
```
Y4[i] ~ dt(theta[4], tau0[4], 2)
Y5[i] ~ ddexp(theta[5], tau0[5])
```

Model	$\overline{D}$	$\hat{D}$	p_D	DIC
1 (normal)	33.99	32.24	1.75	35.73
2 (t_2, normal scale mixture)	24.54	20.65	3.89	28.44
3 (DE, normal scale mixture)	25.15	27.10	–1.95	23.20
4 (t_2, direct fit)	30.19	28.50	1.69	31.88
5 (DE, direct fit)	30.05	27.85	2.20	32.25

Table 4.3 *Comparison of DIC and related statistics for three error distributions, Fisher's sleep data.*

and then adding priors for the requisite new parameters simply by increasing the upper bound in the code's last `for` loop, i.e.,

BUGS code

```
for (k in 1:5) {
  theta[k] ~ dnorm(0,0.00001)  # Vague normal on mean
  sigma[k] ~ dunif(0,100)      # Uniform on sigma
  tau0[k] <- 1/(sigma[k]*sigma[k])
}
```

Table 4.3 shows the resulting fit ($\overline{D}$), complexity (p_D), and overall model choice (DIC) scores for the five models. The t_2 and DE models emerge as DIC-better than the normal, regardless of whether they are fit directly or as a normal scale mixture. In the t_2 case, the results for p_D suggest that the scale mixing parameters contribute only an extra 2 to 2.5 effective parameters (Model 2 versus Model 4). The smaller DIC value obtained in the scale mixing case implies that the mixing parameters are worthwhile in an overall model selection sense. The DIC scores for these two models need not be equal since, although they lead to the same marginal likelihood for the Y_i, the mixing prior across the λ_i creates a different *joint* distribution with a different effective dimension.

Turning to the DE errors, the p_D estimate is actually *negative* in the scale mixing case, the nonsensical result alluded to above. The non-log-concavity of the DE model seems to be causing some problems here. Specifically, it appears that the posterior mean is a poor summary statistic for these very asymmetric posteriors, and the resulting very large deviance has led to the nonsensical negative p_D estimate. As such, the very low DIC value obtained for this model (Model 3) is probably not to be trusted, since it benefits from the unrealistically negative p_D score. ■

As of the present writing, the use of p_D and DIC remains common, though with a general caveat that they should be avoided for models lying far outside the exponential family, where it is not at all clear that a normal approximation to the posterior is sensible. Moreover, Celeux et al. (2006) observe that in missing data settings, multiple DICs can be defined,

depending on whether one integrates out the missing data or treats it as something to be estimated along with the parameters $\boldsymbol{\theta}$. In advanced mixture modeling settings, these authors find none of the resulting DICs to emerge as obviously superior in all cases. See van der Linde (2004, 2005) for further discussion on the general applicability of DIC and related theoretical support. See also Hodges and Sargent (2001) and Lu et al. (2007) for an alternate method of counting parameters using the constraint case formulation in equation (3.23).

4.6.2 Predictive model selection

Besides being rather ad hoc, a problem with penalized likelihood approaches is that the usual choices of penalty function are motivated by asymptotic arguments that are sometimes hard to justify in practice. While p_D does provide one definition of the "effective size" of hierarchical models, it is difficult to justify in many nonparametric or other highly parametrized settings (e.g., Cox-type survival models).

Suppose we return to the cross-validatory approach initially presented in Subsection 2.5.1, and mimic the basic log-likelihood calculation in (4.32). That is, if we think of the product of all n of the CPO values $f(y_i|\mathbf{y}_{(i)})$ given in (2.29) as a "psuedo marginal likelihood," this gives a cross-validatory summary measure of fit. The *log pseudo marginal likelihood* (LPML), originally suggested by Geisser and Eddy (1979), is simply the log of this measure,

$$\text{LPML} = \log\left\{\prod_{i=1}^{n} f(y_i|\mathbf{y}_{(i)})\right\} = \sum_{i=1}^{n} \log f(y_i|\mathbf{y}_{(i)}) \;, \tag{4.35}$$

the log being added primarily for computational convenience. LPML is sometimes used in place of $\overline{D}$ or DIC; for example, Draper and Krnjajic (2007, Sec. 4.1) have shown that DIC approximates the LPML for approximately Gaussian posteriors. Unlike Bayes factors, the LPML remains well defined under improper priors (provided the posterior does), and is quite stable computationally. Ibrahim, Chen, and Sinha (2001) offer a detailed discussion of the use of CPO and LPML with survival data; see also Gelfand and Mallick (1995), Sinha and Dey (1997), and Zhao et al. (2006).

