10 Bayesian Statistics

R. Heller · Date: 2026-05-11 · Last changed: 2026-10-30

Posterior inference via `brms` and Stan: priors, hierarchical models, posterior predictive checks, leave-one-out cross-validation, and Bayes factors. The chapter is opinionated: it treats Bayesian inference as a default, not an alternative.

This chapter contains **35 method pages** and **2 labs**. If you are not sure which method to read, return to Chapter 0 and follow the decision tree to the right node.

10.1 Method pages

Method	Source slug
Bayes Factors	`bayes-factors`
Bayes' Theorem for Parameters	`bayes-theorem-for-parameters`
Bayesian ANOVA	`bayesian-anova`
Bayesian Hierarchical Models	`bayesian-hierarchical-intro`
Bayesian Hypothesis Testing	`bayes-hypothesis-testing`
Bayesian Linear Regression	`linear-regression-bayes`
Bayesian Logistic Regression	`logistic-regression-bayes`
Bayesian Mediation Analysis	`bayesian-mediation`
Bayesian Meta-Analysis	`bayesian-meta-analysis`
Bayesian Mixed Models	`bayesian-mixed-models`
brms Basics	`brms-basics`
Conjugate Priors	`conjugate-priors`
Credible Intervals	`credible-intervals`
Divergences and Pairs Plots	`divergences-pairs-plots`
Gibbs Sampling	`gibbs-sampling`
Hamiltonian Monte Carlo	`hamiltonian-monte-carlo`
Highest Posterior Density Interval	`hpd-interval`
Informative Priors	`informative-priors`
Jeffreys' Prior	`jeffreys-prior`
Leave-One-Out Cross-Validation	`loo-cv`
Metropolis-Hastings	`metropolis-hastings`
Posterior Mean and Variance	`posterior-mean-variance`
Posterior Predictive Checks	`posterior-predictive-checks`
Prior Specification	`prior-specification`
R-hat and ESS Diagnostics	`rhat-ess-diagnostics`

Method	Source slug
rstanarm	`rstanarm`
Stan: Introduction	`stan-introduction`
The Beta-Binomial Model	`beta-binomial-model`
The Gamma-Poisson Model	`gamma-poisson-model`
The Normal-Normal Model	`normal-normal-model`
The Posterior Predictive Distribution	`posterior-predictive-distribution`
The Savage-Dickey Density Ratio	`savage-dickey-ratio`
tidybayes Workflow	`tidybayes-workflow`
WAIC	`waic`
Weakly Informative Priors	`weakly-informative-priors`

10.2 Labs

Lab
Bayesian thinking: α control vs decision error
brms / Stan, LOO, hierarchical models

10.3 Bayes Factors

10.3.1 Introduction

A Bayes factor compares two models by the ratio of how plausible the observed data are under each, after integrating over the parameters of each model with respect to its prior. It is the Bayesian counterpart of a likelihood-ratio test, but it carries a richer interpretation: rather than rejecting or failing to reject a null, it returns a continuous strength-of-evidence measure that can equally support the null model when the data are inconsistent with the alternative. This symmetry — quantifying support for either side — is one of the main reasons Bayes factors recur in applied work where “no effect” is itself a substantively interesting conclusion.

10.3.2 Prerequisites

A working understanding of Bayes’ theorem, the role of the marginal likelihood (evidence) in Bayesian inference, and the difference between point-null and composite hypotheses.

10.3.3 Theory

For two models M_0 and M_1 with parameter vectors θ_0 and θ_1 and priors $p(\theta_0 \mid M_0)$ and $p(\theta_1 \mid M_1)$, the Bayes factor in favour of M_1 is

$$\mathrm{BF}_{10} = \frac{p(y \mid M_1)}{p(y \mid M_0)} = \frac{\int p(y \mid \theta_1, M_1) p(\theta_1 \mid M_1)\, \mathrm{d}\theta_1}{\int p(y \mid \theta_0, M_0) p(\theta_0 \mid M_0)\, \mathrm{d}\theta_0}.$$

Posterior odds equal the Bayes factor times the prior odds; under equal prior odds, the posterior probability of M_1 is $\mathrm{BF}_{10}/(\mathrm{BF}_{10}+1)$. Jeffreys' interpretive scale calls $\mathrm{BF} > 3$ substantial, > 10 strong, > 30 very strong, and > 100 decisive evidence; the same thresholds apply, mirrored, for evidence in favour of M_0.

10.3.4 Assumptions

Both models are well-specified and proper priors are assigned to all parameters that differ between the two; improper priors leave the marginal likelihoods undefined up to an arbitrary constant and the Bayes factor inherits that indeterminacy. Numerical estimation must be stable — bridge sampling, Chib's method, or Savage-Dickey ratios for nested models.

10.3.5 R Implementation

```r
library(BayesFactor)

# Simple example: one-sample t-test equivalent
x ← rnorm(40, mean = 0.5, sd = 1)
ttestBF(x)

# Regression
data(mtcars)
full ← lmBF(mpg ~ wt + hp, data = mtcars)
reduced ← lmBF(mpg ~ wt, data = mtcars)
full / reduced
```

10.3.6 Output & Results

The `BayesFactor` package returns a Bayes factor with an estimated error percentage. For regression-type models the output also includes the implied posterior model probabilities; ratios of `BFBayesFactor` objects deliver pairwise comparisons directly.

10.3.7 Interpretation

A reporting sentence: "A default-prior t-test Bayes factor was $\mathrm{BF}_{10} = 4.7$ ($\pm 0.001\%$), interpreted as substantial evidence that the population mean differs from zero." When reporting, always include the prior and (for `BayesFactor`) the prior scale parameter `rscale`; readers cannot otherwise judge how much of the evidence is data-driven versus prior-driven.

10.3.8 Practical Tips

- Wider priors mechanically shift Bayes factors toward the null (Lindley–Bartlett paradox); a sensitivity analysis across at least two prior scales is mandatory for any Bayes-factor conclusion.

- `bridgesampling::bridge_sampler()` works directly with `brms` and Stan fits; it is the most general way to compute marginal likelihoods for non-trivial models.
- For nested null hypotheses, the Savage-Dickey density ratio is faster, more stable, and avoids marginal-likelihood estimation entirely.
- Bayes factors near 1 are genuinely inconclusive; resist the temptation to declare evidence with $\mathrm{BF} = 1.5$ in either direction.
- Bayes factors are not a substitute for predictive performance; pair them with `loo()` when comparing models for prediction, since the two answer different questions.

10.3.9 R Packages Used

`BayesFactor` for default-prior tests and ANOVA / regression Bayes factors, `bridgesampling` for general marginal-likelihood estimation, `bayestestR` for tidy summaries that work with `brms` and `rstanarm` fits, and `polspline` for Savage-Dickey density estimation when the null is nested.

10.3.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.3.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.4 Bayesian Hypothesis Testing

10.4.1 Introduction

Bayesian decision-making about effects splits into two cultural traditions that complement rather than replace each other. The Bayes-factor school computes a ratio of marginal likelihoods and reads strength of evidence from Jeffreys' scale. The Region of Practical Equivalence (ROPE) school, popularised by Kruschke, defines a range of effects that are practically null and asks how much of the posterior lies inside it. Both produce decisions about hypotheses, but they answer slightly different questions: Bayes factors compare models, while ROPE asks whether the data have ruled out a substantively negligible effect.

10.4.2 Prerequisites

A working understanding of posterior distributions, credible intervals, marginal likelihoods, and the difference between statistical significance and practical importance.

10.4.3 Theory

ROPE. Specify, before looking at the data, a parameter range $[a, b]$ inside which an effect is considered practically zero — for example $\beta \in (-0.1, 0.1)$ on a standardised scale. After fitting, compute the proportion of the 95 % HPD (or the full posterior) inside that range. Conventional rules: $> 95\%$ inside means accept the null for practical purposes, 0% inside means reject it, and intermediate values are inconclusive.

Bayes factor. Ratio of marginal likelihoods under two models; for a point null nested in a continuous alternative, the Savage-Dickey density ratio gives the same answer as the ratio of posterior to prior densities at the null. Both quantities depend on the prior under the alternative, so disclosing the prior is part of the report.

10.4.4 Assumptions

ROPE assumes a domain-relevant equivalence margin set in advance; chasing margins after seeing the posterior is a Bayesian variant of p-hacking. Bayes factors assume proper priors and a likelihood that admits stable marginal-likelihood computation.

10.4.5 R Implementation

```r
library(bayestestR); library(brms)

set.seed(2026)
d <- data.frame(y = rnorm(200), x = rnorm(200))
fit <- brm(y ~ x, data = d, chains = 4, iter = 2000, refresh = 0)

rope(fit, range = c(-0.1, 0.1))
bayesfactor_parameters(fit)
describe_posterior(fit)
```

10.4.6 Output & Results

`rope()` returns the percentage of the HPD (or full posterior, depending on the chosen `ci`) inside the equivalence region for each parameter. `bayesfactor_parameters()` returns the Savage-Dickey Bayes factor against the point null. `describe_posterior()` is a one-stop summary that combines the two with point estimates and credible intervals.

10.4.7 Interpretation

A reporting sentence: “The posterior for β_x had 4 % of its 95 % HPD inside the practical-equivalence region $(-0.1, 0.1)$ and a Savage-Dickey Bayes factor of $\mathrm{BF}_{10} = 7.2$ — substantial evidence for a non-null effect on Jeffreys’ scale, with most of the posterior lying outside the practical-null region.” When the two methods disagree (ROPE inconclusive but Bayes factor decisive, or vice versa), report both and explain which question your study was designed to answer.

10.4.8 Practical Tips

- Pre-register the ROPE based on what is practically relevant in your field; for standardised effects, ± 0.1 is a common default but is arbitrary in absolute terms.
- For ROPE on raw scales, work out the smallest effect that would change clinical or business behaviour and use that as the half-width.
- Bayes factors are sensitive to the prior under the alternative (Lindley–Bartlett paradox); always run a sensitivity analysis with at least two prior scales.
- Use `bayestestR::describe_posterior()` to deliver a single block of summaries that includes credible intervals, ROPE, Bayes factors, and probability of direction; consistent reporting reduces ambiguity.
- ROPE is easier to communicate to non-statistical audiences ("most of the posterior is outside the negligible-effect range") than Bayes factors are; choose the framing that fits your reader.

10.4.9 R Packages Used

`bayestestR` for `rope()`, `bayesfactor_parameters()`, and `describe_posterior()`; `brms` or `rstanarm` for the underlying Bayesian fit; `bridgesampling` for marginal-likelihood-based Bayes factors when the Savage-Dickey shortcut does not apply.

10.4.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.4.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.5 Bayes' Theorem for Parameters

10.5.1 Introduction

Bayesian inference is, mechanically, the application of Bayes' theorem to a parameter rather than to an event. The prior encodes whatever was believed before the data arrived, the likelihood is the data-generating model evaluated at the observed data, and the posterior is the updated belief. The conceptual leap from frequentist statistics is small but consequential: parameters are themselves random quantities whose posterior distribution can be summarised, predicted from, and acted upon, without invoking long-run sampling thought experiments.

10.5.2 Prerequisites

A working understanding of Bayes' theorem for events, the concept of a likelihood function, and the difference between probability density and probability mass.

10.5.3 Theory

For a parameter θ and data y, Bayes' theorem reads

$$p(\theta \mid y) = \frac{p(y \mid \theta) p(\theta)}{p(y)} \propto p(y \mid \theta) p(\theta),$$

where the four quantities are the posterior, the likelihood, the prior, and the marginal likelihood $p(y) = \int p(y \mid \theta) p(\theta) \, \mathrm{d}\theta$. The marginal likelihood appears as a normalisation constant from a single-model perspective but plays the central role in Bayes-factor-based model comparison. For closed-form (conjugate) priors, the posterior comes out in the same family as the prior with updated hyperparameters; otherwise MCMC produces draws from the un-normalised right-hand side.

10.5.4 Assumptions

The prior reflects genuine pre-data information or a defensible weak-regularisation choice, the likelihood correctly captures the data-generating process, and posterior integrability is established (proper priors guarantee this; improper priors must be checked).

10.5.5 R Implementation

```
# Beta-Binomial: prior Beta(2, 2), 8 successes in 10 trials
prior_a ← 2; prior_b ← 2
k ← 8; n ← 10
post_a ← prior_a + k
post_b ← prior_b + n - k

theta ← seq(0, 1, length.out = 400)
plot(theta, dbeta(theta, prior_a, prior_b), type = "l",
     main = "Prior → Posterior", ylab = "Density", col = "grey")
lines(theta, dbeta(theta, k + 1, n - k + 1), col = "orange")  # likelihood scaled
lines(theta, dbeta(theta, post_a, post_b), col = "steelblue", lwd = 2)
legend("topleft", c("Prior", "Likelihood", "Posterior"),
       col = c("grey", "orange", "steelblue"), lty = 1)

# Posterior summaries
c(mean = post_a / (post_a + post_b),
  ci95 = qbeta(c(0.025, 0.975), post_a, post_b))
```

10.5.6 Output & Results

A three-curve plot showing the flat-ish prior, the peaked likelihood, and the posterior between them, plus a summary giving the posterior mean (0.71) and the 95 % equal-tailed credible interval (≈ 0.47 to 0.89). The posterior is closer to the likelihood than to the prior

because the data ($n = 10$) carry more precision than the prior ($a + b = 4$ pseudo-observations).

10.5.7 Interpretation

A reporting sentence: “Combining a Beta(2, 2) prior with 8 successes in 10 trials yielded a Beta(10, 4) posterior with mean 0.71 (95 % credible interval 0.47 to 0.89); the prior contributed the equivalent of about four pseudo-observations and the data dominated the posterior shape.” Always disclose the prior; the same data with a different prior would lead to a slightly different posterior.

10.5.8 Practical Tips

- For conjugate setups (Beta–Binomial, Normal–Normal, Gamma–Poisson, Dirichlet–Multinomial), the posterior is closed-form and arithmetic alone suffices; reach for MCMC only when conjugacy fails.
- Always report a posterior summary that includes both a central tendency and an uncertainty measure; a posterior mean alone is no more informative than a frequentist point estimate.
- Sensitivity to the prior matters most in small samples; if your conclusion changes with a wider or narrower prior, the data are weak and the analysis is genuinely prior-driven.
- Posterior predictive draws (`rbeta(M, post_a, post_b)` followed by `rbinom(M, n_new, theta)`) deliver direct uncertainty intervals for future outcomes.
- Visualising prior, likelihood, and posterior on the same plot is the single most useful pedagogical artefact when explaining Bayesian inference to non-statistical readers.

10.5.9 R Packages Used

Base R for the conjugate update and plotting, `bayestestR` for tidy posterior summaries, and `brms` or `rstanarm` for non-conjugate problems where MCMC is required to obtain posterior draws.

10.5.10 For Reviewers

Use the shared reviewer checklist, together with this method’s assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript’s application of the method.

10.5.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.6 Bayesian ANOVA

10.6.1 Introduction

Classical ANOVA reports an F-statistic, a p-value, and — if the omnibus is significant — pairwise comparisons with a multiple-testing correction tacked on. A Bayesian ANOVA replaces all of that with a single posterior distribution over the group means and any contrast you care to compute. There is no omnibus to clear before contrasts become legitimate, and there is no multiple-comparison adjustment to argue about, because every contrast already has its own honest credible interval reflecting the joint posterior of the parameters that compose it.

10.6.2 Prerequisites

A working understanding of one-way ANOVA, the distinction between fixed and random effects, and Bayesian regression. Familiarity with `emmeans` for marginal means makes the contrast step natural.

10.6.3 Theory

The model is $y_{ij} \sim \mathcal{N}(\mu_i, \sigma^2)$ with $\mu_i = \mu + \alpha_i$. Two parameterisations are common: a fixed-effects parameterisation with weakly informative priors on the contrasts, and a hierarchical parameterisation with $\alpha_i \sim \mathcal{N}(0, \tau^2)$ that pools group estimates toward the grand mean. The hierarchical version is preferable when group counts are large or when groups represent samples from a broader population. Pairwise contrasts and any linear combination of μ_i are computed at the level of MCMC draws, so each draw produces a contrast value and the posterior of the contrast is the empirical distribution of those values.

10.6.4 Assumptions

Approximately Normal residuals within groups, homogeneity of variance unless `sigma` is also modelled by group, exchangeability of groups under the hierarchical version, and proper priors. Independence between observations after group is the same Bayesian assumption as in classical ANOVA.

10.6.5 R Implementation

```r
library(brms); library(bayestestR)

data(PlantGrowth)

fit ← brm(weight ~ group, data = PlantGrowth,
          prior = prior(normal(0, 1), class = "b"),
          chains = 4, iter = 2000, refresh = 0)

# Pairwise contrasts
pairs_summary ← pairs(emmeans::emmeans(fit, ~ group))
pairs_summary

# Bayes factors for effect
bayesfactor_parameters(fit)
```

10.6.6 Output & Results

`emmeans()` returns the posterior marginal mean for each group; `pairs()` returns the posterior of every pairwise difference with an HPD or equal-tailed credible interval. `bayesfactor_parameters()` evaluates the Savage-Dickey density ratio at zero for each contrast, returning a Bayes factor for "non-zero contrast" against the null.

10.6.7 Interpretation

A reporting sentence: "Posterior contrasts from a Bayesian one-way ANOVA showed `trt1` $-$ `ctrl` $= -0.37$ (95 % credible interval -0.95 to 0.21, $\mathrm{BF}_{10} = 1.1$), `trt2` $-$ `ctrl` $= 0.49$ (95 % CrI 0.04 to 0.94, $\mathrm{BF}_{10} = 4.6$); the data weakly support a treatment-2 effect and are inconclusive about treatment 1." Pair the credible interval with the Bayes factor only when the prior under the alternative is disclosed, since the latter depends sensitively on it.

10.6.8 Practical Tips

- For more than five or six groups, switch to a hierarchical $(1 \mid \mathrm{group})$ random-effects parameterisation; shrinkage stabilises small-group estimates and gives a principled population-level summary.
- `emmeans()` works directly on `brms` fits and integrates with `pairs()`, `contrast()`, and `confint()` to produce posterior contrasts on whatever scale you need.
- Credible intervals already adjust for the joint posterior; multiple-comparison corrections in the frequentist sense are unnecessary, but you can still apply a region-of-practical-equivalence (ROPE) test if you want a decision rule.
- For unequal variances, fit `bf(weight ~ group, sigma ~ group)` so the residual scale is also estimated per group; the contrast machinery generalises without changes.
- Report posteriors graphically (`mcmc_areas`, `stat_halfeye`) alongside the numeric summaries; readers absorb the joint structure faster from a plot than from a contrast table.

10.6.9 R Packages Used

`brms` for model specification and sampling, `emmeans` for posterior marginal means and contrasts, `bayestestR` for Bayes factors and ROPE summaries, and `tidybayes` for tidy posterior draws of contrasts when custom plots are needed.

10.6.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.6.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.7 Bayesian Hierarchical Models

10.7.1 Introduction

A hierarchical Bayesian model embeds the group-level parameters of a multilevel design inside a higher-level prior, so that the group-level estimates inform each other rather than being computed in isolation. The pay-off is partial pooling: groups with abundant data are estimated almost from their own observations, groups with little data are pulled toward the overall mean, and every estimate carries a credible interval that reflects both the within-group sample size and the between-group heterogeneity. This single mechanism resolves a long list of practical problems — small-sample variance, unbalanced designs, sparse counts — that frequentist mixed-effects models can only address by reweighting or by appealing to asymptotic theory.

10.7.2 Prerequisites

A working understanding of Bayesian inference, the random-effects mixed model, and the role of exchangeability in motivating shared priors.