As an alternative, we may work with the full posterior predictive distribution (i.e., conditional on all the observed data $\mathbf{y}$), as is done in Bayesian p-value calculations like (2.32). Following Laud and Ibrahim (1995), intuitively appealing model complexity penalty terms can emerge without resorting to complex asymptotics. As in the corresponding part of Subsection 2.5.1, the basic distribution we work with is

$$f(\mathbf{y}_{new}|\mathbf{y}_{obs}) = \int f(\mathbf{y}_{new}|\boldsymbol{\theta})\, p(\boldsymbol{\theta}|\mathbf{y}_{obs})\, d\boldsymbol{\theta} \;, \tag{4.36}$$

where $\boldsymbol{\theta}$ denotes the collection of model parameters, and $\mathbf{y}_{new}$ is viewed as a replicate of the observed data vector $\mathbf{y}_{obs}$. The model selection criterion first selects a discrepancy function $d(\mathbf{y}_{new}, \mathbf{y}_{obs})$, then computes

$$E[d(\mathbf{y}_{new}, \mathbf{y}_{obs})|\mathbf{y}_{obs}, M_i] \ , \tag{4.37}$$

and selects the model which minimizes (4.37). For Gaussian likelihoods, Laud and Ibrahim (1995, p. 250) suggested

$$d(\mathbf{y}_{new}, \mathbf{y}_{obs}) = (\mathbf{y}_{new} - \mathbf{y}_{obs})^T(\mathbf{y}_{new} - \mathbf{y}_{obs}) \ . \tag{4.38}$$

For a non-Gaussian generalized linear mixed model, we may prefer to replace (4.38) by the corresponding deviance criterion. For example, with a Poisson likelihood, we would set

$$d(\mathbf{y}_{new}, \mathbf{y}_{obs}) = 2\sum_l \{y_{l,obs} \log(y_{l,obs}/y_{l,new}) - (y_{l,obs} - y_{l,new})\} \ , \tag{4.39}$$

where l indexes the components of $\mathbf{y}$. Routine calculation shows that the l^{th} term in the summation in (4.39) is strictly convex in $y_{l,new}$ if $y_{l,obs} > 0$. To avoid problems with extreme values in the sample space, we replace (4.39) by

$$\begin{aligned} &\tilde{d}(\mathbf{y}_{new}, \mathbf{y}_{obs}) \\ &\quad = 2\textstyle\sum_l \left\{ \left(y_{l,obs} + \frac{1}{2}\right) \log\left(\frac{y_{l,obs}+\frac{1}{2}}{y_{l,new}+\frac{1}{2}}\right) - (y_{l,obs} - y_{l,new}) \right\} \ . \end{aligned}$$

Suppose we write

$$\begin{aligned} E[\tilde{d}(\mathbf{y}_{new}, \mathbf{y}_{obs}) \mid \mathbf{y}_{obs}, M_i] \ &= \ \tilde{d}(E[\mathbf{y}_{new}|\mathbf{y}_{obs}, M_i], \mathbf{y}_{obs}) \\ &\quad + E[\tilde{d}(\mathbf{y}_{new}, \mathbf{y}_{obs}) \mid \mathbf{y}_{obs}, M_i] - \tilde{d}(E[\mathbf{y}_{new}|\mathbf{y}_{obs}, M_i], \mathbf{y}_{obs}) \ . \end{aligned} \tag{4.40}$$

Intuitive interpretations may be given to the terms in (4.40). The left-hand side is the expected predictive deviance (EPD) for model M_i. The first term on the right-hand side is essentially the likelihood ratio statistic with the MLE for $E(\mathbf{y}_{new}|\boldsymbol{\theta}_i, M_i)$, which need not exist, replaced by $E(\mathbf{y}_{new}|\mathbf{y}_{obs}, M_i)$. Jensen's inequality shows that the second term minus the third is strictly positive, and is the penalty associated with M_i. This difference becomes

$$\begin{aligned} &2\textstyle\sum_l \left(y_{l,obs} + \frac{1}{2}\right) \\ &\quad \times \left\{ \log E\left[y_{l,new} + \tfrac{1}{2} \,\middle|\, \mathbf{y}_{obs}\right] - E\left[\log\left(y_{l,new} + \tfrac{1}{2}\right) \,\middle|\, \mathbf{y}_{obs}\right] \right\} \ . \end{aligned} \tag{4.41}$$

Again, each term in the summation is positive. A second order Taylor series expansion shows that (4.41) is approximately

$$\sum_l \frac{y_{l,obs} + \frac{1}{2}}{[E(y_{l,new} + \frac{1}{2} \mid \mathbf{y}_{obs})]^2} \cdot Var(y_{l,new} \mid \mathbf{y}_{obs}) \ . \tag{4.42}$$

Hence (4.41) can be viewed as a weighted predictive variability penalty, a natural choice in that if M_i is too large (i.e., contains too many explanatory

terms resulting in substantial multicollinearity), predictive variances will increase. Lastly, (4.42) is approximately

$$E\left\{\sum_l \frac{(y_{l,new} - E(y_{l,new}|\mathbf{y}_{obs}))^2}{E(y_{l,new} + \frac{1}{2} \mid \mathbf{y}_{obs})} \,\middle|\, \mathbf{y}_{obs}\right\}$$

and thus may be viewed as a predictive *corrected* goodness-of-fit statistic.

Computation of (4.40) requires calculation of $E(y_{l,new}|\mathbf{y}_{obs}, M_i)$ as well as $E[\log(y_{l,new} + \frac{1}{2})|\mathbf{y}_{obs}, M_i]$. Such predictive expectations are routinely obtained as Monte Carlo integrations.

Finally, we remark that Gelfand and Ghosh (1998) extend the above approach to a completely formal decision-theoretic setting. Their method seeks to choose the model minimizing a particular posterior predictive loss. Like DIC and the less formal method above, the resulting criterion consists of a (possibly weighted) sum of a goodness-of-fit term and a model complexity penalty term.

4.7 Exercises

1. Steensma et al. (2005) presented the data in Table 4.4, from a randomized controlled trial comparing two dosing schedules for the drug erythropoiten. Serum hemoglobin (HGB, in g/dL) was recorded for $N = 365$ cancer patients with anemia over $T = 22$ weeks. The full data file is available at `www.biostat.umn.edu/~brad/data/HGB_data.txt`. We wish to fit a hierarchical simple linear regression model of the form

$$Y_{ij} = \beta_{0i} + \beta_{1i}(X_j - \mu_X) + \epsilon_{ij},\ i = 1, \ldots, 365,\ j = 1, \ldots, 22, \quad (4.43)$$

where $X_j = j$, the week index, $\epsilon_{ij} \overset{iid}{\sim} N(0, \tau)$, $\beta_{0i} \overset{iid}{\sim} N(\mu_0, \tau_0)$, and $\beta_{1i} \overset{iid}{\sim} N(\mu_1, \tau_1)$. That is, as in Example 2.13 (and Example 7.2 in Chapter 7), we allow each subject's HGB trajectory to have its own slope and intercept, but borrow strength from the ensemble by treating these as normal random effects. Note that we also center the week index around its own mean, $\mu_X = 11.5$.

   (a) Use `WinBUGS` to fit the above model, assuming vague priors for the hyperparameters τ, μ_0, τ_0, μ_1, and τ_1. Run multiple chains to assess convergence, and interpret the resulting posterior distributions for the grand slope μ_1 and the individual-specific slopes β_{1i}. How do the interpretations of these parameters differ? Use the `compare` function in `WinBUGS` to examine these for all participants. Are all participants' HGB measurements improving over time?

   (b) Note that many of the Y_{ij} are missing; under the assumption that they are missing at random, `WinBUGS` can impute them according to the fitted model. Monitor the hemoglobin values estimated for participant

Patient	HGB Measurements by Week j, $j = 1, \ldots, 22$					
i	$Y_{i,1}$	$Y_{i,2}$	$Y_{i,3}$	$\ldots$	$Y_{i,21}$	$Y_{i,22}$
1	10.2	9.8	10.3	$\cdots$	NA	11.4
2	10.1	9.4	9.1	$\cdots$	NA	11.6
3	9.9	10	10.2	$\cdots$	NA	12.4
4	10.7	9.7	11.4	$\cdots$	NA	NA
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
365	10.7	10.5	10.4	$\cdots$	14.2	NA

Table 4.4 $T = 22$ *weekly hemoglobin (HGB) measurements.*

10, who is missing data from weeks 5 and 16 through 22. Compare the standard deviation of the estimate at week 5 to those at the end of the study, and explain any differences.