10.7.3 Theory

A two-level model has

- Level 1: $y_{ij} \mid \theta_i \sim p(y \mid \theta_i)$ — the data given group-specific parameters.
- Level 2: $\theta_i \mid \mu, \tau \sim p(\theta \mid \mu, \tau)$ — group-specific parameters drawn from a population distribution.
- Level 3 (hyperprior): $\mu \sim p(\mu)$, $\tau \sim p(\tau)$ — priors on the population parameters.

The posterior of θ_i is a precision-weighted average of the within-group estimate (driven by the data in group i) and the population mean μ, with weight inversely proportional to the within-group variance and to the cross-group variance τ^2. Smaller groups rely more on the population mean; larger groups stand on their own data. This is the formal expression of "shrinkage" or "borrowing strength."

10.7.4 Assumptions

Exchangeability of groups (group labels carry no information beyond what is in the predictors) and a hyperprior whose support spans plausible values of μ and τ. A poorly chosen hyperprior — particularly an overly tight prior on τ — can dominate the posterior in small samples and look like a strong empirical claim.

10.7.5 R Implementation

```r
library(brms)

# 30 hospitals, events and trials per hospital
set.seed(2026)
n_hosp <- 30
n_per  <- sample(5:50, n_hosp, replace = TRUE)
true_rates <- rbeta(n_hosp, 3, 12)
events <- rbinom(n_hosp, n_per, true_rates)

d <- data.frame(hospital = factor(1:n_hosp),
                events = events, trials = n_per)

fit <- brm(events | trials(trials) ~ 1 + (1 | hospital),
           data = d, family = binomial,
           chains = 4, iter = 2000, refresh = 0)
summary(fit)
```

10.7.6 Output & Results

`summary(fit)` returns the population-level intercept (overall log-odds), the between-hospital SD on the logit scale, and per-hospital posterior summaries via `ranef(fit)`. The shrinkage of small-hospital estimates toward the population mean is most visible when you plot raw event proportions against posterior means alongside hospital sample size.

10.7.7 Interpretation

A reporting sentence: "A Bayesian binomial hierarchical model estimated the population baseline at $\text{logit}^{-1}(-1.4) = 0.20$ (95 % credible interval 0.16 to 0.24); the between-hospital standard deviation on the logit scale was 0.51 (95 % CrI 0.34 to 0.74), and small-hospital posterior means were drawn substantially toward the population baseline." Always report cluster-specific and overall estimates; one without the other misses the central insight of the hierarchical fit.

10.7.8 Practical Tips

- Partial pooling is the headline benefit: do not throw it away by switching to a fixed-effects fit just because cluster counts are large.
- For divergent transitions, switch to non-centred parameterisation; `brms` does this automatically when you specify `prior(... , class = "sd", ...)`.
- The prior on τ matters enormously when the number of clusters is small; report a sensitivity analysis with at least one wider and one tighter prior.
- For more than two levels (e.g. patient $\subset$ hospital $\subset$ region), nest random effects: `(1 | region/hospital)`. The same shrinkage logic applies at each level.
- Do not interpret cluster-specific posterior means as if they came from independent fits; they are *intentionally* shrunk and that shrinkage is the right answer to "how good is hospital i?"

10.7.9 R Packages Used

`brms` for the hierarchical Bayesian fit, `rstanarm` for pre-compiled `stan_glmer` alternatives, `tidybayes` for tidy cluster-level draws when bespoke shrinkage plots are needed, and `bayesplot` for diagnostic visualisation.

10.7.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.7.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.8 Bayesian Mediation Analysis

10.8.1 Introduction

Mediation analysis decomposes a treatment effect into the part that flows through an intermediate variable (the mediator) and the part that does not. Frequentist approaches build confidence intervals around the product $a_1 b_2$ via Sobel tests or non-parametric bootstrapping; both have to compensate for the fact that the product of two normally distributed estimates is itself non-normal. A Bayesian formulation sidesteps the issue: the joint posterior of (a_1, b_2) is sampled directly, the indirect effect is computed draw-by-draw as the product, and any quantile of the resulting posterior is a valid credible interval without distributional assumptions.

10.8.2 Prerequisites

A working understanding of multiple regression, the concept of mediation in causal inference, and Bayesian regression. Familiarity with multivariate `brms` formulas accelerates the implementation step.

10.8.3 Theory

The classical Baron-Kenny system has two regressions

$$M = a_0 + a_1 X + \varepsilon_M, \qquad Y = b_0 + b_1 X + b_2 M + \varepsilon_Y,$$

with errors that are typically taken Normal. The direct effect of X on Y is b_1, the indirect effect through M is $a_1 b_2$, and the total effect is $b_1 + a_1 b_2$. In the Bayesian formulation, the two equations are fitted jointly so that the posterior carries the joint uncertainty in (a_1, b_2). The indirect-effect posterior is the empirical distribution of $\{a_1^{(s)} b_2^{(s)}\}_{s=1}^{S}$; the credible interval comes from its quantiles. No symmetry assumption is needed and the posterior captures any skew induced by the product.

10.8.4 Assumptions

Linearity of both equations, Normal residuals (or whatever family you specify), no unmeasured confounders of the $M \to Y$ relation, and correctly ordered causal direction $X \to M \to Y$. Mediation analysis is a causal claim; the Bayesian machinery does not make weak design choices stronger.

10.8.5 R Implementation

```
library(brms)

set.seed(2026)
n ← 300
X ← rnorm(n)
M ← 0.5 * X + rnorm(n)
Y ← 0.3 * X + 0.6 * M + rnorm(n)
d ← data.frame(X, M, Y)

# Joint model using brms multivariate syntax
bf1 ← bf(M ~ X)
bf2 ← bf(Y ~ X + M)

fit ← brm(bf1 + bf2 + set_rescor(FALSE), data = d,
          chains = 4, iter = 2000, refresh = 0)

post ← as_draws_df(fit)
indirect ← post$b_M_X * post$b_Y_M
quantile(indirect, c(0.025, 0.5, 0.975))
```

10.8.6 Output & Results

A multivariate `brms` fit jointly estimates the two regressions and stores draws for both. Computing the elementwise product of posterior draws of a_1 and b_2 produces the indirect-effect posterior; quantiles of that posterior give the credible interval directly. The

total effect, the direct effect, and the proportion mediated all come from the same draw matrix without re-fitting.

10.8.7 Interpretation

A reporting sentence: "Bayesian mediation estimated the indirect effect of X on Y through M at 0.30 (95 % credible interval 0.18 to 0.45) and the direct effect at 0.28 (95 % CrI 0.14 to 0.42); the indirect path explained approximately 52 % of the total effect (95 % CrI 35 % to 68 %)." Always report both direct and indirect effects with credible intervals; reporting only the indirect product hides whether the direct path remains substantive.

10.8.8 Practical Tips

- Use multivariate `brms` formulas (`bf(M ~ X) + bf(Y ~ X + M) + set_rescor(FALSE)`) so that the two regressions are fitted jointly and the joint posterior is preserved; running them separately throws away the cross-equation correlation needed for honest indirect-effect intervals.
- Place priors on each regression's coefficients individually; weakly informative defaults ($\mathcal{N}(0, 2.5)$ on standardised scales) are usually appropriate.
- Sensitivity analysis to unmeasured confounding is more important than the choice of estimator; consider `causalsens` or Bayesian instrumental-variable extensions when the assumption is weak.
- For binary or count mediators / outcomes, change the family in the corresponding `bf()` and the indirect-effect computation generalises; just use `posterior_epred()` to translate to the natural-effects scale.
- Reporting the proportion mediated ($a_1 b_2 / (b_1 + a_1 b_2)$) requires care because the denominator can be small or change sign across draws; truncate or report the indirect and total effects separately when the proportion is unstable.

10.8.9 R Packages Used

`brms` for joint multivariate regressions and direct posterior products of coefficients, `bmlm` for purpose-built Bayesian mediation with multilevel structure, `tidybayes` for tidy posterior draws of the indirect effect, and `bayestestR` for credible intervals and ROPE summaries on the derived quantities.

10.8.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.8.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error

- brms / Stan, LOO, hierarchical models

10.9 Bayesian Meta-Analysis

10.9.1 Introduction

Meta-analysis with k studies is awkward for frequentist random-effects estimators when k is small: REML and DerSimonian-Laird both rely on asymptotic approximations that break down when only five or six studies contribute, and the between-study variance τ^2 can collapse to zero or be wildly imprecise. A Bayesian formulation puts an explicit prior on τ and propagates uncertainty into both the pooled effect and the prediction interval for a new study, which is exactly the inferential question that drives most meta-analyses in clinical and biomedical work.

10.9.2 Prerequisites

A working understanding of fixed- and random-effects meta-analysis, exchangeability assumptions, and hierarchical Bayesian models. Familiarity with `brms` formula syntax accelerates the implementation step.

10.9.3 Theory

The hierarchical model is

$$y_i \sim \mathcal{N}(\theta_i, s_i^2), \qquad \theta_i \sim \mathcal{N}(\mu, \tau^2),$$

where y_i and s_i are the observed effect estimate and its standard error from study i, θ_i is the latent study-specific effect, μ is the population effect, and τ is the between-study standard deviation. Typical priors are weakly informative: $\mu \sim \mathcal{N}(0, 2)$ on the standardised-mean-difference scale and $\tau \sim \text{half-Normal}(0, 0.5)$ or $\text{half-Cauchy}(0, 0.3)$. The half-Cauchy gives the heavier tail favoured when very large heterogeneity is plausible. The posterior delivers μ, τ, every θ_i shrunk toward μ, and a predictive distribution $\mathcal{N}(\mu, \tau^2)$ for the next study.

10.9.4 Assumptions

Exchangeable studies, accurate within-study standard errors s_i, and proper priors. The within-study sampling distribution is taken as known; if s_i are themselves estimated from few patients, an additional level of hierarchy is needed.

10.9.5 R Implementation

```r
library(brms)

studies ← data.frame(
  yi = c(0.2, 0.3, -0.05, 0.4, 0.25),
  sei = c(0.10, 0.15, 0.08, 0.12, 0.20)
)

fit ← brm(yi | se(sei) ~ 1 + (1 | seq_len(nrow(studies))),
          data = studies,
          prior = prior(normal(0, 1), class = "Intercept") +
                  prior(cauchy(0, 0.5), class = "sd"),
          chains = 4, iter = 4000, refresh = 0)

summary(fit)
```

10.9.6 Output & Results

`summary(fit)` returns the pooled effect (`Intercept`) and the between-study SD (`sd(seq_len(...))`) with 95 % credible intervals, $\hat{R}$, and ESS. Per-study posterior means come from `ranef(fit)`; the predictive distribution for a future study is `posterior_predict()` with `re_formula = NA`.

10.9.7 Interpretation

A reporting sentence: "A Bayesian random-effects meta-analysis with a half-Cauchy(0, 0.5) prior on τ pooled the five studies to an effect of 0.23 (95 % credible interval 0.07 to 0.39); the between-study heterogeneity was small ($\tau = 0.09$, 95 % CrI 0.01 to 0.25), and the 95 % predictive interval for a future trial was -0.05 to 0.51." Reporting the predictive interval together with the pooled estimate avoids the false sense of precision that a credible interval on μ alone can convey.

10.9.8 Practical Tips

- With $k < 10$ studies, the prior on τ matters substantially; a half-$t(3, 0, 1)$ or half-Normal(0, 0.5) is a reasonable compromise between letting the data speak and avoiding implausible values.
- Always report the prior in the methods section and conduct a sensitivity analysis with at least one wider and one tighter prior on τ.
- For binary outcomes, fit on the log-odds-ratio or risk-ratio scale rather than risk differences; the Normal-Normal hierarchy is more reasonable on the log scale.
- The `bayesmeta` package provides closed-form computations for simple models and is faster than MCMC when the prior is conjugate.
- For network meta-analysis with multiple treatments, switch to `gemtc` or `multinma`; the random-effects machinery generalises but the implementation is non-trivial.

10.9.9 R Packages Used

`brms` for the hierarchical Bayesian fit with study-level standard errors, `bayesmeta` for closed-form alternatives, `bayesplot` for diagnostic plots, and `gemtc` / `multinma` for the network-meta extension.

10.9.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.9.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.10 Bayesian Mixed Models

10.10.1 Introduction

Bayesian mixed models treat the random-effect variance components as parameters with their own priors instead of plug-in estimates from REML. The pay-off is most visible in small samples and at the boundary: ML and REML routinely return zero variance estimates when only a handful of clusters contribute, but a half-Normal or half-Cauchy prior keeps the variance away from zero in a principled way and propagates the resulting uncertainty into every fixed-effect interval. The cost is computational — sampling instead of optimising — but for any analysis that depends on credible variance components, that cost is a worthwhile investment.

10.10.2 Prerequisites

A working understanding of frequentist mixed-effects models (`lme4`-style formulas), Bayesian regression, and the role of half-distributions as priors on standard deviations.

10.10.3 Theory

A random-intercept-and-slope model has the form

$$y_{ij} = \beta_0 + \beta_1 X_{ij} + u_{0i} + u_{1i} X_{ij} + \varepsilon_{ij},$$

with $(u_{0i}, u_{1i}) \sim \mathcal{N}(0, \Sigma)$ and $\varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$. Priors are placed on every component: weakly informative Normal priors on β, half-Normal or half-Cauchy on the diagonal of Σ, an LKJ prior on the correlation, and an exponential or half-t on σ. The LKJ(η) prior with $\eta > 1$ pulls correlations toward zero gently while still allowing strong correlations when

the data demand them. Posterior draws of u_i are *shrunk* toward zero by an amount that depends on within-cluster signal-to-noise — the same partial-pooling behaviour that makes mixed models attractive in the first place.

10.10.4 Assumptions

Conditional independence of observations given random effects, exchangeability of clusters, approximately Normal random effects with the prior structure above, and proper priors on every variance and correlation parameter. Convergence diagnostics ($\hat{R}$, divergent transitions, ESS) are part of the assumptions in practice.

10.10.5 R Implementation

```
library(brms)

data(sleepstudy, package = "lme4")

fit ← brm(Reaction ~ Days + (Days | Subject), data = sleepstudy,
           prior = prior(normal(0, 30), class = "b") +
                   prior(normal(250, 50), class = "Intercept") +
                   prior(exponential(0.1), class = "sd") +
                   prior(exponential(0.1), class = "sigma") +
                   prior(lkj(2), class = "cor"),
           chains = 4, iter = 2000, refresh = 0)
summary(fit)

# Visualise random effects
ranef(fit)
```

10.10.6 Output & Results

`summary(fit)` reports the population-level fixed effects, the standard deviations of the random intercept and slope, the correlation between them, and the residual scale, all with credible intervals, $\hat{R}$, and ESS. `ranef(fit)` returns a list of per-cluster posterior summaries — the analogue of `lme4::ranef()` but with full credible intervals.

10.10.7 Interpretation

A reporting sentence: “A Bayesian linear mixed model estimated the population-level slope at 10.5 ms per day of sleep deprivation (95 % credible interval 7.4 to 13.6); between-subject heterogeneity in slope was substantial ($\mathrm{SD} = 6.0$, 95 % CrI 4.0 to 8.8) and weakly negatively correlated with the intercept ($r = -0.13$, 95 % CrI -0.49 to 0.27).” Always report the variance components with their credible intervals; point estimates alone can be misleading when posteriors are skewed.

10.10.8 Practical Tips

- Use `LKJ(2)` for correlation priors as a sensible default; tighten to `LKJ(4)` only when you have strong reason to expect weak correlations.
- For divergent transitions in hierarchical models, switch to non-centred parameterisation (`brms` does this automatically when you set `prior(... , class =`

`"sd", ...`).

- Standardise continuous predictors before fitting; weakly informative priors are then meaningful across columns.
- Compare nested random-effect structures with `loo_compare()` rather than likelihood-ratio tests; LOO works correctly when ML-LRT statistics are on the boundary of the parameter space.
- For very small n_{cluster} (under 5), report a sensitivity analysis across at least two priors on the random-effect SD; the choice can dominate the posterior.
- `rstanarm::stan_lmer()` mirrors `lme4` syntax and uses pre-compiled Stan models, useful when fit time matters and the model is standard.

10.10.9 R Packages Used

`brms` for formula-based Bayesian mixed models with full prior control, `rstanarm` for the pre-compiled `stan_lmer` interface, `bayesplot` for diagnostic plots, and `tidybayes` for tidy posterior draws when bespoke shrinkage plots are needed.

10.10.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.10.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.11 The Beta-Binomial Model

10.11.1 Introduction

The beta-binomial conjugate pair is the textbook entry point to Bayesian inference: a Beta prior on a probability combined with a binomial likelihood yields a Beta posterior whose parameters are obtained by adding successes to one shape parameter and failures to the other. The closed form makes the entire workflow — prior elicitation, posterior, credible interval, posterior predictive — tractable on a calculator. It is also the conceptual backbone for hierarchical models of proportions, which use a Beta prior at the population level to share strength across many groups.

10.11.2 Prerequisites

Familiarity with the Beta distribution as a flexible density on $[0, 1]$, the binomial likelihood for independent Bernoulli trials, and the basic mechanics of Bayesian updating.

10.11.3 Theory

With prior $\theta \sim \mathrm{Beta}(a, b)$ and likelihood $X \sim \mathrm{Binomial}(n, \theta)$, the posterior is

$$\theta \mid x \sim \mathrm{Beta}(a + x, b + n - x).$$

The shape parameters of the prior have a direct interpretation as pseudo-counts: a pseudo-successes and b pseudo-failures observed before the data arrived, so the prior effective sample size is $a + b$. The posterior mean is

$$\mathbb{E}[\theta \mid x] = \frac{a + x}{a + b + n},$$

a weighted average of the prior mean $a/(a+b)$ and the sample proportion x/n, with weights determined by the prior pseudo-count and the actual sample size. Conjugacy also gives closed-form expressions for the posterior variance, credible intervals, and the predictive distribution for future Bernoulli trials.

10.11.4 Assumptions

Independent Bernoulli trials with a constant success probability θ, a Beta prior plausible enough that its tails do not exclude reasonable values, and a sample size that is fixed in advance — informative censoring or stopping rules invalidate the binomial likelihood.

10.11.5 R Implementation

```
# Prior Beta(2, 2) -- weakly informative around 0.5
# Data: 15 successes in 25 trials
a <- 2; b <- 2
x <- 15; n <- 25

post_a <- a + x
post_b <- b + n - x

c(prior_mean = a / (a + b),
  sample_prop = x / n,
  posterior_mean = post_a / (post_a + post_b),
  ci95 = qbeta(c(0.025, 0.975), post_a, post_b))

theta <- seq(0, 1, length.out = 400)
plot(theta, dbeta(theta, a, b), type = "l", col = "grey",
     ylab = "Density", main = "Beta-Binomial update")
lines(theta, dbeta(theta, post_a, post_b), col = "steelblue", lwd = 2)
```

10.11.6 Output & Results

A short numeric vector with the prior mean (0.50), the sample proportion (0.60), the posterior mean (0.59), and the 95 % equal-tailed credible interval (0.40 to 0.76). The prior-versus-posterior plot makes the pull of the prior visible: with only 25 trials the posterior shifts noticeably toward the data but retains the curvature of the prior.

10.11.7 Interpretation

A reporting sentence: "A Beta(2, 2) prior combined with 15 successes in 25 trials yielded a Beta(17, 12) posterior with mean 0.59 (95 % credible interval 0.40 to 0.76); the data have moved the posterior 0.09 above the prior mean while excluding values below 0.40 with 97.5 % probability." Always disclose the prior in the methods section so that readers can see how much shrinkage it imposes.

10.11.8 Practical Tips

- $\mathrm{Beta}(1, 1)$ is the uniform reference prior; $\mathrm{Beta}(0.5, 0.5)$ is Jeffreys; both keep posterior inference data-driven.
- For genuinely informative priors, set $a + b$ equal to the prior effective sample size you can defend (e.g. results from a prior trial of size 20 with proportion 0.7 give $\mathrm{Beta}(14, 6)$).
- Use `qbeta()` for credible intervals and `pbeta()` for posterior tail probabilities; HPD intervals come from `HDInterval::hdi(qbeta(p, post_a, post_b))` evaluated on a grid.
- For multi-category outcomes, generalise to the Dirichlet–multinomial, which inherits the conjugate structure on a probability simplex.
- When the prior is contested, report a sensitivity analysis across at least $\mathrm{Beta}(1, 1)$, $\mathrm{Beta}(2, 2)$, and one informative pair; if the conclusion holds across all three, the result is robust to prior choice.

10.11.9 R Packages Used

Base R suffices for the conjugate update (`dbeta`, `pbeta`, `qbeta`, `rbeta`); `bayestestR::point_estimate()` and `HDInterval::hdi()` add tidy summaries and HPD intervals when a more polished workflow is needed.

10.11.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.11.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.12 brms Basics

10.12.1 Introduction

`brms` is the most widely used high-level interface between R and Stan. It accepts the familiar `lme4` formula syntax for fixed and random effects, automatically translates the model to Stan, compiles it, runs the No-U-Turn Sampler (NUTS), and returns a posterior together with diagnostic and predictive helpers. This means you can fit Bayesian generalised linear, mixed, multinomial, ordinal, censored, distributional, or non-linear models without writing Stan by hand, while still being able to inspect or override the generated code when the situation demands it.

10.12.2 Prerequisites

Comfort with classical regression, an understanding of priors and posteriors, and a working `rstan` or `cmdstanr` toolchain (a C++ compiler is required on first use). Familiarity with `lme4` formulas accelerates the learning curve dramatically because almost every random-effects expression carries over unchanged.

10.12.3 Theory

A `brms` model has three pieces: a **formula** (mean structure plus optional distributional terms), a **family** (likelihood and link), and **priors** (per parameter class). Random effects use `(slope | group)` syntax, with `||` for uncorrelated terms. Distributional models, set up through `bf(y ~ ..., sigma ~ ...)`, let any parameter of the family depend on predictors. Priors are attached by class: `b` for population-level slopes, `Intercept` for the centred intercept, `sd` for random-effect standard deviations, `sigma` for residual scale, and `cor` for correlation matrices via the LKJ distribution. Defaults are weakly informative but should be reviewed for any analysis intended for publication.

10.12.4 Assumptions

Inherits the assumptions of the chosen family — for `gaussian` that means linearity in the linear predictor, additivity, and Normal residuals after partial pooling. Convergence diagnostics ($\hat{R} < 1.01$, no divergent transitions, sensible effective sample size) are also a precondition for trusting any posterior summary.

10.12.5 R Implementation

```r
library(brms)

data(sleepstudy, package = "lme4")

fit ← brm(Reaction ~ Days + (Days | Subject),
          data = sleepstudy,
          family = gaussian,
          prior = prior(normal(0, 30), class = "b") +
                  prior(exponential(0.1), class = "sigma") +
                  prior(exponential(0.1), class = "sd") +
                  prior(lkj(2), class = "cor"),
          chains = 4, iter = 2000, refresh = 0)

summary(fit)
pp_check(fit)
```

10.12.6 Output & Results

`summary()` reports posterior means, standard errors, 95 % credible intervals, $\hat{R}$, bulk effective sample size, and tail effective sample size for every population-level coefficient, group-level standard deviation, and the residual scale. `pp_check()` overlays simulated datasets on the observed outcome; if the empirical density falls inside the cloud of replicated densities, the model captures the marginal distribution adequately.

10.12.7 Interpretation

A reporting sentence: “A Bayesian linear mixed model (`brms` 2.21, four chains, 2,000 iterations) estimated reaction time to increase by 10.5 ms per day of sleep deprivation (95 % CrI 7.4 to 13.6), with substantial between-subject heterogeneity in both intercept ($SD = 26$) and slope ($SD = 6.5$); $\hat{R} = 1.00$ for all parameters.” Always pair the point estimate with the credible interval and a convergence statistic.

10.12.8 Practical Tips

- Run `get_prior()` on your formula before fitting; it lists every prior class so you can override defaults instead of inheriting them silently.
- Inspect `stancode(fit)` and `standata(fit)` when something behaves oddly; the generated code is readable and reveals exactly how `brms` parameterised the model.
- `conditional_effects()` produces marginal effect plots with credible bands and is the fastest way to communicate non-linear or interaction effects to non-Bayesian audiences.
- Use `add_criterion(fit, "loo")` to attach LOO-CV for later model comparison; storing it on the object avoids re-fitting.
- For very large datasets, switch to `backend = "cmdstanr"` and `threads_per_chain = 2` to exploit within-chain parallelism.
- When defaults yield divergent transitions, increase `adapt_delta` to 0.95 or 0.99 before reaching for stronger priors.

10.12.9 R Packages Used

`brms` for model specification and compilation, `rstan` or `cmdstanr` as the sampling backend, `bayesplot` for diagnostic plots used internally by `pp_check()`, and `posterior` for handling draw arrays.

10.12.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.12.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.13 Conjugate Priors

10.13.1 Introduction

Before MCMC made arbitrary Bayesian models tractable, conjugate priors provided the only route to a tractable posterior. A prior is **conjugate** to a likelihood if the resulting posterior belongs to the same distributional family as the prior – the prior and posterior share a form, with updated parameters that have simple closed-form expressions. Conjugate pairs remain pedagogically central because they make the Bayesian update arithmetic, and they remain practically useful as building blocks within larger hierarchical models.

10.13.2 Prerequisites

The reader should know Bayes' theorem and be comfortable with the beta, normal, and gamma distributions. Familiarity with basic R plotting is assumed.

10.13.3 Theory

10.13.3.1 Beta-Binomial

For a binomial observation $Y \sim \text{Binomial}(n, \theta)$ with success probability θ and a beta prior $\theta \sim \text{Beta}(\alpha, \beta)$, the posterior is

$$\theta \mid Y = y \;\sim\; \text{Beta}(\alpha + y, \beta + n - y).$$

The update rule is nothing more than "add successes to the first shape parameter and failures to the second". The posterior mean is $(\alpha + y)/(\alpha + \beta + n)$, a weighted average of the prior mean $\alpha/(\alpha+\beta)$ and the observed proportion y/n, with weights determined by the prior's "effective sample size" $\alpha + \beta$.

10.13.3.2 Normal-Normal with known variance

For observations $Y_1, \ldots, Y_n \overset{iid}{\sim} \mathcal{N}(\mu, \sigma^2)$ with σ^2 known, and a normal prior $\mu \sim \mathcal{N}(\mu_0, \tau_0^2)$, the posterior is

$$\mu \mid y \;\sim\; \mathcal{N}(\mu_n, \tau_n^2),$$

where the posterior precision is $1/\tau_n^2 = 1/\tau_0^2 + n/\sigma^2$, and the posterior mean is a precision-weighted average $\mu_n = \tau_n^2(\mu_0/\tau_0^2 + n\bar{y}/\sigma^2)$.

10.13.3.3 Gamma-Poisson

For independent counts $Y_i \sim \text{Poisson}(\lambda)$ and a gamma prior $\lambda \sim \text{Gamma}(\alpha, \beta)$ (shape-rate parameterisation), the posterior is

$$\lambda \mid y \;\sim\; \text{Gamma}\Big(\alpha + \sum y_i,\; \beta + n\Big)\,.$$

Each of these updates can be memorised as "add the data to the prior parameters", with the precise form determined by the sufficient statistics of the likelihood.

10.13.4 Assumptions

Conjugate analyses assume the same independence and distributional conditions as the corresponding likelihoods. In addition, they assume that the prior is genuinely an expression of prior belief, not a mathematical convenience chosen for conjugacy alone. An informative conjugate prior that contradicts the data produces a posterior that is a compromise between the two; if this is not what you intended, the prior was a bad choice.

10.13.5 R Implementation

A visualisation of the beta-binomial update makes the idea concrete:

```r
library(ggplot2)

alpha_0 <- 2
beta_0  <- 2
y       <- 8
n       <- 10

alpha_n <- alpha_0 + y
beta_n  <- beta_0 + n - y

theta <- seq(0, 1, length.out = 401)
df <- data.frame(
  theta = rep(theta, 3),
  density = c(dbeta(theta, alpha_0, beta_0),
              dbinom(y, n, theta) * n,
              dbeta(theta, alpha_n, beta_n)),
  kind = factor(rep(c("Prior", "Likelihood (scaled)", "Posterior"),
                    each = 401),
                levels = c("Prior", "Likelihood (scaled)", "Posterior"))
)

ggplot(df, aes(x = theta, y = density, colour = kind, linetype = kind)) +
  geom_line(linewidth = 1) +
  labs(x = expression(theta), y = "Density",
       title = "Beta-Binomial update with y = 8 of n = 10") +
  theme_minimal(base_size = 12)
```

A `Beta(2, 2)` prior (symmetric, weakly informative around 0.5) combined with 8 successes in 10 trials gives a `Beta(10, 4)` posterior concentrated around 0.71.

10.13.6 Output & Results

Posterior summaries from the beta distribution are:

- Mean: $(\alpha + y)/(\alpha + \beta + n) = 10/14 \approx 0.714$
- 95% equal-tailed credible interval: `qbeta(c(0.025, 0.975), 10, 4)` $\approx (0.47, 0.91)$
- Probability that $\theta > 0.5$: `1 - pbeta(0.5, 10, 4)` ≈ 0.95.

10.13.7 Interpretation

The posterior in a conjugate model is not an approximation; it is the exact posterior under the stated prior. A Bayesian reporting this analysis would write: “With a weakly informative $\text{Beta}(2, 2)$ prior and data of 8 successes in 10 trials, the posterior for θ is $\text{Beta}(10, 4)$, with posterior mean 0.71 (95% CrI 0.47 to 0.91). The posterior probability that θ exceeds 0.5 is 0.95.”

10.13.8 Practical Tips

- Report the prior explicitly so readers can judge its influence.
- Use informative priors when justified by domain knowledge; use weakly informative priors as the default for exploratory analyses.
- Conjugate updating extends to multi-parameter cases (normal-inverse-gamma for (μ, σ^2), Dirichlet-multinomial for categorical probabilities), but outside the exponential family it usually does not hold.

- For small-sample binomial problems the beta-binomial is particularly useful: it avoids the ad-hoc continuity corrections of frequentist proportion tests and gives natural credible intervals.
- When your likelihood is non-conjugate, do not force a conjugate prior on it; use MCMC or variational methods instead.

10.13.9 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.13.10 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.14 Credible Intervals

10.14.1 Introduction

A credible interval is the Bayesian summary of posterior uncertainty most often reported alongside a point estimate. Unlike a confidence interval, whose probability statement refers to long-run frequency under hypothetical resampling, a credible interval supports the direct claim that "given the data and prior, there is a 95 % probability that the parameter lies inside this interval." That single sentence is the reason students consistently misinterpret confidence intervals — and the reason credible intervals are increasingly preferred in applied biomedical reporting.

10.14.2 Prerequisites

A working understanding of Bayesian inference, posterior distributions, and the distinction between a posterior and a sampling distribution.

10.14.3 Theory

Two definitions dominate practice:

- **Equal-tailed interval (ETI)**: the interval bounded by the $\alpha/2$ and $1-\alpha/2$ quantiles of the posterior. It is invariant under monotone reparameterisation and easy to compute from MCMC draws.
- **Highest posterior density (HPD) interval**: the narrowest set with $1-\alpha$ posterior mass; for unimodal posteriors it is a single interval centred on the region of highest density, and it is always at least as short as the ETI.

For symmetric posteriors the two intervals coincide; for skewed posteriors they can differ markedly. A 95 % level is conventional but arbitrary — some Bayesian methodologists advocate 89 % to discourage misreading the threshold as a hypothesis test.

10.14.4 Assumptions

A proper posterior distribution and either MCMC draws (post-warm-up, with adequate effective sample size) or a closed-form posterior. ETI endpoints are quantiles and tolerate small samples better than HPD endpoints, which depend on density estimation in the tails.

10.14.5 R Implementation

```r
library(HDInterval)

set.seed(2026)
theta <- rbeta(10000, 10, 4)

# Equal-tailed
quantile(theta, c(0.025, 0.975))

# HPD
hdi(theta, credMass = 0.95)
```

10.14.6 Output & Results

`quantile()` returns the ETI; `hdi()` returns the HPD. For a Beta(10, 4) posterior the ETI is approximately $(0.50, 0.90)$ and the HPD is approximately $(0.53, 0.93)$ — slightly shifted toward the right because the posterior is left-skewed and the HPD chases the highest density region rather than splitting tail probability evenly.

10.14.7 Interpretation

A reporting sentence: “The posterior probability had a 95 % equal-tailed credible interval of 0.50 to 0.90; the highest-posterior-density interval was 0.53 to 0.93. There is a 95 % posterior probability that the true proportion falls in either interval, given the prior and the observed data.” Always state which interval is reported; mixing definitions across tables in the same paper is a common reproducibility headache.

10.14.8 Practical Tips

- Report ETIs for symmetric or near-symmetric posteriors; they are more familiar and reparameterisation-invariant.
- Report HPDs for skewed posteriors when the additional precision matters and the density estimate is stable.
- HPD intervals can be disjoint under multimodal posteriors; set `allowSplit = TRUE` in `HDInterval::hdi()` so the structure is preserved.
- Disclose the prior in any reporting sentence; the credible interval is not a property of the data alone.

- Use post-warm-up MCMC draws and check ESS in the tails — credible-interval endpoints depend on quantile estimation, which needs more effective draws than the mean.

10.14.9 R Packages Used

`HDInterval` for the canonical `hdi()` function, `bayestestR::ci()` for an integrated wrapper that selects between ETI and HPD on `brms` and `rstanarm` model objects, and base R (`quantile()`) for the equal-tailed case.

10.14.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.14.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.15 Divergences and Pairs Plots

10.15.1 Introduction

A divergent transition is HMC's most informative warning: it tells you that the leapfrog integrator could not navigate a region of the posterior accurately, and the chain therefore did *not* explore that region. Unlike a high $\hat{R}$, which can sometimes be cured by running longer chains, a divergent transition signals a structural problem with the posterior geometry that no amount of additional sampling will fix. Diagnosing where divergences occur — typically through pairs plots that colour divergent draws — is the path to a reparameterisation that lets the sampler do its job.

10.15.2 Prerequisites

A working understanding of Hamiltonian Monte Carlo, leapfrog integration, and the geometric origins of divergent transitions; familiarity with `bayesplot::mcmc_pairs()` for visualising joint posteriors.

10.15.3 Theory

The leapfrog integrator approximates Hamiltonian dynamics with finite step size ϵ. In regions of high curvature (typically funnels in hierarchical models, where a small variance parameter shrinks all dependent effects), even a moderate ϵ produces a Hamiltonian error

large enough that the proposal is biased. Stan flags such proposals as divergent and refuses to use them, so the chain effectively avoids the problem region instead of exploring it. The classical remedy is a *non-centred* reparameterisation: replacing $\beta_i \sim \mathcal{N}(\mu, \sigma^2)$ with $\tilde{z}_i \sim \mathcal{N}(0,1)$ and $\beta_i = \mu + \sigma\tilde{z}_i$ decouples the funnel geometry and makes the posterior easier to sample.

10.15.4 Assumptions

HMC or NUTS sampling; the divergent-transition flag is specific to these algorithms. Random-walk samplers do not produce divergences but suffer different pathologies (slow mixing, mode-trapping) that need separate diagnostics.

10.15.5 R Implementation

```r
library(brms); library(bayesplot)

data(sleepstudy, package = "lme4")

fit ← brm(Reaction ~ Days + (Days | Subject), data = sleepstudy,
          chains = 4, iter = 2000, refresh = 0,
          control = list(adapt_delta = 0.99))

# Divergence diagnostics
rstan::check_hmc_diagnostics(fit$fit)

# Pairs plot of parameters with divergences coloured
np ← nuts_params(fit)
mcmc_pairs(as.array(fit), pars = c("b_Days", "sd_Subject__Days"),
           np = np, np_style = pairs_style_np(div_color = "red"))
```

10.15.6 Output & Results

`check_hmc_diagnostics()` reports the count of divergent transitions, tree-depth saturation events, and the energy Bayesian fraction of missing information (E-BFMI). `mcmc_pairs()` overlays the joint draws of two parameters with red points marking divergences; clustering of red points along the boundary of the parameter space pinpoints the geometry that broke.

10.15.7 Interpretation

A reporting sentence: “An initial fit produced 12 divergent transitions concentrated near small values of the random-slope standard deviation; switching to a non-centred parameterisation and increasing `adapt_delta` to 0.99 eliminated all divergences and yielded $\hat{R} = 1.00$ for every parameter.” Report the diagnostic and the fix together; a published Bayesian analysis with unmentioned divergences is not a credible analysis.

10.15.8 Practical Tips

- Any divergence is worth investigating; even a single one signals a region of the posterior the sampler did not explore correctly.

- The classic funnel — variance parameter near zero with correlated dependent effects — is the most common offender; non-centred parameterisation is almost always the right fix.
- Increasing `adapt_delta` to 0.95 or 0.99 reduces step size and eliminates marginal divergences, but it is a band-aid: if the geometry is genuinely difficult, reparameterise.
- `bayesplot::mcmc_parcoord()` complements pairs plots by showing divergent transitions across many parameters simultaneously, which is helpful in models with dozens of variance components.
- When divergences persist after non-centred parameterisation and high `adapt_delta`, examine the prior — improperly tight priors can pinch the posterior into geometry that no sampler can navigate.

10.15.9 R Packages Used

`bayesplot` for `mcmc_pairs()`, `mcmc_parcoord()`, and `nuts_params()`; `rstan` and `cmdstanr` for the underlying diagnostic outputs (`check_hmc_diagnostics()`, `get_sampler_params()`); `posterior` for tidy access to NUTS metadata when building custom diagnostic summaries.

10.15.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.15.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.16 The Gamma-Poisson Model

10.16.1 Introduction

The Gamma–Poisson conjugate pair is the count-data analogue of the Beta–Binomial model: a Gamma prior on a Poisson rate yields a Gamma posterior whose parameters are obtained by adding the observed event count and the observed exposure to the prior shape and rate respectively. It is the natural starting point whenever you are inferring an event rate per unit time, area, or population, and it generalises directly to Poisson regression once predictors enter the linear predictor.

10.16.2 Prerequisites

A working understanding of the Poisson distribution, the Gamma distribution, and the concept of conjugacy.

10.16.3 Theory

With prior $\lambda \sim \text{Gamma}(\alpha_0, \beta_0)$ and likelihood $y_i \sim \text{Poisson}(\lambda t_i)$ where t_i is the exposure for observation i, the posterior is

$$\lambda \mid y \sim \text{Gamma}\left(\alpha_0 + \sum_i y_i,\ \beta_0 + \sum_i t_i\right).$$

The prior shape α_0 behaves like a pseudo-count of events and the prior rate β_0 like a pseudo-exposure, so a $\text{Gamma}(2, 1)$ prior is equivalent to two prior events observed in one unit of exposure. The posterior mean is $(\alpha_0 + \sum y_i)/(\beta_0 + \sum t_i)$ and the posterior variance is the posterior mean divided by the posterior rate. The Poisson–Gamma compound distribution is a Negative-Binomial, which gives the posterior predictive distribution in closed form.

10.16.4 Assumptions

Independent Poisson counts conditional on the rate, an exposure t_i that is known and not estimated from the same data, and a constant rate over each observation window (after conditioning on any covariates). Overdispersion relative to Poisson — variance exceeding the mean — motivates a Negative-Binomial likelihood instead.