(c) As you've already noticed, the `WinBUGS` data file actually contains information on one more variable, `newarm`, a binary variable indicating which dosing schedule (or *treatment arm*, 1 or 2) was used for each patient. Modify model (4.43) above to allow different grand intercepts and/or slopes for the two treatment arms. Discuss any resulting problems with MCMC convergence, how they might arise, and how they can be handled. Are the two groups significantly different in any respect? Draw a quick plot in `R` comparing the fitted grand means in the two groups. Would a DIC comparison of this "full" model and the "reduced" model in part (a) be sensible here?

2. Refer to the data in Table 2.3, also available on the web in `WinBUGS` format at `http://www.biostat.umn.edu/~brad/data.html`. These pharmacokinetic (PK) data of Wakefield et al. (1994) were initially presented in Example 2.13, and analyzed there using an approximate linear hierarchical model. Recall these are the plasma concentrations Y_{ij} of the drug cadralazine at up to $T = 8$ different time lags x_{ij} following the administration of a single dose of 30 mg in $N = 10$ cardiac failure patients. Here, $i = 1, \ldots, 10$ indexes the patient, while $j = 1, \ldots, n_i$ indexes the observations, where $5 \le n_i \le 8$. Wakefield et al. suggest a "one-compartment" nonlinear pharmacokinetic model wherein the mean plasma concentration $\eta_{ij}(x_{ij})$ at time x_{ij} is given by

$$\eta_{ij}(x_{ij}) = 30\alpha_i^{-1} \exp(-\beta_i x_{ij}/\alpha_i) \; .$$

Subsequent unpublished work by these same authors suggests this model is best fit on the log scale. That is, we suppose

$$Z_{ij} \equiv \log Y_{ij} = \log \eta_{ij}(x_{ij}) + \epsilon_{ij} \; ,$$

where $\epsilon_{ij} \overset{ind}{\sim} N(0, \tau_i)$. The mean structure for the Z_{ij}'s thus emerges as

$$\begin{aligned} \log \eta_{ij}(x_{ij}) &= \log \left[30\alpha_i^{-1} \exp(-\beta_i x_{ij}/\alpha_i)\right] \\ &= \log 30 - \log \alpha_i - \beta_i x_{ij}/\alpha_i \\ &= \log 30 - a_i - \exp(b_i - a_i) x_{ij} \ , \end{aligned}$$

where $a_i = \log \alpha_i$ and $b_i = \log \beta_i$.

(a) Assuming the two sets of random effects are independently distributed as $a_i \overset{iid}{\sim} N(\mu_a, \tau_a)$ and $b_i \overset{iid}{\sim} N(\mu_b, \tau_b)$, use `WinBUGS` or `BRugs` to analyze these data. You may adopt a vague prior structure, though the original authors suggest moderately informative independent $G(1, 0.04)$ priors for τ_a and τ_b.

Note that the full conditional distributions of the random effects are not simple conjugate forms nor guaranteed to be log-concave, so `BUGS`' Metropolis capability is required. Besides the usual posterior summaries and convergence checks, investigate the acceptance rate of the Metropolis algorithm, and the predictive distribution of $Y_{2,7}$ and $Y_{2,8}$, the missing observations on outlying patient 2. How do your results compare to those in Figure 2.22? You may also want to compare model fit, complexity, and overall quality via $\overline{D}, p_D$, and DIC.

(b) The assumption that the random effects a_i and b_i are independent within individuals was probably unrealistic. Instead, follow the original analysis of Wakefield et al. (1994), as well as that in Example 2.13, and assume the $\boldsymbol{\theta}_i \equiv (a_i, b_i)'$ are i.i.d. from a $N_2(\boldsymbol{\mu}, \Omega)$ distribution, where $\boldsymbol{\mu} = (\mu_a, \mu_b)$. Again adopt a corresponding vague conjugate prior specification, namely $\boldsymbol{\mu} \sim N_2(\mathbf{0}, C)$ and $\Omega \sim \text{Wishart}((\rho R)^{-1}, \rho)$ with $C \approx \mathbf{0}$, $\rho = 2$, and $R = Diag(0.01, 0.01)$.