10.16.5 R Implementation

```
# Prior Gamma(2, 1) -- 2 events per 1 unit of exposure
alpha0 ← 2; beta0 ← 1

# Data: 15 events in 10 units of exposure
y ← 15; t ← 10

alpha_n ← alpha0 + y
beta_n  ← beta0 + t

c(posterior_mean = alpha_n / beta_n,
  posterior_sd   = sqrt(alpha_n / beta_n^2),
  ci95 = qgamma(c(0.025, 0.975), alpha_n, beta_n))
```

10.16.6 Output & Results

Closed-form summaries: posterior mean approximately 1.55 events per unit exposure, posterior SD approximately 0.38, and 95 % equal-tailed credible interval roughly $(0.94, 2.28)$. The posterior is right-skewed because Gamma densities have a heavier right tail than left, so report the median or HPD when the asymmetry matters.

10.16.7 Interpretation

A reporting sentence: "A $\mathrm{Gamma}(2, 1)$ prior combined with 15 events in 10 units of exposure yielded a $\mathrm{Gamma}(17, 11)$ posterior with mean 1.55 events per unit exposure (95 % credible interval 0.94 to 2.28); the prior contributed the equivalent of two pseudo-events and one pseudo-unit of exposure, which the data overwhelmed." Quote both the posterior mean and the credible interval, and note any skew.

10.16.8 Practical Tips

- Weakly informative defaults: $\mathrm{Gamma}(0.5, 0.5)$ or $\mathrm{Gamma}(1, 1)$; the latter has prior mean 1 and prior variance 1.
- For Poisson regression, place priors on regression coefficients on the log-rate scale (typically $\mathcal{N}(0, 1)$ or $\mathcal{N}(0, 2.5)$) rather than on the rate itself; the linear predictor is more naturally constrained.
- The posterior predictive distribution is a Negative-Binomial: $\tilde{y} \mid y \sim \mathrm{NegBinom}\big(\alpha_n, t_{\mathrm{new}}/(\beta_n + t_{\mathrm{new}})\big)$, so prediction intervals come for free without simulation.
- Persistent overdispersion after fitting motivates a Negative-Binomial likelihood with a separate dispersion parameter; the Gamma–Poisson is its conjugate root.
- For zero-inflation (more zeros than the Poisson predicts), augment with a zero-inflation component rather than tightening the prior.

10.16.9 R Packages Used

Base R suffices for the conjugate update (`dgamma` , `qgamma`); `bayestestR` for tidy summaries; `brms` with `family = poisson()` (or `negbinomial()`) when covariate-dependent rates are needed; `rstanarm::stan_glm` with `family = poisson` for a quick GLM-style alternative.

10.16.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.16.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.17 Gibbs Sampling

10.17.1 Introduction

Gibbs sampling is one of the oldest practical MCMC algorithms and remains an instructive special case of more general samplers. It exploits a structural property of multidimensional posteriors: even when the joint distribution is intractable, the conditional distribution of one parameter given all others can often be derived in closed form. Cycling through the parameters and updating each from its full conditional generates a Markov chain whose stationary distribution is the joint posterior, with no acceptance step and no tuning of proposals.

10.17.2 Prerequisites

A working understanding of MCMC, conditional probability, and the conjugate-prior structure that makes full conditionals tractable for canonical models (Normal–Normal, Beta–Binomial, Gamma–Poisson, Dirichlet–Multinomial).

10.17.3 Theory

For a joint posterior $p(\theta_1, \ldots, \theta_K \mid y)$, the algorithm initialises at some $\theta^{(0)}$ and, at iteration t, sweeps through the parameters in order:

$$\theta_k^{(t)} \sim p\big(\theta_k \mid \theta_1^{(t)}, \ldots, \theta_{k-1}^{(t)}, \theta_{k+1}^{(t-1)}, \ldots, \theta_K^{(t-1)}, y\big)$$

for $k = 1, \ldots, K$. Under standard regularity conditions, the chain is irreducible and aperiodic and converges to the joint posterior. Each update is a draw from a one-dimensional or low-dimensional conditional, often via standard distribution libraries; no Metropolis acceptance is needed because the proposal *is* the conditional. When some conditionals are not available analytically, the affected blocks are updated with a Metropolis step embedded inside the Gibbs sweep (“Metropolis-within-Gibbs”).

10.17.4 Assumptions

Every full conditional must be either a recognisable distribution or a distribution one can sample from with a Metropolis step. Strongly correlated parameters update slowly under a coordinate-wise sweep; this is the central pathology of Gibbs and the reason HMC is now preferred for general continuous models.

10.17.5 R Implementation

```
set.seed(2026)

# Conjugate normal-normal model: mu and sigma2 both unknown
# Priors: mu ~ N(mu0, tau0^2); 1/sigma2 ~ Gamma(alpha, beta)
y ← rnorm(50, 3, 2)
n ← length(y)
mu0 ← 0; tau0 ← 10
alpha ← 2; beta ← 1

n_iter ← 5000
mu_chain    ← numeric(n_iter)
sig2_chain  ← numeric(n_iter)
sig2_chain[1] ← var(y)

for (t in 2:n_iter) {
  # Sample mu | sigma^2, y
  prec ← 1/tau0^2 + n/sig2_chain[t-1]
  mean_mu ← (mu0/tau0^2 + sum(y)/sig2_chain[t-1]) / prec
  mu_chain[t] ← rnorm(1, mean_mu, sqrt(1/prec))

  # Sample sigma^2 | mu, y  (inverse gamma)
  alpha_n ← alpha + n/2
  beta_n  ← beta + sum((y - mu_chain[t])^2)/2
  sig2_chain[t] ← 1/rgamma(1, alpha_n, beta_n)
}

# Summaries
c(mean_mu = mean(mu_chain[-(1:1000)]),
  mean_sigma = mean(sqrt(sig2_chain[-(1:1000)])))
```

10.17.6 Output & Results

The chain produces 5,000 draws of μ and σ^2. Discarding the first 1,000 as burn-in and summarising the rest gives posterior means for μ near 3.1 and for σ near 2.0 — close to the data-generating parameters and within the expected Monte Carlo error.

10.17.7 Interpretation

A reporting sentence: “A Gibbs sampler for the conjugate Normal–Inverse-Gamma model produced 4,000 post-burn-in draws with posterior mean $\mu = 3.10$ (95 % credible interval 2.55 to 3.65) and posterior mean $\sigma = 2.01$ (95 % CrI 1.65 to 2.49).” For pedagogical examples, also report the rejection rate (Gibbs has none) and the autocorrelation function so readers can see the per-iteration efficiency.

10.17.8 Practical Tips

- Block-update correlated parameters jointly when the conditional is available in closed form; this reduces autocorrelation dramatically compared with coordinate-wise updates.
- For non-conjugate components, embed a Metropolis step (Metropolis-within-Gibbs); JAGS does this automatically when it cannot resolve a node analytically.

- Check convergence with $\hat{R}$ across multiple chains and trace plots; a single chain cannot reveal mode-trapping.
- For continuous, smooth posteriors of moderate-to-high dimension, HMC (Stan) typically mixes far better than Gibbs because it follows the gradient instead of cycling through coordinates.
- Gibbs remains the algorithm of choice for discrete-state-space models, latent-allocation samplers, and any setting where conjugacy makes a single sweep almost free.

10.17.9 R Packages Used

Base R for hand-coded Gibbs, `rjags` and `nimble` for declarative model specification with automatic Gibbs–Metropolis dispatch, and `coda` for diagnostic plots and convergence statistics on the resulting chains.

10.17.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.17.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.18 Hamiltonian Monte Carlo

10.18.1 Introduction

Hamiltonian Monte Carlo (HMC) is the current state of the art for sampling from continuous Bayesian posteriors of moderate-to-high dimension. Where random-walk Metropolis takes small, undirected steps and Gibbs cycles through coordinates, HMC uses gradient information to propose long-distance moves that almost always land in regions of high posterior density. The No-U-Turn Sampler (NUTS), Stan's default, automates the choice of trajectory length so that users do not need to tune step counts by hand. The result is mixing rates that are orders of magnitude better than coordinate-wise samplers for posteriors with strong correlations.

10.18.2 Prerequisites

A working understanding of MCMC, basic calculus (gradients), and the relationship between log-posterior gradients and the geometry of the posterior surface. Familiarity with Stan or `brms` makes the implementation step trivial.

10.18.3 Theory

HMC introduces auxiliary momentum variables p and samples from the joint distribution $p(\theta, p) = p(\theta \mid y)\mathcal{N}(p \mid 0, M)$, where M is the mass matrix. Hamiltonian dynamics conserve the joint Hamiltonian $H(\theta, p) = -\log p(\theta \mid y) + \frac{1}{2} p^T M^{-1} p$, so trajectories simulated with leapfrog integration drift through level sets of constant log-density and explore the posterior efficiently. After L leapfrog steps with step size ϵ, the proposal is accepted with a Metropolis correction that compensates for integrator error. NUTS replaces the manual choice of L with a "no-U-turn" criterion that terminates the trajectory when it starts doubling back, eliminating the most error-prone tuning parameter of plain HMC.

10.18.4 Assumptions

A continuous, almost-everywhere differentiable log-posterior. Discrete parameters must be marginalised out. The mass matrix should approximate the posterior covariance — Stan adapts it during warm-up — and the step size must be small enough to keep integrator error in check.

10.18.5 R Implementation

```r
library(brms)

set.seed(2026)
d <- data.frame(x = rnorm(100))
d$y <- 2 + 1.5 * d$x + rnorm(100)

fit <- brm(y ~ x, data = d,
           chains = 4, iter = 2000, refresh = 0,
           control = list(adapt_delta = 0.95, max_treedepth = 10))

summary(fit)
# Diagnostics: R-hat, divergences, energy
rstan::check_hmc_diagnostics(fit$fit)
```

10.18.6 Output & Results

`summary(fit)` reports posterior summaries with $\hat{R}$, ESS, and credible intervals. `rstan::check_hmc_diagnostics()` reports divergent transitions, tree-depth saturation, energy diagnostics (E-BFMI), and is the first-line check that the geometry was navigable. Healthy HMC delivers $\hat{R}$ near 1.00, no divergent transitions, and ESS that is a substantial fraction of the post-warm-up draws.

10.18.7 Interpretation

A reporting sentence: "HMC/NUTS produced 4,000 post-warm-up draws across four chains with no divergent transitions and $\hat{R} = 1.00$ for every parameter; the slope posterior was 1.52 (95 % credible interval 1.31 to 1.74)." When divergences occur, report them and describe the remedy (reparameterisation, raising `adapt_delta`); a divergent transition is a missed region of the posterior, not a tuning curiosity.

10.18.8 Practical Tips

- For hierarchical models with funnel geometry, switch to non-centred parameterisation; `brms` does this automatically when you specify `prior(... , class = "sd", ...)`.
- Raising `adapt_delta` to 0.95 or 0.99 reduces divergences at the cost of smaller step sizes and slower sampling; reparameterisation is usually a better fix.
- Tree-depth saturation (`treedepth__` hitting `max_treedepth`) signals that NUTS could not turn around within the budget — increase `max_treedepth` or simplify the model.
- Reparameterise constrained variables onto unconstrained scales when you can: `log(sigma)` rather than `sigma > 0` keeps the leapfrog integrator stable.
- HMC scales to thousands of parameters where Gibbs or random-walk MH would be unworkable, but it cannot rescue an ill-conditioned posterior; geometry matters more than algorithm.

10.18.9 R Packages Used

`rstan` and `cmdstanr` as the HMC/NUTS backends, `brms` and `rstanarm` for high-level interfaces, `bayesplot` for diagnostic plots (`mcmc_pairs`, `mcmc_parcoord`) that visualise divergent transitions, and `posterior` for the underlying draw arrays.

10.18.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.18.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.19 Highest Posterior Density Interval

10.19.1 Introduction

The highest posterior density (HPD) interval is the narrowest range that contains a chosen probability mass — typically 95 % — under a posterior distribution. Two intervals can both cover 95 % of the posterior, yet the HPD will always be at least as short as the equal-tailed interval (ETI) and will include only those parameter values whose density is above some threshold. For unimodal posteriors the two intervals are similar; for skewed or bimodal posteriors they can disagree dramatically, and the choice between them shapes how readers perceive the result.

10.19.2 Prerequisites

A working understanding of posterior distributions and credible intervals, and access to MCMC draws (via `brms`, `rstanarm`, raw Stan, or any other sampler).

10.19.3 Theory

For a posterior $p(\theta \mid y)$, the $(1-\alpha)$ HPD region is

$$\{\theta : p(\theta \mid y) \geq h\}, \quad \text{with } h \text{ such that } P(\theta \in \text{region}) = 1-\alpha.$$

For a unimodal posterior this region is a single interval centred near the mode; for a multimodal posterior it can be a union of disjoint intervals, one per mode that exceeds the threshold. The ETI, by contrast, places the same tail probability $\alpha/2$ in each tail, so for a skewed posterior it shifts toward the heavier tail and is wider. HPD therefore favours the shape of the posterior; ETI favours interpretability and invariance to monotone transformations.

10.19.4 Assumptions

Adequate MCMC draws (after warm-up, with $\hat{R}$ near 1.00) or a closed-form posterior density. The HPD is sensitive to noisy density estimates in the tails, so very short chains can yield unstable interval endpoints.

10.19.5 R Implementation

```r
library(HDInterval)

set.seed(2026)
theta <- c(rbeta(5000, 5, 15), rbeta(5000, 15, 5))  # bimodal

# Equal-tailed (may span both modes)
quantile(theta, c(0.025, 0.975))

# HPD (disjoint regions)
hdi(theta, credMass = 0.95, allowSplit = TRUE)
```

10.19.6 Output & Results

`hdi()` returns a numeric vector of two endpoints when the HPD is a single interval, or a matrix of begin / end pairs when `allowSplit = TRUE` and the posterior is multimodal. `quantile()` returns the ETI for direct comparison; the gap between ETI and HPD widths is the most direct numerical signal of posterior asymmetry.

10.19.7 Interpretation

A reporting sentence: "The 95 % HPD interval for θ was $(0.13, 0.38) \cup (0.60, 0.89)$, reflecting the bimodal posterior; the equal-tailed interval $(0.13, 0.89)$ obscures this structure by including the low-density valley between modes." When the posterior is unimodal and roughly symmetric, the two intervals will agree and the choice does not

matter; when they disagree, prefer the one whose definition matches the inferential question — HPD for "where does the posterior concentrate?" and ETI for "what range of values is at most 2.5 % above / below the truth?"

10.19.8 Practical Tips

- Set `allowSplit = TRUE` whenever multimodality is plausible; a single interval reported across modes can be misleading.
- HPD is *not* invariant under non-linear reparameterisation: the HPD for σ^2 is not in general the square of the HPD for σ, while the ETI quantile-based interval is. Decide on the reporting scale before computing the interval.
- For bounded parameters (probabilities, variances) the HPD often touches the boundary; report this explicitly so readers know the posterior is not symmetric there.
- Use post-warm-up MCMC draws and check that the chains have mixed; HPD endpoints are noisier than means, so bigger effective sample sizes help.
- `bayestestR::hdi()` provides a similar interface and integrates with model objects from `brms`, `rstanarm`, and `posterior`.

10.19.9 R Packages Used

`HDInterval` for the canonical `hdi()` function with `allowSplit`, `bayestestR` for an integrated wrapper around model objects, and `posterior` for handling MCMC draw arrays consistently across backends.

10.19.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.19.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.20 Informative Priors

10.20.1 Introduction

An informative prior encodes genuine pre-data knowledge: a quantitative summary of previous studies, a mechanistic constraint that any plausible parameter must respect, or a structured elicitation from domain experts. Done well, informative priors stabilise inference in small samples, integrate evidence across studies, and let a single analysis quantify how new data update what was already known. Done badly, they paper over weak

data and make conclusions look more certain than they should. The discipline is to be explicit about the source, justify the choice quantitatively, and run a sensitivity analysis whenever the prior is doing meaningful work.

10.20.2 Prerequisites

A working understanding of Bayesian inference, the role of weakly informative priors as a default, and basic familiarity with meta-analytic summaries on the relevant effect scale.

10.20.3 Theory

Three classical sources of informative priors:

- **Meta-analytic priors** condense effect estimates from previous trials into a Normal (or log-Normal, or t) prior on the relevant scale: $\beta \sim \mathcal{N}(\bar{\beta}, \tau^2)$ where $\bar{\beta}$ and τ^2 come from a published random-effects meta-analysis.
- **Mechanistic priors** are derived from physical, biological, or chemical constraints; for example, drug half-lives are positive and bounded above by metabolic limits, so a log-Normal or truncated prior is appropriate.
- **Expert-elicited priors** capture structured beliefs from clinicians or subject-matter specialists, typically via the SHELF protocol or a similar formal procedure to reduce the influence of a single dominant expert.

A power prior or commensurate prior down-weights the prior contribution when it conflicts with current data — a principled way to guard against unverified historical assumptions.

10.20.4 Assumptions

The prior information is relevant to the current analysis (same population, same outcome definition, same scale) and is not derived from the same data being analysed (no double-counting). Mechanistic constraints must be physically defensible, not stylistic.

10.20.5 R Implementation

```r
library(brms)

# Meta-analytic prior: previous studies gave OR of 0.8 (95% CI 0.7 to 0.9)
# On log-OR scale: mean -0.223, SD ~ 0.05 (narrow informative prior)
set.seed(2026)
d <- data.frame(x = rnorm(100), y = rbinom(100, 1, 0.3))

fit_inf <- brm(y ~ x, data = d, family = bernoulli,
               prior = prior(normal(-0.223, 0.05), class = "b", coef = "x"),
               chains = 4, iter = 2000, refresh = 0)

# Compare with weakly informative
fit_weak <- brm(y ~ x, data = d, family = bernoulli,
                prior = prior(normal(0, 2.5), class = "b"),
                chains = 4, iter = 2000, refresh = 0)

# Posterior summaries
fixef(fit_inf); fixef(fit_weak)
```

10.20.6 Output & Results

`fixef(fit_inf)` returns the posterior mean and credible interval under the meta-analytic prior; `fixef(fit_weak)` does the same under a weakly informative prior. The two posteriors differ in proportion to the relative precision of the prior and the data: with $\mathcal{N}(-0.223, 0.05)$ the prior carries the weight of roughly $1/0.05^2 = 400$ "pseudo-observations" on the log-odds scale, dominating the small simulated dataset.

10.20.7 Interpretation

A reporting sentence: "Using a meta-analytic prior on the log-odds ratio derived from three previous trials ($\mathcal{N}(-0.223, 0.05^2)$, equivalent to roughly 400 pseudo-observations), the posterior estimate was -0.19 (95 % credible interval -0.28 to -0.10); under a weakly informative $\mathcal{N}(0, 2.5)$ prior, the posterior was -0.04 (95 % CrI -0.43 to 0.34)." Reporting both posteriors lets readers see exactly how much of the conclusion is driven by the prior.

10.20.8 Practical Tips

- Document the source of every informative prior in the methods section: cite the meta-analysis, describe the elicitation protocol, or state the mechanistic constraint and its justification.
- Sensitivity analyses are mandatory whenever an informative prior is used; report the posterior under at least one weakly informative alternative.
- Power priors with discount factor $a_0 \in [0, 1]$ scale the prior likelihood and let the analyst choose how strongly historical data should influence the current analysis.
- Commensurate priors borrow more aggressively when historical and current studies are similar and back off when they conflict; they are particularly useful in regulatory submissions.
- For elicited priors, use formal tools such as the SHELF protocol; ad-hoc elicitation tends to underestimate uncertainty and overstate consensus.

- In regulatory contexts (FDA, EMA), prior choice often requires pre-specification and agency agreement; budget time for that conversation before running the analysis.

10.20.9 R Packages Used

`brms` and `rstanarm` for placing arbitrary priors on regression parameters; `RBesT` for meta-analytic predictive priors and effective sample size calculations; `SHELF` for expert elicitation; `BayesPPD` for power-prior workflows in regulatory settings.

10.20.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.20.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.21 Jeffreys' Prior

10.21.1 Introduction

Jeffreys' prior is the classical "objective" or "reference" prior in Bayesian inference. Defined as proportional to the square root of the Fisher information, it has the elegant property of being invariant under monotone reparameterisation: the prior on a transformation $\eta = g(\theta)$ derived from the Jeffreys prior on θ equals the Jeffreys prior derived directly for η. This invariance — absent from a flat prior, which is parameterisation-dependent — is what historically made Jeffreys' rule the default attempt at a non-informative prior. In modern Bayesian practice, however, weakly informative priors have largely displaced it because they regularise small-sample inference more reliably and behave better in hierarchical models.

10.21.2 Prerequisites

A working understanding of Fisher information, the role of reparameterisation in inference, and the difference between proper and improper priors.

10.21.3 Theory

For a scalar parameter θ with likelihood $p(y \mid \theta)$, the Jeffreys prior is

$$p(\theta) \propto \sqrt{I(\theta)}, \qquad I(\theta) = -\mathbb{E}\left[\frac{\partial^2 \log p(y \mid \theta)}{\partial \theta^2}\right].$$

Standard examples:

- Binomial(n, θ): $I(\theta) \propto 1/[\theta(1-\theta)]$, so the Jeffreys prior is $\mathrm{Beta}(\frac{1}{2}, \frac{1}{2})$.
- Poisson(λ): $I(\lambda) \propto 1/\lambda$, so $p(\lambda) \propto \lambda^{-1/2}$.
- Normal mean (known variance): flat prior $p(\mu) \propto 1$.
- Normal scale (known mean): $p(\sigma) \propto 1/\sigma$.

Jeffreys priors are often improper, but the posterior is usually proper as long as the sample size is sufficient. For multivariate parameters, $I(\theta)$ is a matrix and the prior is $\sqrt{|I(\theta)|}$.

10.21.4 Assumptions

A regular likelihood (twice differentiable in θ, with finite Fisher information across the parameter space). Boundary problems and parameter-dimension growth complicate the construction; reference priors (Bernardo) generalise Jeffreys' rule for hierarchical and high-dimensional cases.

10.21.5 R Implementation

```r
# Binomial: posterior under Jeffreys is Beta(1/2 + x, 1/2 + n - x)
n ← 10; x ← 8
post_a ← 0.5 + x
post_b ← 0.5 + n - x
c(mean = post_a / (post_a + post_b),
  ci95 = qbeta(c(0.025, 0.975), post_a, post_b))

# Compare with uniform Beta(1, 1) and Beta(2, 2)
data.frame(
  prior = c("Uniform(0,1)", "Jeffreys", "Beta(2,2)"),
  post_mean = c((1 + x)/(2 + n), (0.5 + x)/(1 + n), (2 + x)/(4 + n))
)
```

10.21.6 Output & Results

Closed-form summaries under three priors. With 8 successes in 10 trials the Jeffreys posterior mean is approximately 0.77 (vs. 0.75 under the uniform $\mathrm{Beta}(1, 1)$ and 0.71 under the more informative $\mathrm{Beta}(2, 2)$), and the 95 % credible interval is roughly $(0.47, 0.94)$. The comparison highlights how mildly Jeffreys differs from a flat prior in this setting.

10.21.7 Interpretation

A reporting sentence: "Jeffreys' prior $\mathrm{Beta}(0.5, 0.5)$ on the success probability combined with 8 successes in 10 trials yielded a posterior mean of 0.77 (95 % credible interval 0.47 to 0.94); the result is essentially equivalent to a uniform prior at this sample size, justifying

the reference-prior framing." When Jeffreys differs materially from a weakly informative prior, the prior is doing real work and you should be explicit about why you chose it.

10.21.8 Practical Tips

- The invariance property is mathematically attractive but in practice rarely affects applied conclusions; weakly informative priors are a more flexible default in modern workflows.
- Jeffreys priors can behave badly in hierarchical models — the multivariate version often produces improper posteriors when variance components are at the boundary. Reach for weakly informative half-Cauchy or half-Normal priors instead.
- Always check that the resulting posterior is proper, especially when the prior itself is improper; a back-of-the-envelope check is to confirm that the posterior integrates to a finite quantity.
- Reference priors (Bernardo) generalise Jeffreys' rule for ordered parameters of interest and nuisance components; they are the modern theoretical successor when an objective prior is genuinely required.
- For pedagogical purposes, Jeffreys' prior is invaluable because it makes invariance under reparameterisation a hands-on exercise; for applied analyses, prefer weakly informative defaults unless invariance is a publication requirement.

10.21.9 R Packages Used

Base R suffices for closed-form Jeffreys updates in conjugate models; `BayesianTools` and `mcmc` allow Jeffreys priors via custom log-prior functions; `bayestestR::point_estimate()` and `HDInterval::hdi()` deliver tidy summaries on the resulting posteriors.

10.21.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.21.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.22 Bayesian Linear Regression

10.22.1 Introduction

Bayesian linear regression keeps the same linear predictor and Normal residual structure as ordinary least squares but treats every parameter as a random variable with a prior. Inference returns a joint posterior over coefficients and the residual scale rather than a point estimate plus standard errors. The practical gain is direct: credible intervals carry the probability statement that students often (incorrectly) attribute to confidence intervals, predictions inherit propagated uncertainty automatically, and small-sample analyses no longer rely on asymptotic approximations of t distributions.

10.22.2 Prerequisites

A working understanding of multiple linear regression, comfort with the idea of a likelihood, and a basic sense of priors. A Stan-capable backend (`rstan` or `cmdstanr`) is needed because `brms` compiles models on demand; `rstanarm` ships with pre-compiled models if compile time is a concern.

10.22.3 Theory

The model is

$$y_i = X_i\beta + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, \sigma^2),$$

with priors $\beta_j \sim \mathcal{N}(0, \tau_j)$ and $\sigma \sim \text{Exponential}(\lambda)$ or a half-t. With weakly informative priors and reasonable sample size, the posterior mean is close to the OLS estimate while small-n inference is regularised toward the prior. The default sampler is Hamiltonian Monte Carlo with the No-U-Turn termination criterion (NUTS); convergence is assessed through $\hat{R}$, bulk and tail effective sample size, and divergent-transition counts.

10.22.4 Assumptions

Linearity in the linear predictor, additive Normal residuals with constant variance, independence of observations, and proper priors. Convergence diagnostics are part of the model assumptions in practice — a chain that has not mixed cannot deliver a credible posterior summary.

10.22.5 R Implementation

```r
library(brms)

set.seed(2026)
d ← data.frame(x = rnorm(100), y = 2 + 1.5 * rnorm(100))

fit ← brm(y ~ x, data = d,
          prior = prior(normal(0, 2.5), class = "b") +
                  prior(normal(0, 5),   class = "Intercept") +
                  prior(exponential(1), class = "sigma"),
          chains = 4, iter = 2000, refresh = 0)

summary(fit)
plot(fit)
```

10.22.6 Output & Results

`summary(fit)` returns posterior mean, standard error of the mean, posterior standard deviation, 95 % credible interval, $\hat{R}$, and bulk and tail effective sample size for every parameter. `plot(fit)` shows trace plots (one line per chain) alongside marginal density plots; well-mixed chains overlap throughout warm-up and produce indistinguishable density plots.

10.22.7 Interpretation

A reporting sentence: "A Bayesian linear regression with weakly informative priors estimated the slope at 1.53 (95 % CrI 1.23 to 1.84) and the residual scale at 1.02 (95 % CrI 0.90 to 1.17); $\hat{R} = 1.00$ for all parameters, with $n_{\text{eff}} > 3{,}500$." Always report a convergence statistic alongside the interval — without it, a reader cannot tell whether the posterior summary is trustworthy.

10.22.8 Practical Tips

- Standardise predictors before fitting so a single $\mathcal{N}(0, 2.5)$ prior on slopes is sensible across columns.
- Run a prior predictive check (`sample_prior = "only"`) and look at the implied range of `y`; if it spans implausible values, tighten the prior before fitting to the data.
- For interactions or polynomials, use `bs()` or `poly()` inside the formula but check that the prior scale still makes sense after the transformation.
- Use `bayesplot::pp_check(fit)` to overlay replicated datasets on the observed outcome; systematic deviation in the tails signals the wrong family.
- Compare nested models with `loo_compare(loo(fit1), loo(fit2))` rather than F-tests; the difference in expected log predictive density and its standard error is a richer summary than a p-value.
- For non-Gaussian outcomes, change `family = gaussian` to `bernoulli()`, `poisson()`, `student()`, or any other supported family — the prior structure carries over unchanged.

10.22.9 R Packages Used

`brms` for formula-based Bayesian regression, `rstan` or `cmdstanr` as the sampling backend, `bayesplot` for diagnostic and posterior predictive plots, and `loo` for cross-validation.

10.22.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.22.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.23 Bayesian Logistic Regression

10.23.1 Introduction

Bayesian logistic regression keeps the same Bernoulli likelihood and logit link as the maximum-likelihood version but replaces the point estimate of each coefficient with a full posterior distribution. The biggest practical pay-off is robustness in small samples or in the presence of complete or quasi-complete separation, where MLE drifts to $\pm\infty$ but a weakly informative prior pulls the estimate to a finite, defensible value. The posterior also feeds directly into propagated uncertainty for predicted probabilities, an awkward computation in the frequentist world.

10.23.2 Prerequisites

Familiarity with classical logistic regression and odds ratios, an idea of what a prior and a posterior are, and access to a Stan-capable backend (`rstan` or `cmdstanr`).

10.23.3 Theory

The model is

$$y_i \sim \text{Bernoulli}\big(\text{logit}^{-1}(X_i\beta)\big),$$

with priors $\beta \sim \mathcal{N}(0, \sigma_\beta^2)$ on the regression coefficients and a separate, often wider prior on the intercept because the average log-odds of success can lie outside the typical predictor scale. A $\mathcal{N}(0, 2.5)$ prior on standardised slopes is the default in `rstanarm` and a

sensible starting point in `brms`. Under separation the likelihood becomes monotone in β, and any proper prior — even a very wide one — yields a well-defined posterior with finite credible intervals; this is the Bayesian analogue of Firth's penalised likelihood.

10.23.4 Assumptions

Independent Bernoulli observations conditional on covariates, a logit link that is a reasonable approximation to the true response surface, and valid (proper) priors. As in MLE logistic regression, the linear predictor should not contain redundant columns.

10.23.5 R Implementation

```
library(brms)

set.seed(2026)
d ← data.frame(x = rnorm(200))
d$y ← rbinom(200, 1, plogis(-1 + 1.2 * d$x))

fit ← brm(y ~ x, data = d, family = bernoulli,
          prior = prior(normal(0, 2.5), class = "b") +
                  prior(normal(0, 5), class = "Intercept"),
          chains = 4, iter = 2000, refresh = 0)
summary(fit)

# Odds ratios
exp(fixef(fit))
```

10.23.6 Output & Results

`summary(fit)` reports posterior mean, standard error, and 95 % CrI on the log-odds scale together with $\hat{R}$ and effective sample size; `exp(fixef(fit))` returns the same summaries on the odds-ratio scale. `posterior_predict()` delivers Bernoulli replicates that feed `pp_check()` to test calibration.

10.23.7 Interpretation

A reporting sentence: "A Bayesian logistic regression with weakly informative $\mathcal{N}(0, 2.5)$ priors estimated the odds ratio for x at 3.4 (95 % CrI 2.5 to 4.5); under the model, a one-standard-deviation increase in x triples the odds of success." When predicted probabilities are required, summarise `posterior_epred()` rather than transforming `fixef()` by hand — only the former propagates the joint uncertainty in intercept and slope.

10.23.8 Practical Tips

- For binary regression with rare outcomes, prefer a $\mathcal{N}(0, 1)$ prior on standardised slopes; the default $\mathcal{N}(0, 2.5)$ allows odds ratios up to about $\exp(5) \approx 150$, which can be implausibly extreme.
- Compare predictive performance with `loo()` rather than likelihood-ratio tests; LOO is well-defined under proper priors and gives Pareto-k diagnostics for influential

points.

- Hierarchical extensions are a one-line addition: `y ~ x + (1 | site)` lets baseline log-odds vary across sites and shrinks small-cluster estimates toward the population mean.
- For severe separation, fit with `prior(normal(0, 1.5), class = "b")` rather than ML's Firth correction; the Bayesian solution is a single line of code and propagates uncertainty consistently.
- Always inspect `pp_check(fit, type = "bars")` for binary outcomes; deviations between observed and replicated proportions reveal calibration problems that summary tables hide.

10.23.9 R Packages Used

`brms` for model specification, `bayesplot` for posterior predictive plots, `loo` for cross-validation and model comparison, and `tidybayes` if you want tidy tibbles of posterior draws for downstream plotting.

10.23.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.23.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.24 Leave-One-Out Cross-Validation

10.24.1 Introduction

Leave-one-out cross-validation (LOO-CV) is the gold standard for estimating out-of-sample predictive performance, but exact LOO requires refitting the model once per observation — prohibitive for any Bayesian model that takes more than a few seconds to sample. Pareto-Smoothed Importance Sampling LOO (PSIS-LOO), introduced by Vehtari, Gelman, and Gabry, approximates exact LOO from a single full-data posterior using importance weights, while providing per-observation diagnostics that flag where the approximation breaks down. It has become the standard model-comparison tool for `brms`, `rstanarm`, and raw Stan workflows.

10.24.2 Prerequisites

A working understanding of cross-validation, importance sampling, and the role of the log pointwise predictive density (elpd) in measuring predictive performance.

10.24.3 Theory

The expected log predictive density under leave-one-out is

$$\mathrm{elpd}_{\mathrm{loo}} = \sum_{i=1}^{n} \log p(y_i \mid y_{-i}),$$

where y_{-i} is the data with observation i removed. PSIS approximates each $p(y_i \mid y_{-i})$ via importance weights $w_i^{(s)} \propto 1/p(y_i \mid \theta^{(s)})$ applied to the full-data posterior draws $\theta^{(s)}$. To stabilise the right tail of the importance weights, PSIS fits a generalised Pareto distribution to the largest weights and replaces them with order-statistic estimates. The shape parameter $\hat{k}$ of that fit acts as a per-observation diagnostic: $\hat{k} < 0.5$ is excellent, $0.5 \le \hat{k} < 0.7$ acceptable, and $\hat{k} > 0.7$ indicates that PSIS-LOO is unreliable for that point.

10.24.4 Assumptions

Exchangeable observations and a posterior with sufficient effective sample size to estimate the importance-weight tail; for hierarchical or temporal data, replace standard LOO with leave-one-cluster-out or leave-future-out variants whose theoretical properties match the dependence structure.

10.24.5 R Implementation

```
library(loo); library(brms)

data(mtcars)
fit1 ← brm(mpg ~ wt, data = mtcars, chains = 2, iter = 2000, refresh = 0)
fit2 ← brm(mpg ~ wt + hp, data = mtcars, chains = 2, iter = 2000, refresh = 0)

loo1 ← loo(fit1)
loo2 ← loo(fit2)

loo_compare(loo1, loo2)

# k-hat diagnostic
plot(loo2)
```

10.24.6 Output & Results

`loo()` returns the elpd, the effective parameter count p_{loo}, and a vector of per-observation $\hat{k}$ values. `loo_compare()` aligns two `loo` objects and returns the elpd difference with its standard error, ordered from best to worst. `plot(loo)` displays $\hat{k}$ values against observation index, immediately revealing influential points.

10.24.7 Interpretation

A reporting sentence: "PSIS-LOO favoured the two-predictor model with $\Delta\text{elpd} = 6$ (SE 2.3); all $\hat{k} < 0.5$, so the importance-sampling approximation is reliable, and no observation needs an exact refit." When some $\hat{k}$ values exceed 0.7, rerun with `loo(fit, reloo = TRUE)` to refit the model on the affected leave-out folds; the elpd estimate is then a hybrid of PSIS for safe folds and exact fits for the troublesome ones.

10.24.8 Practical Tips

- Check $\hat{k}$ values *before* interpreting an elpd comparison; a single high-leverage observation can dominate the ranking.
- For time-series, use leave-future-out CV via `loo::loo_subsample()` or `loo_moment_match()`; standard LOO leaks future information into past folds and biases elpd downward.
- For hierarchical models with few clusters, leave-one-cluster-out CV is the appropriate generalisation: it answers the prediction question that matters in practice ("how would the model perform on a new cluster?").
- Differences in elpd are interpretable in absolute terms: $\Delta\text{elpd} > 4 \times \text{SE}$ is decisive; $|\Delta\text{elpd}| < 2 \times \text{SE}$ is inconclusive.
- LOO is well-defined for non-nested models, unlike likelihood-ratio tests, which makes it the natural tool when comparing models that differ in family, link, or predictor set.

10.24.9 R Packages Used

`loo` for `loo()`, `loo_compare()`, `loo_subsample()`, and `loo_moment_match()`, `brms` and `rstanarm` for fits that expose pointwise log-likelihoods automatically, and `bayesplot` for posterior predictive plots that complement the elpd-based comparison.

10.24.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.24.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.25 Metropolis-Hastings

10.25.1 Introduction

The Metropolis-Hastings algorithm is the original general-purpose MCMC sampler and remains the conceptual reference against which all subsequent algorithms are explained. Given only an unnormalised target density, it produces a Markov chain whose stationary distribution is that target by repeatedly proposing a candidate value and accepting or rejecting it based on a simple density-ratio rule. The trade-off is universality versus efficiency: MH works for any target one can evaluate, but tuning proposals for high-dimensional or strongly correlated posteriors becomes painful, which is why HMC has displaced random-walk MH for most modern Bayesian software.

10.25.2 Prerequisites

A working understanding of Markov chains, conditional probability, and the role of the unnormalised posterior in Bayesian inference. Familiarity with R's distribution functions (`rnorm`, `dunif`) is needed for the example.

10.25.3 Theory

Given the current state θ, one iteration of MH proceeds as:

1. Propose θ^* from a proposal kernel $q(\theta^* \mid \theta)$.
2. Compute the acceptance probability $\alpha = \min\left(1, \dfrac{p(\theta^*)q(\theta \mid \theta^*)}{p(\theta)q(\theta^* \mid \theta)}\right)$.
3. With probability α set the next state to θ^*; otherwise keep θ.

For a symmetric proposal (Gaussian random walk centred at the current state) the proposal terms cancel and the acceptance ratio is simply the ratio of target densities. Detailed balance ensures the chain has the target as its invariant distribution; ergodicity (irreducibility plus aperiodicity) ensures convergence regardless of the starting value.

10.25.4 Assumptions

The unnormalised target density is computable everywhere in the support. The proposal kernel must give the chain a positive probability of reaching every part of the support; otherwise the chain is reducible and never explores the full posterior.

10.25.5 R Implementation

```r
set.seed(2026)

# Target: Beta(4, 8) (unnormalised density is fine)
log_target ← function(x) {
  if (x ⩽ 0 || x ⩾ 1) return(-Inf)
  (4 - 1) * log(x) + (8 - 1) * log(1 - x)
}

n_iter ← 10000
theta ← numeric(n_iter)
theta[1] ← 0.5
sigma_prop ← 0.1

for (i in 2:n_iter) {
  prop ← rnorm(1, theta[i-1], sigma_prop)
  log_ratio ← log_target(prop) - log_target(theta[i-1])
  if (log(runif(1)) < log_ratio) theta[i] ← prop
  else theta[i] ← theta[i-1]
}

# Diagnostic: trace + histogram
par(mfrow = c(1, 2))
plot(theta, type = "l")
hist(theta[-(1:1000)], breaks = 50, freq = FALSE, main = "Posterior samples")
curve(dbeta(x, 4, 8), add = TRUE, col = "red", lwd = 2)
par(mfrow = c(1, 1))

mean(theta[-(1:1000)])     # posterior mean
```

10.25.6 Output & Results

A 10,000-step chain whose post-burn-in histogram aligns with the analytic Beta(4, 8) density. The trace plot, the histogram, and the posterior mean (around $4/(4+8) = 0.33$) confirm convergence; the autocorrelation function tells you how many independent draws the chain is worth.