Describe the changes (if any) in your answers from those in part (a).

3. Consider again the binary dugong modeling of Example 4.4. Suppose we wished to obtain side-by-side boxplots of the posteriors of the effect of log-age, β_1, across all three link functions (logit, probit, and cloglog). Modify the `WinBUGS` code given in that example so that the comparison tool on the `Inference` menu can be used to obtain this display.

(*Hint:* Use three copies of the dataset, one for each link function. This will also enable computation of three separate DIC p_D scores, since the DIC tool in `WinBUGS` will then decompose DIC by the three observation variable names.)

4. Spiegelhalter et al. (1995b) analyze the flour beetle mortality data in Table 3.3 using `WinBUGS`. These authors use only the usual, two-parameter parametrization for $p_i \equiv P(death|w_i)$, but compare the logit, probit, and complementary log-log link functions using the centered covariate $z_i = w_i - \bar{w}$.

Location	Y	X
1	0	1.9823
2	0	2.8549
3	0	2.7966
...	...	...
601	0	2.8189
602	1	2.0212
603	1	1.3979

Table 4.5 *Forestry co-presence data.*

(a) The full conditional distributions for α and β have no closed form, but `WinBUGS` does recognize them as being log-concave, and thus capable of being sampled using the Gilks and Wild (1992) adaptive rejection algorithm. Prove this log-concavity under the logit model.

(b) Following the model of Example 4.4, actually carry out the data analysis in `WinBUGS` (the program above, along with properly formatted data and initial value lists, are included in the "examples" section of the help materials). Do the estimated posteriors for the dosage effect β substantially differ for different link functions?

5. Consider `www.biostat.umn.edu/~brad/data/copresence_data.txt`, a data set for which a few records are shown in Table 4.5. Here, Y is a binary variable indicating co-presence of two species in a particular forest at $n = 603$ sampled locations. The lone predictor variable, X, is the log of the distance of each location to the forest edge. Suppose we use a logistic model for p, the probability of co-presence, namely

$$logit(p_i) = \beta_0 + \beta_1 X_i,\ i = 1, \ldots, n.$$

(a) Again following the model of Example 4.4, fit this model in `WinBUGS`, using vague priors. Is proximity to the forest edge a significant predictor of species co-presence?

(b) Fit the same model in `R` using a Metropolis-Hastings algorithm. Report parameter estimates and mean acceptance ratios for each chain. How many samples do you need (both here and in part (a)) to obtain results for β_1 reliable to two digits? Three digits?

(c) Replace the logit link above with the complementary log-log link, $\log[-\log(1 - p_i)]$, and compare the two posteriors for β_1. Also plot the two fitted curves as in Figure 4.6, and compare the models more formally using DIC or some other Bayesian model choice statistic. Does the choice of link function matter much for these data?

6. Consider the data displayed in Table 4.6, originally collected by Treloar (1974) and reproduced in Bates and Watts (1988). These data record the "velocity" y_i of an enzymatic reaction (in counts/min/min) as a function of substrate concentration x_i (in ppm), where the enzyme has been treated with puromycin.

Case (i)	x_i	y_i	Case (i)	x_i	y_i
1	0.02	76	7	0.22	159
2	0.02	47	8	0.22	152
3	0.06	97	9	0.56	191
4	0.06	107	10	0.56	201
5	0.11	123	11	1.10	207
6	0.11	139	12	1.10	200

Table 4.6 *Puromycin experiment data.*

A common model for analyzing biochemical kinetics data of this type is the *Michaelis-Menten* model, wherein we adopt the mean structure

$$\mu_i = \gamma + \alpha x_i/(\theta + x_i) \ ,$$

where $\alpha, \gamma \in \Re$ and $\theta \in \Re^+$. In the nomenclature of Example 4.6, we have design matrix

$$F_\theta^T = \begin{pmatrix} 1 & \cdots & 1 \\ \frac{x_1}{\theta+x_1} & \cdots & \frac{x_n}{\theta+x_n} \end{pmatrix}$$

and $\boldsymbol{\beta}^T = (\gamma, \alpha)$. For this model, obtain estimates of the marginal posterior density of the parameter α, and also the marginal posterior density of the mean velocity at $X = 0.5$, a concentration not represented in the original data, assuming that the error density of the data is

(a) normal
(b) Student's t with 2 degrees of freedom
(c) double exponential.