10.25.7 Interpretation

A reporting sentence: “A random-walk Metropolis-Hastings sampler with proposal $\mathcal{N}(\theta^{(t)}, 0.1^2)$ produced 9,000 post-burn-in draws; the histogram tracked the analytic Beta(4, 8) target, and the posterior mean of 0.33 matched the closed-form value to within Monte Carlo error.” Quote the acceptance rate so readers can judge whether the proposal was tuned reasonably.

10.25.8 Practical Tips

- Tune the proposal scale for an acceptance rate near 25–40 % for one-dimensional random walks (Roberts–Gelman–Gilks rule); for higher dimensions, lower rates are optimal.
- Discard a burn-in of 1,000–5,000 iterations and inspect the trace plot for visible drift before computing summaries.

- When parameters are correlated, a coordinate-wise random walk has poor mixing; switch to block updates with a multivariate Normal proposal whose covariance approximates the posterior, or move to HMC.
- Use $\hat{R}$ across multiple chains rather than relying on a single trace; a single chain cannot reveal mode-trapping.
- For tall posteriors with many parameters, modern HMC samplers (Stan, NUTS) almost always mix dramatically better than basic MH; reach for MH only when the target is hard to differentiate or has discrete components.

10.25.9 R Packages Used

Base R is enough for hand-coded MH; `MCMCpack` and `mcmc` provide tuned implementations for canonical models, and `coda` contributes diagnostic plots and convergence statistics applicable to MH chains from any source.

10.25.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.25.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.26 The Normal-Normal Model

10.26.1 Introduction

The Normal-Normal model is the simplest example of conjugate Bayesian updating for a continuous parameter: a Normal prior on the mean, a Normal likelihood with known variance, and a Normal posterior obtained by a transparent precision-weighted average. It is the classical pedagogical example, and despite its simplicity it sits at the heart of every hierarchical model that contains a Normal level — random-effects meta-analysis, partial pooling in mixed models, and the mean structure of Gaussian processes all rely on the same closed-form update.

10.26.2 Prerequisites

A working understanding of the Normal distribution, the concept of conjugacy, and the difference between variance and precision (inverse variance). Basic algebra is enough to follow the derivation.

10.26.3 Theory

With prior $\mu \sim \mathcal{N}(\mu_0, \tau_0^2)$ and likelihood $y_i \sim \mathcal{N}(\mu, \sigma^2)$ for $i = 1, \ldots, n$ with known σ, the posterior is

$$\mu \mid y \sim \mathcal{N}(\mu_n, \tau_n^2),$$

where the posterior precision is the sum of the prior precision and the data precision,

$$\tau_n^{-2} = \tau_0^{-2} + n/\sigma^2,$$

and the posterior mean is the precision-weighted average

$$\mu_n = \tau_n^2 \left(\tau_0^{-2} \mu_0 + n \bar{y} / \sigma^2 \right).$$

The posterior mean lies between the prior mean and the sample mean, with weight that depends on relative precisions. The precision-additive structure makes the prior's "pseudo-sample-size" interpretable: τ_0^2 corresponds to a prior sample of size σ^2/τ_0^2.

10.26.4 Assumptions

Known σ^2 — so the only unknown is the mean — and a Normal prior. If the variance is also unknown, the conjugate setup uses a Normal-Inverse-Gamma prior and the posterior on μ is a t distribution.

10.26.5 R Implementation

```r
# Prior: mu ~ N(100, 10^2); data: xbar = 105, sigma = 15, n = 20
mu0 <- 100; tau0 <- 10
xbar <- 105; sigma <- 15; n <- 20

prec_post <- 1/tau0^2 + n/sigma^2
tau_post  <- 1/sqrt(prec_post)
mu_post   <- tau_post^2 * (mu0/tau0^2 + n*xbar/sigma^2)

c(posterior_mean = mu_post,
  posterior_sd   = tau_post,
  ci95 = mu_post + c(-1, 1) * 1.96 * tau_post)
```

10.26.6 Output & Results

Posterior mean approximately 104.2 with posterior standard deviation 2.4 and 95 % credible interval roughly $(99.6, 108.9)$. The shrinkage from the sample mean of 105 toward the prior mean of 100 is small because the data precision ($n/\sigma^2 \approx 0.089$) is much larger than the prior precision ($1/\tau_0^2 = 0.01$).

10.26.7 Interpretation

A reporting sentence: "Combining a $\mathcal{N}(100, 10^2)$ prior with 20 observations of sample mean 105 (known $\sigma = 15$) yielded a posterior mean of 104.2 (95 % credible interval 99.6 to 108.9); the prior contributed the equivalent of approximately 2.25 additional observations."

The "equivalent observations" framing is one of the most intuitive ways to communicate prior strength to non-Bayesians.

10.26.8 Practical Tips

- As $n \to \infty$, the posterior mean tends to the sample mean and the posterior variance tends to σ^2/n; the prior's influence vanishes asymptotically but can dominate in small samples.
- When σ is unknown, switch to the Normal–Inverse-Gamma conjugate pair; the marginal posterior on μ then has t tails.
- The precision-form update generalises directly to multivariate Normals with covariance matrices in place of variances; everything that was scalar becomes matrix.
- Use this model as a sanity check for software: any general-purpose Bayesian engine (Stan, JAGS, `brms`) should reproduce the closed-form posterior to within Monte Carlo error.
- Hierarchical mean structures are built from this update at every level, which is why understanding it deeply pays dividends in mixed models and meta-analysis.

10.26.9 R Packages Used

Base R is sufficient because the update is closed-form (`dnorm`, `pnorm`, `qnorm`); for didactic plotting compare prior, likelihood, and posterior with `ggplot2`, and for general Bayesian alternatives any of `brms`, `rstanarm`, or `BayesFactor` reproduce the same answer.

10.26.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.26.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.27 Posterior Mean and Variance

10.27.1 Introduction

A Bayesian analysis returns a full posterior distribution, but applied work eventually demands a number to put in a sentence: a coefficient, a probability, an effect size. The posterior mean, median, and mode are the three most common point summaries, each

motivated by a different decision-theoretic loss. The posterior variance and standard deviation quantify the uncertainty around the chosen point. Treating the posterior as the headline and the point estimate as a one-line summary derived from it is a more honest reporting style than collapsing the posterior to a single number first and worrying about uncertainty later.

10.27.2 Prerequisites

A working understanding of Bayes' theorem and the difference between a posterior distribution and a frequentist sampling distribution; basic familiarity with MCMC draws or a closed-form posterior.

10.27.3 Theory

For a posterior $p(\theta \mid y)$, the standard summaries are:

- **Posterior mean**: $\mathbb{E}[\theta \mid y]$ minimises expected squared-error loss; it is the Bayes estimator under quadratic loss and the natural target of MCMC averaging.
- **Posterior median**: minimises expected absolute-error loss; preferred for skewed posteriors because it does not chase a long tail.
- **Posterior mode (MAP)**: maximises $p(\theta \mid y)$ and often coincides with a penalised maximum-likelihood estimate when the prior is interpreted as a penalty.
- **Posterior variance**: $\mathrm{Var}(\theta \mid y)$ measures spread; the standard deviation $\sqrt{\mathrm{Var}}$ is on the original scale of θ.

The mean and variance always exist for proper integrable posteriors; the mode may not be unique under multimodal posteriors and the median is unique only for continuous distributions.

10.27.4 Assumptions

A proper posterior with finite mean and variance, and either closed-form expressions or sufficient MCMC draws. For heavy-tailed posteriors, the mean and variance can be unstable estimators even when they exist mathematically, so report effective sample sizes alongside the summaries.

10.27.5 R Implementation

```r
set.seed(2026)

# Simulated samples from a posterior (e.g., from MCMC)
theta ← rbeta(5000, shape1 = 10, shape2 = 4)

c(mean = mean(theta),
  median = median(theta),
  mode = density(theta)$x[which.max(density(theta)$y)],
  sd = sd(theta),
  var = var(theta))

# Credible interval
quantile(theta, c(0.025, 0.975))
```

10.27.6 Output & Results

Five scalar summaries and a 95 % equal-tailed credible interval. For a Beta(10, 4) posterior the mean is approximately $10/14 \approx 0.71$, the standard deviation is around 0.11, and the 95 % interval excludes the lower decile of the distribution, communicating both the central tendency and the spread.

10.27.7 Interpretation

A reporting sentence: “The posterior probability had mean 0.71 (posterior $\mathrm{SD} = 0.11$, 95 % credible interval 0.47 to 0.89), placing high credibility on values above 0.5 and effectively excluding the prior-favoured region near 0.3.” When the posterior is roughly symmetric the choice of mean versus median rarely matters; when it is skewed, report both and let readers see the asymmetry.

10.27.8 Practical Tips

- For symmetric posteriors, mean and median are essentially equal; choose whichever is conventional in your field.
- For posteriors over bounded parameters (probabilities, variances, r-coefficients), the mode can fall on the boundary; the mean stays in the interior and is a safer default.
- Posterior standard deviation is *not* the same as Monte Carlo standard error — the former is genuine uncertainty in θ, the latter is sampling noise from finite MCMC iterations. Report MCSE separately when it is appreciable relative to the posterior SD.
- Avoid reporting a “Bayesian point estimate” without specifying which one; default conventions differ across software and journals.
- For predictions, summarise the posterior predictive distribution rather than transforming a point estimate of θ; only the former propagates parameter uncertainty.

10.27.9 R Packages Used

Base R suffices for raw summaries (`mean`, `median`, `var`, `sd`, `quantile`), `posterior::summarise_draws()` provides tidy tibbles with Monte Carlo standard errors, and `bayestestR::point_estimate()` exposes mean / median / MAP through one consistent interface for `brms` and `rstanarm` model objects.

10.27.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.27.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.28 Posterior Predictive Checks

10.28.1 Introduction

A posterior predictive check is the Bayesian answer to "does this model produce data that look like ours?" After fitting, you draw replicated datasets from the model — each one using a different posterior parameter draw — and compare them to the observed data through summary statistics, density overlays, or grouped subsets. Systematic discrepancies between observed and replicated quantities are a model criticism: they tell you which feature of the data the current specification fails to capture, which is far more actionable than a single goodness-of-fit p-value.

10.28.2 Prerequisites

A working understanding of the posterior predictive distribution and access to MCMC draws or any object that exposes a `posterior_predict()` method (`brms`, `rstanarm`, raw Stan).

10.28.3 Theory

The posterior predictive distribution is

$$p(\tilde{y} \mid y) = \int p(\tilde{y} \mid \theta) p(\theta \mid y) \, \mathrm{d}\theta,$$

approximated by drawing θ from the posterior and then $\tilde{y}$ from the likelihood given that θ. A posterior predictive check chooses a test statistic $T(y)$ — the mean, the standard deviation, the proportion of zeros, the maximum, the skew, or any function of the data — and compares its observed value to the distribution of $T(\tilde{y})$ across replicates. If the observed statistic falls in the bulk of the replicate distribution, the model captures that aspect of the data; if it falls in a tail, the model misses it.

10.28.4 Assumptions

A converged posterior with adequate effective sample size, since each replicate dataset depends on a posterior draw. PPCs do not require held-out data and are therefore not a measure of generalisation; they are a within-sample diagnostic.

10.28.5 R Implementation

```r
library(brms); library(bayesplot)

set.seed(2026)
d ← data.frame(y = rnorm(200, 3, 1.5))

fit ← brm(y ~ 1, data = d, family = gaussian,
          chains = 2, iter = 2000, refresh = 0)

# Density overlay
pp_check(fit)

# Test-statistic check: SD of simulated vs observed
pp_check(fit, type = "stat", stat = "sd")

# Interval check
pp_check(fit, type = "intervals")
```

10.28.6 Output & Results

`pp_check(fit)` returns a `ggplot` overlaying the empirical density of `y` on top of densities for several replicate datasets. `type = "stat"` plots the distribution of the chosen statistic across replicates with the observed value as a vertical line. `type = "intervals"` shows the 50 % and 90 % posterior predictive intervals point-by-point and is the most informative check for time-series and ordered data.

10.28.7 Interpretation

A reporting sentence: “Posterior predictive density overlays showed the observed distribution falling well within the cloud of replicates; the standard-deviation check placed the observed value near the centre of the replicate distribution ($p_{\text{ppc}} = 0.48$), indicating that the Gaussian model captures both location and dispersion adequately.” When a check fails, name the feature: “skew of replicates was systematically lower than the observed skew”, which directly suggests a heavy-tailed family or a transformation.

10.28.8 Practical Tips

- A density overlay is the broadest check; supplement it with tests on specific statistics (skew, kurtosis, proportion of zeros) that target likely misspecifications.
- Group-wise PPCs (`pp_check(fit, type = "stat_grouped", group = "site")`) catch problems concentrated in specific subgroups that pooled checks would mask.
- For count outcomes, `type = "rootogram"` is more informative than density overlays because it highlights overdispersion and zero-inflation directly.
- A passing PPC does not guarantee good predictions on new data; pair PPCs with `loo()` or held-out evaluation when generalisation matters.
- When PPCs reveal misspecification, the fix is almost always a richer family (heavy tails, dispersion, mixtures) before a richer mean structure.

10.28.9 R Packages Used

`bayesplot` for the `pp_check()` and `ppc_*` graphics, `brms` (or `rstanarm`) as a backend that exposes `posterior_predict()` directly, and `posterior` for handling draw arrays when building custom PPCs by hand.

10.28.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.28.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.29 The Posterior Predictive Distribution

10.29.1 Introduction

The posterior predictive distribution (PPD) is the Bayesian answer to the question "what range of values can a future observation plausibly take?" It accounts for two sources of uncertainty simultaneously: the sampling variability that would persist even if the parameters were known, and the residual parameter uncertainty quantified by the posterior. A plug-in prediction that uses only the posterior mean throws away the second source and produces intervals that are too narrow — a frequent and consequential mistake in applied work.

10.29.2 Prerequisites

A working understanding of posterior distributions, the difference between parameter inference and observation prediction, and Monte Carlo sampling from MCMC draws.

10.29.3 Theory

For data y and a prospective new observation $\tilde{y}$, the posterior predictive distribution is

$$p(\tilde{y} \mid y) = \int p(\tilde{y} \mid \theta) p(\theta \mid y) \, \mathrm{d}\theta.$$

The Monte Carlo approximation is direct: for each posterior draw $\theta^{(s)}$, sample $\tilde{y}^{(s)} \sim p(\tilde{y} \mid \theta^{(s)})$. The collection $\{\tilde{y}^{(s)}\}_{s=1}^{S}$ is itself a sample from the PPD, and any quantile, interval, or expectation of interest comes from it. Because the integral over θ is automatic in this formulation, the resulting prediction interval is wider than the corresponding interval based on a plug-in $\hat{\theta}$, and that extra width is the propagated parameter uncertainty.

10.29.4 Assumptions

Valid posterior draws (converged chains, adequate ESS) and the assumed data-generating model $p(\tilde{y} \mid \theta)$ for the new observation. If the new observation is from a different population or covariate range, the PPD inherits the model's extrapolation behaviour and may be unrealistic.

10.29.5 R Implementation

```r
library(brms)

set.seed(2026)
n ← 100
d ← data.frame(x = rnorm(n), y = NA)
d$y ← 2 + 1.5 * d$x + rnorm(n)

fit ← brm(y ~ x, data = d, chains = 2, iter = 2000, refresh = 0)

# Posterior predictive samples
pp ← posterior_predict(fit, newdata = data.frame(x = 0:3))
apply(pp, 2, quantile, c(0.025, 0.5, 0.975))
```

10.29.6 Output & Results

`posterior_predict()` returns a matrix with one row per posterior draw and one column per `newdata` row. Summarising columnwise gives the posterior predictive median and 95 % interval at each new x. The resulting interval combines the residual standard deviation with the parameter uncertainty inherited from the posterior of β_0 and β_1.

10.29.7 Interpretation

A reporting sentence: "At $x = 2$ the posterior predictive 95 % interval for a new observation is 3.1 to 6.8, with median 5.0; the corresponding interval for the conditional mean is 4.7 to 5.3, illustrating that residual variability dominates the predictive uncertainty in this model." Distinguishing the predictive interval from the credible interval on the mean is essential — they answer different questions and the former is always wider.

10.29.8 Practical Tips

- Use `posterior_epred()` for the conditional-mean credible interval (no residual variability), `posterior_predict()` for the full predictive distribution including residual variation, and `posterior_linpred()` if you need the linear predictor before any link transformation.
- Pair PPD draws with `pp_check(fit)` to confirm that the model captures the marginal distribution of the data; a misspecified family will produce predictive intervals that are too tight or too wide where they matter.
- For binary, count, or ordinal outcomes, the PPD is a discrete distribution; visualise with rootograms or bar charts rather than density curves.
- When predicting beyond the observed predictor range, report posterior predictive intervals separately for in-sample and out-of-sample x; readers should see where extrapolation begins.
- Always summarise the PPD on the original scale of the outcome; transforming a credible interval on the linear predictor and adding back-of-the-envelope residual SD is rarely correct.

10.29.9 R Packages Used

`brms` and `rstanarm` for `posterior_predict()`, `posterior_epred()`, and `posterior_linpred()`; `tidybayes::predicted_draws()` for tidy long-format predictive draws; `bayesplot` for `pp_check()` and other PPD diagnostic plots.

10.29.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.29.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.30 Prior Specification

10.30.1 Introduction

Choosing a prior is the most consequential modelling decision in any Bayesian analysis after the choice of likelihood. The prior shapes the posterior strongly when data are sparse, almost imperceptibly when data are abundant, and unpredictably in the middle ground that most applied work occupies. Three categories cover most needs: flat or improper priors, weakly informative priors that regularise without imposing substantive content, and informative priors that incorporate genuine pre-data knowledge. Modern Bayesian practice has converged on weakly informative priors as the default because they stabilise sampling, regularise small-sample inference, and remain agnostic enough that the data dominate the posterior whenever they are informative.

10.30.2 Prerequisites

A working understanding of Bayes' theorem, the role of the prior in posterior updating, and basic familiarity with parameter scales (standardised vs. raw, log-scale vs. linear).

10.30.3 Theory

Flat / improper priors assign constant density across the support. They sometimes appear intuitive but can yield improper posteriors in hierarchical models or near boundary points, and they are not invariant under reparameterisation.

Weakly informative priors are proper distributions broad enough to admit any plausible value while ruling out absurd ones. Standard choices are $\mathcal{N}(0, 2.5)$ for standardised slopes, $\mathcal{N}(0, 10)$ for intercepts, $\text{half-Cauchy}(0, 5)$ or $\text{half-Normal}(0, 2.5)$ for variance components, and $\text{LKJ}(2)$ for correlation matrices. They regularise estimates without injecting domain claims.

Informative priors carry substantive content from previous studies, theoretical bounds, or expert elicitation. They are appropriate when the information is defensible and reproducible, and inappropriate when chosen post-hoc to push the posterior somewhere convenient.

The **prior predictive check** — simulating from the prior, propagating through the likelihood, and inspecting the implied range of `y` — is the single most useful sanity check. If the prior implies blood pressures of 1,000 mm Hg or odds ratios of 10^6, it is too wide.

10.30.4 Assumptions

The prior reflects pre-data information honestly, the likelihood is correctly specified, and the prior is proper unless the resulting posterior has been verified to be proper analytically.

10.30.5 R Implementation

```r
library(brms)

# Logistic regression with weakly informative priors
priors <- prior(normal(0, 2.5), class = "b") +
          prior(normal(0, 10),  class = "Intercept")

# brms: inspect priors before fitting
get_prior(y ~ x1 + x2, data = data.frame(y = rbinom(100, 1, 0.5),
                                         x1 = rnorm(100), x2 = rnorm(100)),
          family = bernoulli)
```

10.30.6 Output & Results

`get_prior()` lists every parameter class with its default prior, ready to be overridden. After fitting, `prior_summary(fit)` confirms exactly which priors entered the model — a step worth running for any analysis intended for publication.

10.30.7 Interpretation

A reporting sentence: "We placed weakly informative priors on all coefficients ($\beta \sim \mathcal{N}(0, 2.5)$ on standardised predictors, intercept $\sim \mathcal{N}(0, 10)$) and a $\text{half-Cauchy}(0, 2.5)$ prior on the random-effect standard deviation; a sensitivity analysis under tighter $\mathcal{N}(0, 1)$ slopes produced indistinguishable posterior summaries." Always disclose priors and at least one sensitivity check; reviewers increasingly expect both.

10.30.8 Practical Tips

- Defaults from `brms` and `rstanarm` are weakly informative and well-engineered; override them only when you have a reason.
- Flat priors are not "neutral"; they can dominate hierarchical posteriors at the boundary and make sampling unstable.
- For variance components, use half-Cauchy or half-Normal priors with scales matched to the natural variability of the data; never use a flat prior on a SD.
- For unstandardised predictors, multiply prior scales by the empirical SD of the predictor so that the implied effect range is plausible.
- Run prior predictive simulations (`sample_prior = "only"` in `brms`) before fitting; the implied range of `y` is the simplest check that the prior is not absurd.
- Sensitivity analysis is mandatory for small-sample studies; report the posterior under at least two prior scales differing by an order of magnitude.

10.30.9 R Packages Used

`brms` for prior specification with `get_prior()` / `prior_summary()`, `rstanarm` for an alternative interface with auto-scaled defaults, `bayesplot` for prior predictive visualisation, and `bayestestR` for tidy summaries that report priors alongside posteriors.

10.30.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.30.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.31 R-hat and ESS Diagnostics

10.31.1 Introduction

An MCMC posterior summary is only as good as the chain that produced it. Two questions decide whether the draws are usable: have the chains converged on the same distribution, and how many independent observations is the autocorrelated chain actually worth? The first is answered by $\hat{R}$, the second by effective sample size (ESS). These two numbers, reported alongside every parameter in modern Bayesian output, are the minimum diagnostic checklist before any posterior interval is interpreted.

10.31.2 Prerequisites

A working understanding of Markov chain Monte Carlo, the role of warm-up versus sampling iterations, and the autocorrelation structure of MCMC draws.

10.31.3 Theory

$\hat{R}$ compares the variance between chains to the variance within chains. If chains have converged on the same target, the two variance estimates agree and $\hat{R} \to 1$; persistent disagreement inflates $\hat{R}$. The Vehtari et al. (2021) rank-normalised variant handles heavy-tailed posteriors and is the default in `posterior::summarise_draws()`. ESS expresses the variance of the Monte Carlo estimator as if the draws were independent: $n_{\text{eff}} = N/(1 + 2\sum_k \rho_k)$ for autocorrelations ρ_k. Bulk ESS quantifies precision of the posterior mean and median; tail ESS quantifies precision of the 5 % and 95 % quantiles, which require many more draws because they are estimated from a thin part of the distribution.

10.31.4 Assumptions

At least four chains started from dispersed initial values, with adequate warm-up so that the sampler has tuned step size and mass matrix. A single chain cannot diagnose convergence, regardless of how long it is — between-chain variance is undefined.

10.31.5 R Implementation

```r
library(brms); library(posterior)

data(sleepstudy, package = "lme4")
fit ← brm(Reaction ~ Days + (Days | Subject), data = sleepstudy,
          chains = 4, iter = 2000, refresh = 0)

# Summary with R-hat and ESS
summary(fit)

# Per-parameter
post_draws ← as_draws_df(fit)
summarise_draws(post_draws)
```

10.31.6 Output & Results

`summary(fit)` reports `Rhat`, `Bulk_ESS`, and `Tail_ESS` for every parameter. `summarise_draws()` returns the same diagnostics in a tidy tibble, ready for `dplyr` filtering when a model has hundreds of parameters and you want to know which one needs attention.

10.31.7 Interpretation

A reporting sentence: "All parameters had $\hat{R} \leq 1.00$, bulk ESS $> 2{,}000$, and tail ESS $> 1{,}500$; sampling is adequate for posterior intervals." Conventional thresholds: $\hat{R} < 1.01$ indicates convergence, $\hat{R} > 1.05$ is a clear failure; bulk and tail ESS should each exceed 400 for credible interval estimation to be reliable.

10.31.8 Practical Tips

- If $\hat{R}$ exceeds 1.01, double the iteration count first; if the problem persists, reparameterise (e.g. non-centred random effects) rather than throwing more iterations at it.
- Low ESS relative to total draws (ratios under 0.1) signals strong autocorrelation; raise `adapt_delta` or examine the model for funnel-shaped geometry.
- Divergent transitions in HMC are a separate, more serious diagnostic — they indicate that the sampler missed regions of the posterior, not just that it explored slowly. Always inspect them with `bayesplot::mcmc_parcoord()` or `pairs()`.
- Trace plots remain a useful supplement: a "hairy caterpillar" with chains overlapping is what convergence looks like; a chain wandering away or stuck in a valley is visible long before $\hat{R}$ catches it for short runs.
- For very high-dimensional models, monitor $\hat{R}$ and ESS for *derived* quantities of interest (e.g. predictions, contrasts), not just raw parameters; convergence of one does not imply the other.

10.31.9 R Packages Used

`posterior` for the rank-normalised $\hat{R}$, bulk and tail ESS, and tidy-draw structures, `bayesplot` for diagnostic plots, and `brms` (or `rstanarm`, `cmdstanr`) as the modelling backend that produces the draws in the first place.

10.31.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.31.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.32 rstanarm

10.32.1 Introduction

`rstanarm` is a sister package to `brms` from the Stan team. Where `brms` translates each formula into a fresh Stan program and compiles it on the fly, `rstanarm` ships with a fixed catalogue of pre-compiled models — `stan_lm`, `stan_glm`, `stan_glmer`, `stan_polr`, `stan_betareg`, and a handful of others — that match the most common R regression interfaces. The trade-off is straightforward: no compilation latency at fit time, weakly informative defaults curated by experts, and a learning curve identical to base-R modelling functions, in exchange for less flexibility when the model deviates from the standard catalogue.

10.32.2 Prerequisites

A working knowledge of generalised linear models, an idea of what a prior and a posterior are, and comfort with R's `lm()` / `glm()` / `lmer()` syntax. No C++ toolchain is needed beyond what installing the package itself requires, since the Stan models are pre-compiled.

10.32.3 Theory

Each `stan_*` function is a thin R wrapper around a Stan program that has been compiled when the package was built. The program already includes weakly informative default priors centred at zero with scales determined by the standard deviation of each predictor. Override defaults through the `prior`, `prior_intercept`, `prior_aux`, and `prior_covariance` arguments, each accepting one of `normal()`, `student_t()`,

`cauchy()`, `laplace()`, `exponential()`, or `decov()`. Because the priors are scaled to the predictors, the same numerical scale (e.g. `normal(0, 2.5)`) is sensible across very different predictor ranges.

10.32.4 Assumptions

The assumptions of the underlying GLM or GLMM apply: linearity in the linear predictor, independence after grouping, the appropriate link / variance relationship for the family. Convergence ($\hat{R} < 1.01$, no divergent transitions) is a precondition for trusting the posterior summary that `rstanarm` returns.

10.32.5 R Implementation

```
library(rstanarm)

set.seed(2026)
d ← data.frame(y = rnorm(100), x = rnorm(100))

# Same syntax as lm() but Bayesian
fit ← stan_glm(y ~ x, data = d,
               prior = normal(0, 2.5),
               prior_intercept = normal(0, 5),
               prior_aux = exponential(1),
               chains = 4, iter = 2000, refresh = 0)
summary(fit)
posterior_interval(fit, prob = 0.95)
```

10.32.6 Output & Results

`summary(fit)` lists the posterior mean, standard deviation, 10 / 50 / 90 % quantiles, mean posterior predictive checks (PPP), $\hat{R}$, and effective sample size for each parameter. `posterior_interval()` returns the requested credible interval. Because the same Stan program is reused, fits typically take seconds rather than minutes.

10.32.7 Interpretation

A reporting sentence: “A Bayesian linear regression (`stan_glm`, weakly informative priors, four chains × 2,000 iterations) estimated the slope at 0.47 (95 % CrI 0.27 to 0.68); the posterior overlaps the OLS point estimate but quantifies the residual uncertainty rather than just standard errors.” Compared with frequentist output, the credible interval has the direct probability interpretation that students often (incorrectly) assign to confidence intervals.

10.32.8 Practical Tips

- Use `prior_summary(fit)` to verify exactly which priors were applied — the auto-scaling can surprise you when predictors are on unusual scales.
- For mixed models, `stan_glmer()` mirrors `lme4::glmer()` syntax including correlated random slopes.

- `pp_check(fit)` provides graphical posterior predictive checks; `bayes_R2()` returns the Bayesian counterpart of R^2 as a posterior distribution.
- `loo(fit)` and `waic(fit)` attach information criteria with Pareto-k diagnostics, ready for `loo_compare()` between models.
- When the model you need is not in the catalogue (for example, custom likelihoods, censored data, ordinal-with-arbitrary-link), reach for `brms` or raw Stan; `rstanarm` is deliberately scoped.
- For very small datasets, prefer `student_t(df = 3)` priors over `normal()` to reduce posterior sensitivity to a single influential observation.

10.32.9 R Packages Used

`rstanarm` for the pre-compiled Stan models, `bayesplot` for diagnostic plots underlying `pp_check()`, `loo` for cross-validation, and `posterior` for working with the underlying draws.

10.32.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.32.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.33 The Savage-Dickey Density Ratio

10.33.1 Introduction

The Savage-Dickey density ratio is a small but powerful identity: when a point-null hypothesis is nested inside a continuous alternative, the Bayes factor in favour of the null can be read off as the ratio of the marginal posterior to the marginal prior, both evaluated at the null value. This sidesteps the notoriously hard computation of the marginal likelihood under each model and turns Bayes-factor calculation into something you can do from MCMC draws and a one-line evaluation of the prior density. It is the Bayesian counterpart of asking "did the data move probability mass away from the null?" and quantifying the answer.

10.33.2 Prerequisites

A working understanding of Bayes factors, posterior densities, and the difference between point-null and interval-null hypotheses. Familiarity with density estimation from MCMC samples is also useful.

10.33.3 Theory

For a point null $H_0 : \theta = \theta_0$ nested in a continuous alternative $H_1 : \theta \in \Theta$, with the prior under H_1 taken as the marginal prior $p(\theta)$ on the parameter of interest, the Bayes factor in favour of the null is

$$\mathrm{BF}_{01} = \frac{p(\theta_0 \mid y)}{p(\theta_0)}.$$

If the posterior puts more density at the null than the prior did, the data support H_0; if it puts less, they support H_1. The quantity is interpretable on Jeffreys' evidence scale and is invariant under monotone reparameterisations of nuisance parameters because the alternative's prior must be set up consistently. The identity assumes the prior under H_1 reduces to the prior under H_0 when $\theta = \theta_0$; this is automatic for nested-null setups but can fail under improper or hierarchical priors.

10.33.4 Assumptions

A continuous parameter, the null nested inside the alternative, and a proper prior whose density at θ_0 is finite and known. Improper priors invalidate the construction because the Bayes factor depends on a normalisation constant that has been thrown away.

10.33.5 R Implementation

```r
library(polspline)

set.seed(2026)
# Posterior samples (e.g., from MCMC)
theta_post <- rnorm(5000, mean = 0.2, sd = 0.3)
theta_prior_density <- dnorm(0, mean = 0, sd = 1)   # N(0, 1) prior at 0

# Posterior density at 0 via logspline fit
logspl <- logspline(theta_post)
post_density_at_0 <- dlogspline(0, logspl)

BF01 <- post_density_at_0 / theta_prior_density
BF01; 1/BF01
```

10.33.6 Output & Results

The ratio of two scalars, ideally accompanied by its inverse so both directions are visible. A logspline density fit to the posterior draws is a robust choice: it captures skew and tail behaviour better than a normal approximation and is less noisy than a kernel density estimate at the boundary.

10.33.7 Interpretation

A reporting sentence: "The Savage-Dickey Bayes factor was $\mathrm{BF}_{10} = 3.1$ under a $\mathcal{N}(0, 1)$ prior on the alternative — moderate evidence for a non-zero effect — versus $\mathrm{BF}_{01} = 0.32$." On Jeffreys' scale, $1 < \mathrm{BF} < 3$ is anecdotal, 3–10 is moderate, 10–30 strong, and beyond 30 very strong; the symmetric thresholds apply to BF_{01} for evidence in favour of the null.

10.33.8 Practical Tips

- Always disclose the prior under the alternative; the Bayes factor is sensitive to its scale, particularly via the Lindley-Bartlett paradox where a very wide prior favours the null mechanically.
- Use a logspline or kernel density estimate that is robust at the boundary — naive Gaussian approximations break down when the posterior is skewed.
- Confirm enough posterior draws fall near θ_0 for the density estimate to be stable; if the data strongly support H_1, the posterior density at the null is small and noisy. Increase chain length when reporting strong evidence against H_0.
- For interval nulls or composite hypotheses, the Savage-Dickey identity does not apply; use bridge sampling (`bridgesampling::bridge_sampler()`) instead.
- Prefer a sensitivity analysis: report the Bayes factor under at least two prior scales, so readers can judge how prior-driven the conclusion is.

10.33.9 R Packages Used

`polspline` for the logspline posterior density estimate at the null, `bayestestR` for an integrated Savage-Dickey wrapper that works directly with `brms` and `rstanarm` fits, and `bridgesampling` for the more general marginal-likelihood approach when the Savage-Dickey shortcut does not apply.

10.33.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.33.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.34 Stan: Introduction

10.34.1 Introduction

Stan is a probabilistic programming language with an R-like syntax that compiles to highly optimised C++. Writing a model directly in Stan gives you full control over the likelihood, the prior, and any reparameterisation that improves sampling — control that the higher-level wrappers `brms` and `rstanarm` deliberately abstract away. The cost is verbosity, but the payoff is a transparent specification: every line in the `model` block contributes a term to the log-posterior, and a Stan program can be archived as the canonical statement of an analysis.

10.34.2 Prerequisites

A working understanding of Bayesian inference, the difference between a prior and a likelihood, and the basics of Hamiltonian Monte Carlo. You also need a working C++ toolchain (Rtools on Windows, build-essential on Linux) so that `rstan` or `cmdstanr` can compile the model.

10.34.3 Theory

A Stan program is organised into blocks evaluated in a specific order:

- `data`: variables that come from the user, dimensioned and constrained.
- `transformed data` (optional): deterministic transformations of the data computed once at compile time.
- `parameters`: continuous unknowns to be inferred. Constraints (`<lower=0>`) are reparameterised internally, which is why discrete parameters have to be marginalised out.
- `transformed parameters` (optional): deterministic functions of parameters, recomputed at every leapfrog step.
- `model`: contributions to the log-posterior — both priors on parameters and the likelihood given the data.
- `generated quantities` (optional): posterior predictive draws, log-likelihoods for LOO-CV, or any deterministic quantity to monitor.

The default sampler is the No-U-Turn variant of HMC, which adapts step size and mass matrix during a warm-up phase before producing draws.

10.34.4 Assumptions

Continuous parameters or correctly marginalised discrete latent states. The posterior must be log-concave only locally, but pathological geometry (funnels, ridges, multimodality) often shows up as divergent transitions and demands reparameterisation.

10.34.5 R Implementation

```
// Stan model: linear regression
data {
  int<lower=1> N;
  vector[N] x;
  vector[N] y;
}
parameters {
  real alpha;
  real beta;
  real<lower=0> sigma;
}
model {
  alpha ~ normal(0, 10);
  beta  ~ normal(0, 2.5);
  sigma ~ exponential(1);
  y ~ normal(alpha + beta * x, sigma);
}
```

```
library(rstan)

set.seed(2026)
d ← list(N = 100, x = rnorm(100), y = rnorm(100, 2 + 1.5 * rnorm(100), 1))

# Assume the Stan code is saved to "model.stan"
# fit ← stan("model.stan", data = d, chains = 4, iter = 2000)
# print(fit, pars = c("alpha", "beta", "sigma"))
```

10.34.6 Output & Results

`print(fit)` returns posterior mean, standard error of the mean, posterior standard deviation, 2.5 / 25 / 50 / 75 / 97.5 % quantiles, effective sample size, and $\hat{R}$ for every monitored parameter. Diagnostics also include divergent-transition counts, maximum tree depth saturation, and energy diagnostics — any of which, if non-zero, signals geometry trouble.

10.34.7 Interpretation

Compared with `lm()` or `glmer()` style output, Stan output is more verbose but more explicit: there is no implicit prior, no black-box optimisation, and the credible interval has a direct probability interpretation conditional on the model. A reporting sentence: "A Stan-compiled linear regression with weakly informative priors estimated the slope at 1.49 (95 % CrI 1.30 to 1.69), with $\hat{R} = 1.00$ and $n_{\text{eff}} > 3{,}500$ for every parameter."

10.34.8 Practical Tips

- Use `brms::stancode()` or `rstanarm::stan_glm` source as templates — they are well-engineered models you can copy and modify.
- Reparameterise hierarchical models in non-centred form ($\theta_j = \mu + \tau \cdot \tilde{\theta}_j$ with $\tilde{\theta}_j \sim \mathcal{N}(0, 1)$) to eliminate funnel-shaped divergences.

- Prefer `cmdstanr` over `rstan` for new projects: it tracks upstream Stan releases, has a cleaner API, and avoids the binary-compatibility surprises that plague R-package builds.
- Vectorise statements (`y ~ normal(mu, sigma)` over `for` loops); the compiler optimises vectorised forms more aggressively.
- Place posterior predictive draws in `generated quantities`, not the `model` block, so they do not enter the log-density evaluation and slow sampling.

10.34.9 R Packages Used

`rstan` or `cmdstanr` to compile and sample from Stan programs, `posterior` for working with draw arrays, `bayesplot` for diagnostic and posterior plots, and `loo` for leave-one-out cross-validation.

10.34.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.34.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.35 tidybayes Workflow

10.35.1 Introduction

`tidybayes` is the bridge between Bayesian model objects and the rest of the tidyverse. After fitting a model with `brms`, `rstanarm`, or raw Stan, you typically want to compare posteriors across parameters, summarise interval estimates by subgroup, or layer credible bands on top of raw observations. Doing this with the matrices that samplers natively produce is awkward; `tidybayes` reshapes them into long-format tibbles in which every draw of every parameter occupies its own row, making the familiar `dplyr` verbs and `ggplot2` geoms work without modification.

10.35.2 Prerequisites

A working understanding of Bayesian inference (priors, likelihoods, posteriors), comfort with `dplyr` pipelines, and a fitted MCMC model object. You do not need to write Stan by hand — `brm()` or `stan_glm()` is enough.

10.35.3 Theory

A posterior is a joint distribution over parameters represented as S MCMC draws. `tidybayes::spread_draws()` returns one row per draw per parameter, so a model with K regression coefficients yields $S \times K$ rows. `gather_draws()` produces an even longer format with a `.variable` column, useful when faceting. The package adds geoms — `stat_halfeye`, `stat_pointinterval`, `stat_lineribbon` — that compute density and interval summaries on the fly, so the data frame stays untransformed. Predictive draws come from `epred_draws()` (expectation of the posterior predictive) and `predicted_draws()` (full predictive distribution including residual variation).