(*Hint:* The "duplicate the data" trick in Example 4.7 will not work in `WinBUGS` anymore if you handle the missing data prediction aspect in the usual way, i.e., by increasing N from 12 to 13 and adding 0.5 to the X vector and NA to the Y vector; you'll have to think of something else!)

7. For the point null prior partitioning setting described in Subsection 4.2.2, show that requiring $G \in \mathcal{H}_c$ as given in (4.12) is equivalent to requiring $BF \leq \left(\frac{p}{1-p}\right)\left(\frac{1-\pi}{\pi}\right)$, where BF is the Bayes factor in favor of the null hypothesis.

8. For the interval null prior partitioning setting of Subsection 4.2.2, derive an expression for the set of priors $\mathcal{H}_c$ that correspond to rejecting H_0, similar to expression (4.12) for the point null case.
9. Suppose we are estimating a set of residuals r_i using a Monte Carlo approach, as described in Section 4.3. Due to a small sample size n, we are concerned that the approximation

$$E(y_i|\mathbf{y}_{(i)}) = \int E(y_i|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y}_{(i)})d\boldsymbol{\theta} \approx \int E(y_i|\boldsymbol{\theta})p(\boldsymbol{\theta}|\mathbf{y})d\boldsymbol{\theta}$$

will not be accurate. A faculty member suggests the importance sampling estimate

$$\begin{aligned} E(y_i|\mathbf{y}_{(i)}) &= \int E(y_i|\boldsymbol{\theta})\frac{p(\boldsymbol{\theta}|\mathbf{y}_{(i)})}{p(\boldsymbol{\theta}|\mathbf{y})}p(\boldsymbol{\theta}|\mathbf{y})d\boldsymbol{\theta} \\ &\approx \frac{1}{G}\sum_{g=1}^{G} E(y_i|\boldsymbol{\theta}^{(g)})\frac{p(\boldsymbol{\theta}^{(g)}|\mathbf{y}_{(i)})}{p(\boldsymbol{\theta}^{(g)}|\mathbf{y})} \end{aligned}$$

as an alternative. Is this a practical solution? What other approaches might we try? (*Hint:* See equation (3.4).)
10. Refer again to the cross-protocol data and model in Example 2.12.
    (a) Create `WinBUGS` code that will estimate cross validation ("leave one out") residuals and CPO values via importance sampling, as in equation (3.4). Compare your findings to those obtained via the "exact" and "approximate" methods in Example 2.16. Does your importance sampling approximation appear to be sufficiently accurate for the purpose of outlier identification? (*Hint:* Your solution to the previous question may be helpful.)
    (b) Redo the "exact" calculation using the `BRugs` package, following the approach suggested for the stack loss data in Chapter 2, Exercise 22.
11. Suppose we have a convergent MCMC algorithm for drawing samples from $p(\boldsymbol{\theta}|\mathbf{y}) \propto f(\mathbf{y}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$. We wish to locate potential outliers by computing the conditional predictive ordinate $f(y_i|\mathbf{y}_{(i)})$ given in equation (2.29) for each $i = 1, \ldots, n$. Give a computational formula we could use to obtain a simulation-consistent estimate of $f(y_i|\mathbf{y}_{(i)})$. What criterion might we use to classify y_i as a suspected outlier?
12. In the previous problem, suppose we wish to evaluate the model using the model check p'_D given in equation (2.33) with an independent validation data sample $\mathbf{z}$. Give a computational formula we could use to obtain a simulation-consistent estimate of p'_D.
13. Consider again the data in Table 3.3. Define model 1 to be the three variable model given in Example 3.7, and model 2 to be the reduced model having $m_1 = 1$ (i.e., the standard logistic regression model). Compute the Bayes factor in favor of model 1 using

Case (i)	y_i	x_i	z_i	Case (i)	y_i	x_i	z_i
1	3040	29.2	25.4	22	3840	30.7	30.7
2	2470	24.7	22.2	23	3800	32.7	32.6
3	3610	32.3	32.2	24	4600	32.6	32.5
4	3480	31.3	31.0	25	1900	22.1	20.8
5	3810	31.5	30.9	26	2530	25.3	23.1
6	2330	24.5	23.9	27	2920	30.8	29.8
7	1800	19.9	19.2	28	4990	38.9	38.1
8	3110	27.3	27.2	29	1670	22.1	21.3
9	3160	27.1	26.3	30	3310	29.2	28.5
10	2310	24.0	23.9	31	3450	30.1	29.2
11	4360	33.8	33.2	32	3600	31.4	31.4
12	1880	21.5	21.0	33	2850	26.7	25.9
13	3670	32.2	29.0	34	1590	22.1	21.4
14	1740	22.5	22.0	35	3770	30.3	29.8
15	2250	27.5	23.8	36	3850	32.0	30.6
16	2650	25.6	25.3	37	2480	23.2	22.6
17	4970	34.5	34.2	38	3570	30.3	30.3
18	2620	26.2	25.7	39	2620	29.9	23.8
19	2900	26.7	26.4	40	1890	20.8	18.4
20	1670	21.1	20.0	41	3030	33.2	29.4
21	2540	24.1	23.9	42	3030	28.2	28.2

Table 4.7 *Radiata pine compressive strength data.*

(a) the harmonic mean estimator, (4.20)

(b) the importance sampling estimator, (4.21).

Does the second perform better, as expected?

14. Consider the dataset of Williams (1959), displayed in Table 4.7. For $n = 42$ specimens of radiata pine, the maximum compressive strength parallel to the grain y_i was measured, along with the specimen's density, x_i, and its density adjusted for resin content, z_i (resin contributes much to density but little to strength of the wood). For $i = 1, \ldots, n$, we wish to compare the two models $M = 1$ and $M = 2$ where

$$M = 1: \quad y_i = \alpha + \beta(x_i - \bar{x}) + \epsilon_i, \quad \epsilon_i \overset{iid}{\sim} \text{Normal}(0, \sigma^2) \,,$$

and

$$M = 2: \quad y_i = \gamma + \delta(z_i - \bar{z}) + \epsilon_i, \quad \epsilon_i \overset{iid}{\sim} \text{Normal}(0, \tau^2) \,.$$

We desire prior distributions that are roughly centered around the appropriate least squares parameter estimate, but are extremely vague

(though still proper). As such, we place

$$N\left((3000, 185)^T,\ Diag(10^6, 10^4)\right)$$

priors on $(\alpha, \beta)^T$ and $(\gamma, \delta)^T$, and

$$IG\left(3\ ,\ (2 \cdot 300^2)^{-1}\right)$$

priors on σ^2 and τ^2, having both mean and standard deviation equal to 300^2.

Compute the Bayes factor in favor of model 2 (the adjusted density model) using

(a) the marginal density estimation approach of Subsection 4.4.2

(b) the product space search approach of Subsection 4.5.1

(c) the reversible jump approach of Subsection 4.5.3.

How do the methods compare in terms of accuracy? Ease of use? Which would you attempt first in a future problem?

15. In the previous problem, implement both models in `WinBUGS`, and use DIC instead of Bayes factors to choose between them. Is your preference between the two models materially altered by this change? What are the advantages and disadvantages of using DIC instead of a Bayes factor here?

16. Consider again the stack loss data originally presented in Example 2.16. Suppose we wish to compare the assumptions of normal, t_4, and DE errors for these data.

    (a) Repeat the outlier analysis in Example 4.7 for these data. Do the λ_i posteriors for the nonnormal models effectively identify the outliers?

    (b) Use the `Comparison` tool to obtain a side-by-side comparison of the boxplots of the posterior median θ across the three error models, similar to Figure 4.9. Then do a similar comparison of the width of the central 95% credible interval for the errors, $q_{.975}(\sigma) - q_{.025}(\sigma)$, and explain any differences. (*Hint:* Dump the $\sigma^{(g)}$ Gibbs samples into `R` using the `Coda` function in `WinBUGS`, convert them to the quantile difference scale, and then do the boxplot comparison. Use the `qnorm` and `qt` functions in `R`; the DE quantiles can be determined analytically.)

    (c) Repeat the DIC analysis in Example 4.8 for these data. What model is DIC-best? Are negative p_D values again a problem here?