10.35.4 Assumptions

The model has already converged: chains have mixed, $\hat{R}$ is below 1.01, and effective sample size is adequate. `tidybayes` does not diagnose convergence; it visualises whatever draws you give it, so a poorly mixed chain produces a misleading posterior summary.

10.35.5 R Implementation

```
library(brms); library(tidybayes); library(ggplot2)

data(mtcars)
fit ← brm(mpg ~ wt + hp, data = mtcars, chains = 4, iter = 2000, refresh = 0)

# Tidy draws
draws ← fit ▷
  spread_draws(b_wt, b_hp)

head(draws)

# Visualise posterior
draws ▷
  pivot_longer(c(b_wt, b_hp), names_to = "parameter", values_to = "value") ▷
  ggplot(aes(x = value, y = parameter)) +
  stat_halfeye(fill = "#2A9D8F") +
  theme_minimal()

# Conditional effects
fit ▷
  epred_draws(newdata = data.frame(wt = 2.5, hp = seq(50, 300, 10))) ▷
  ggplot(aes(x = hp, y = .epred)) +
  stat_lineribbon(alpha = 0.3) +
  theme_minimal()
```

10.35.6 Output & Results

`spread_draws()` returns a tibble with `.chain`, `.iteration`, `.draw`, and one column per parameter. The half-eye plot shows the posterior density together with a black point at the median and bars at the 66 % and 95 % credible intervals. `stat_lineribbon` produces a shaded band that widens at horsepower values poorly supported by the data, communicating where extrapolation is risky.

10.35.7 Interpretation

A reporting sentence: "The posterior median effect of weight on miles-per-gallon was -3.9 (95 % CrI -5.4 to -2.4), while horsepower contributed -0.03 (95 % CrI -0.05 to -0.01); both intervals exclude zero, consistent with each predictor having a non-trivial association after the other is controlled for." Pair narrative numbers with the half-eye plot so readers can see the full posterior shape and not just an interval.

10.35.8 Practical Tips

- Use `spread_draws()` when you want one column per parameter and `gather_draws()` when you plan to facet by parameter name.
- Regular expressions are accepted: `spread_draws(fit, b_[a-z]+)` extracts every regression coefficient at once.
- `add_epred_draws()` and `add_predicted_draws()` attach predictive draws directly to a `newdata` frame, which is convenient when overlaying credible bands on observed points.
- Use `mean_qi()`, `median_qi()`, or `mode_hdi()` to collapse draws into interval summaries; `point_interval = mean_qi` is the default in most geoms but is worth setting explicitly for reproducibility.
- The package is engine-agnostic: the same code path supports `rstan`, `cmdstanr`, `rstanarm`, and `brms`, so refactoring between them rarely affects downstream plotting code.
- Avoid coercing draws to wide format prematurely; `ggplot2` is happiest with the long tibble that `tidybayes` returns.

10.35.9 R Packages Used

`brms` for model fitting, `tidybayes` for posterior reshaping and stat geoms, `ggplot2` and `dplyr` for plotting and manipulation, plus `posterior` (loaded transitively) for low-level draw arrays.

10.35.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.35.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.36 WAIC

10.36.1 Introduction

The widely applicable information criterion (WAIC), introduced by Watanabe, estimates the expected out-of-sample prediction score from posterior samples without holding out data. It is the Bayesian successor to AIC and DIC: AIC requires a single point estimate (and assumes regularity that fails for hierarchical models); DIC plug-ins a posterior mean of the deviance and is unstable in mixed models. WAIC instead averages the pointwise log-posterior predictive density across posterior draws and penalises by an effective-parameter term, making it directly applicable to any model with a well-defined pointwise likelihood.

10.36.2 Prerequisites

A working understanding of information criteria, posterior predictive distributions, and the relationship between in-sample fit and out-of-sample prediction.

10.36.3 Theory

For data $y_1, \dots, y_n$ and posterior draws $\theta^{(s)}$, the log pointwise predictive density is

$$\widehat{\mathrm{lpd}} = \sum_{i=1}^{n} \log \left(\frac{1}{S} \sum_{s=1}^{S} p(y_i \mid \theta^{(s)}) \right).$$

The effective parameter count p_{waic} is the sum of pointwise posterior variances of the log-likelihood. Then

$$\mathrm{WAIC} = -2\big(\widehat{\mathrm{lpd}} - p_{\mathrm{waic}}\big).$$

Lower is better. Watanabe's asymptotic theory shows that for regular models WAIC is asymptotically equivalent to leave-one-out cross-validation. The standard error of a WAIC difference between two models comes from the sample standard deviation of the pointwise differences scaled by $\sqrt{n}$, providing a principled threshold for "meaningful" differences.

10.36.4 Assumptions

A pointwise likelihood is well-defined (so each observation contributes one term to the sum) and the posterior is well-mixed. WAIC inherits the exchangeability assumption of standard cross-validation — for clustered or temporal data, replace it with leave-one-cluster-out or leave-future-out variants.

10.36.5 R Implementation

```r
library(brms); library(loo)

data(mtcars)
fit1 ← brm(mpg ~ wt, data = mtcars, chains = 2, iter = 2000, refresh = 0)
fit2 ← brm(mpg ~ wt + hp, data = mtcars, chains = 2, iter = 2000, refresh = 0)

w1 ← waic(fit1)
w2 ← waic(fit2)

loo_compare(w1, w2)
```

10.36.6 Output & Results

`waic()` returns the WAIC value, p_{waic}, and a vector of pointwise contributions. `loo_compare()` aligns two WAIC objects, reports their differences and standard errors, and orders models from best (top) to worst (bottom).

10.36.7 Interpretation

A reporting sentence: "WAIC favoured the two-predictor model with $\Delta\text{WAIC} = -12 \pm 4$ (standard error of the difference); under the rule-of-thumb that differences exceeding $2 \times \text{SE}$ are meaningful, the data clearly support including horsepower in addition to weight." Always quote both the difference and its standard error — a "12-point improvement" without context is uninformative.

10.36.8 Practical Tips

- Use PSIS-LOO from the `loo` package whenever possible; it is more numerically stable than WAIC and provides Pareto-k diagnostics that flag influential observations one at a time.
- WAIC can be unstable when a single observation contributes a large fraction of the variance of the log-likelihood; if you see p_{waic} comparable to the number of observations, do not trust the criterion and switch to LOO.
- Differences in WAIC are noisy; only invest belief in differences that exceed 2 standard errors.
- WAIC is well-defined for non-nested models, unlike the likelihood-ratio test, which makes it natural for comparing models with different families or link functions.
- For hierarchical models, choose the leave-out unit deliberately: leaving out a single observation may understate predictive uncertainty if all other observations from the same cluster remain in the training fold.

10.36.9 R Packages Used

`loo` for the canonical `waic()` and `loo()` implementations and `loo_compare()`, `brms` and `rstanarm` for fits that expose pointwise log-likelihoods automatically, and `posterior` for the underlying draw arrays.

10.36.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.36.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.37 Weakly Informative Priors

10.37.1 Introduction

Weakly informative priors are the modern default in applied Bayesian work. They assign non-trivial probability to a plausible range of parameter values while remaining broad enough that the data dominate the posterior. They differ from "non-informative" priors (which can be improper or pathological) and from informative priors (which encode substantive knowledge): the goal is to regularise the inference, rule out absurd values, and stabilise sampling without injecting domain claims that the data cannot challenge. The result is a posterior that looks much like a frequentist estimate when data are abundant, but that behaves sensibly under separation, sparsity, and high-dimensional regression where likelihood-only methods misbehave.

10.37.2 Prerequisites

A working understanding of priors, the role of the prior in Bayesian updating, and basic familiarity with regression on standardised predictors.

10.37.3 Theory

A prior is *weakly informative* if it gives small but non-zero probability to extreme values, has a finite second moment when possible, and is centred at a value that does not bias the analysis (zero, in most regression contexts). Standard choices for standardised predictors:

- Regression slopes: $\mathcal{N}(0, 2.5)$ or $\mathcal{N}(0, 5)$ on standardised covariates.
- Intercept (after centring y): $\mathcal{N}(0, 10)$.
- Standard deviations of random effects or residuals: $\text{half-Normal}(0, 2.5)$, $\text{half-Cauchy}(0, 5)$, or $\text{Exponential}(\lambda)$ with rate matched to the SD scale.
- Correlation matrices: LKJ prior with shape $\eta \geq 2$, which gently regularises toward weakly correlated structures.

A useful calibration test: simulate from the prior predictive distribution and check that the implied range of `y` is plausible. If the prior implies effects of $\pm 1{,}000$ on a quantity bounded by physiology, it is too wide.

10.37.4 Assumptions

Predictors are on a sensible scale (typically standardised), priors are proper (so the posterior is integrable), and the prior predictive range covers the plausible space without dominating it.

10.37.5 R Implementation

```r
library(brms)

# Default brms priors are weakly informative
data_ex <- data.frame(y = rnorm(100), x = rnorm(100))
get_prior(y ~ x, data = data_ex)

# Explicit weakly informative
fit <- brm(y ~ x, data = data_ex,
           prior = prior(normal(0, 2.5), class = "b") +
                   prior(normal(0, 5),   class = "Intercept") +
                   prior(exponential(1), class = "sigma"),
           iter = 2000, chains = 4, refresh = 0)
```

10.37.6 Output & Results

`get_prior()` lists every parameter class together with the package's default prior; explicit overrides replace them by class. After fitting, `prior_summary(fit)` confirms exactly which priors entered the model — a step worth running for any analysis intended for publication.

10.37.7 Interpretation

A reporting sentence: "Weakly informative priors $\mathcal{N}(0, 2.5)$ on standardised slopes and $\text{Exponential}(1)$ on the residual scale yielded a posterior centred close to the OLS estimate ($\beta = 0.51$, 95 % credible interval 0.31 to 0.71), with negligible prior pull at this sample size; a sensitivity analysis under $\mathcal{N}(0, 1)$ produced essentially identical posterior summaries." Disclosing both the prior and a sensitivity check is the gold-standard report.

10.37.8 Practical Tips

- Gelman's heuristic: half-Cauchy with scale 2.5–5 for variance components — heavy enough to allow large variances when warranted, regularising enough to prevent boundary-zero estimates.
- For unstandardised predictors, multiply the prior scale by the predictor's empirical SD to keep the implied effect range plausible.
- LKJ(2) for correlation matrices is a sensible default; tighten only when you have reason to expect weak correlations.
- Always run prior predictive checks (`sample_prior = "only"`); they reveal absurd implied ranges before the data can mask them.

- For non-standardised log-odds problems where the baseline log-odds is far from zero, widen the prior on the intercept (e.g. $\mathcal{N}(0, 10)$) so the prior mode does not exclude the data-implied baseline.

10.37.9 R Packages Used

`brms` for formula-based prior specification with `get_prior()` / `prior_summary()`, `rstanarm` for an alternative interface with auto-scaled defaults, `bayestestR` for posterior summaries that report the prior alongside the posterior, and `bayesplot` for prior predictive visualisation.

10.37.10 For Reviewers

Use the shared reviewer checklist, together with this method's assumptions and practical tips.

Method-specific reviewer notes have not yet been developed for this entry. The links above support a structured review; they do not certify the example or a manuscript's application of the method.

10.37.11 See also — labs in this chapter

- Bayesian thinking; α control vs decision error
- brms / Stan, LOO, hierarchical models

10.38 Week 3, Session 1 — Bayesian thinking; α control vs decision error

Testing labs use the main template: Hypothesis → Visualise → Assumptions → Conduct → Conclude.

10.38.1 Learning objectives

- Compute a beta-binomial posterior and interpret its credible intervals in terms of a decision.
- Contrast frequentist α-control (type I error rate across repeated experiments) with Bayesian decision error for a specific experiment.
- Show how prior sensitivity translates into decision sensitivity.

10.38.2 Prerequisites

Probability basics and conjugacy.

10.38.3 Background

Frequentist testing controls the long-run rate of falsely rejecting a true null across repeated experiments; the α = 0.05 convention is a statement about procedure, not about the current dataset. Bayesian inference instead returns a posterior distribution of the unknown and turns that into a decision by weighing costs. For a binomial proportion with a beta prior, the posterior is beta again, so the arithmetic is clean enough to follow carefully.

The two frameworks answer different questions. “Given these data and this model, what is the probability that the treatment is beneficial?” is a Bayesian question. “Across many repetitions of this design, how often would I reject a true null?” is a frequentist question. In applied biomedicine, both questions are relevant; knowing which one you are answering prevents confused write-ups.

Prior sensitivity is a useful discipline regardless of philosophy. A posterior that shifts materially when the prior is varied across reasonable alternatives is less trustworthy than one that does not, and sharing that sensitivity is the Bayesian analogue of transparent model-building.

10.38.4 Setup

```
library(tidyverse)
set.seed(42)
theme_set(theme_minimal(base_size = 12))
```

10.38.5 1. Hypothesis

A small trial observes $k = 7$ successes in $n = 20$ patients of a new response-rate biomarker. Is the underlying response rate meaningfully above 0.2?

10.38.6 2. Visualise

Prior vs posterior for three priors.

```r
k <- 7; n <- 20
theta <- seq(0, 1, length.out = 400)
priors <- tibble(
  name = c("uniform (Beta(1,1))", "sceptical (Beta(2,8))", "flat informative
(Beta(4,4))"),
  a0   = c(1, 2, 4),
  b0   = c(1, 8, 4)
)
post <- priors |>
  rowwise() |>
  mutate(data = list(tibble(
    theta = theta,
    prior = dbeta(theta, a0, b0),
    posterior = dbeta(theta, a0 + k, b0 + n - k)
  ))) |>
  unnest(data) |>
  pivot_longer(c(prior, posterior), names_to = "density")

ggplot(post, aes(theta, value, colour = density)) +
  geom_line() + facet_wrap(~ name) +
  labs(x = expression(theta), y = "density")
```

10.38.7 3. Assumptions

Bernoulli trials, exchangeable, conjugate beta prior. With no prior data on the response rate in this population, we carry all three priors through and compare decisions.

10.38.8 4. Conduct

Posterior summaries.

```r
summ <- priors |>
  rowwise() |>
  mutate(
    post_mean = (a0 + k) / (a0 + b0 + n),
    ci_lo = qbeta(0.025, a0 + k, b0 + n - k),
    ci_hi = qbeta(0.975, a0 + k, b0 + n - k),
    p_gt_0_2 = 1 - pbeta(0.2, a0 + k, b0 + n - k)
  ) |>
  dplyr::select(name, post_mean, ci_lo, ci_hi, p_gt_0_2)
summ
```

A frequentist analogue for comparison.

```r
bt <- binom.test(k, n, p = 0.2, alternative = "greater")
c(estimate = bt$estimate, p_value = bt$p.value,
  ci_lo = bt$conf.int[1], ci_hi = bt$conf.int[2])
```

10.38.9 5. Conclude

> With 7 responders out of 20, the posterior probability that the response rate exceeds 0.2 is `round(summ$p_gt_0_2[1], 2)` under a flat prior and `round(summ$p_gt_0_2[2], 2)` under a sceptical prior. A one-sided exact binomial test against $p_0 = 0.2$ gives $p =$ `signif(bt$p.value, 2)`.

All three priors point the same direction, but the strength of the conclusion depends on the prior. For a go/no-go decision, reporting the posterior probability of exceeding a decision threshold — along with its prior-sensitivity — is more decision-relevant than a *p*-value.

10.38.10 Common pitfalls

- Treating a 95% credible interval as a 95% confidence interval; they are calibrated for different inferences.
- Choosing a prior that pushes the posterior toward the desired decision and not reporting the sensitivity.
- Comparing a one-sided Bayesian tail probability to a two-sided *p*-value.

10.38.11 Further reading

- Spiegelhalter DJ, Abrams KR, Myles JP, *Bayesian Approaches to Clinical Trials and Health-Care Evaluation*.
- McElreath R, *Statistical Rethinking*, ch. 2–3.

10.38.12 Session info

```
sessionInfo()
```

10.38.13 See also — chapter index

- Methods in this chapter
- Find your method (Chapter 0)

10.39 Week 3, Session 2 — brms / Stan, LOO, hierarchical models

Workflow labs use the variant template: Goal → Approach → Execution → Check → Report.

10.39.1 Learning objectives

- Write a simple multilevel model in `brms` syntax and describe each prior.
- Compare two candidate models using leave-one-out (LOO) cross- validation.
- Perform a prior predictive check and interpret it.

10.39.2 Prerequisites

Beta-binomial Bayesian intuition; linear mixed-effects models.

10.39.3 Background

`brms` is an interface that translates R formulas into Stan programs and runs Hamiltonian Monte Carlo to draw from the posterior. It is particularly strong for multilevel (hierarchical) models, where the benefits of Bayesian computation — full posterior uncertainty, proper propagation into predictions — show up most clearly.

LOO cross-validation, implemented via the `loo` package, estimates out-of-sample predictive accuracy from the posterior without refitting, using Pareto-smoothed importance sampling. It is a modern replacement for AIC/BIC in Bayesian model comparison, and comes with diagnostics that warn when the approximation is unreliable.

Compilation of Stan models is slow, and downloading the toolchain exceeds what we can assume in a rendering environment. We therefore show the `brms` code with `#| eval: false` and walk through prior predictive simulation and a small frequentist counterpart that runs end-to-end.

10.39.4 Setup

```
library(tidyverse)
library(lme4)
set.seed(42)
theme_set(theme_minimal(base_size = 12))
```

10.39.5 1. Goal

Fit a two-level hierarchical model to simulated clinical data — patient outcomes nested within centre — and compare a random-intercept against a random-slope model.

10.39.6 2. Approach

Simulate 20 centres with varying mean outcomes and varying dose effects.

```
n_centre ← 20; per_centre ← 15
centre ← rep(seq_len(n_centre), each = per_centre)
alpha0 ← rnorm(n_centre, 0, 1)
beta0  ← rnorm(n_centre, 0.5, 0.3)
dose ← rnorm(n_centre * per_centre)
y ← alpha0[centre] + beta0[centre] * dose + rnorm(n_centre * per_centre, 0, 0.7)
d ← tibble(y = y, dose = dose, centre = factor(centre))
ggplot(d, aes(dose, y, colour = centre)) +
  geom_point(alpha = 0.5) + geom_smooth(method = "lm", se = FALSE) +
  theme(legend.position = "none")
```

10.39.7 3. Execution

A classical `lmer` random-intercept-and-slope fit as a stand-in we can run.

```
fit_lmer_ri  ← lmer(y ~ dose + (1 | centre), data = d)
fit_lmer_rs  ← lmer(y ~ dose + (dose | centre), data = d)
summary(fit_lmer_rs)$varcor
anova(fit_lmer_ri, fit_lmer_rs)
```

Prior predictive simulation (frequentist-style) for the Bayesian counterpart we are about to sketch.

```
prior_sim ← replicate(20, {
  a ← rnorm(1, 0, 1); b ← rnorm(1, 0, 1); sd_a ← abs(rnorm(1, 0, 1))
  a + b * dose + rnorm(length(dose), 0, sd_a)
})
ggplot(tibble(y = as.numeric(prior_sim)), aes(y)) +
  geom_histogram(bins = 40, fill = "grey60", colour = "white") +
  labs(x = "y drawn under priors only", y = "count")
```

The `brms` counterpart.

```
library(brms)
priors ← c(
  prior(normal(0, 2), class = "Intercept"),
  prior(normal(0, 1), class = "b"),
  prior(exponential(1), class = "sd"),
  prior(exponential(1), class = "sigma")
)
m_ri ← brm(y ~ dose + (1 | centre), data = d,
           prior = priors, chains = 2, iter = 2000, refresh = 0)
m_rs ← brm(y ~ dose + (dose | centre), data = d,
           prior = priors, chains = 2, iter = 2000, refresh = 0)
loo_compare(loo(m_ri), loo(m_rs))
```

10.39.8 4. Check

Compare to `lmer` and report pooled vs unpooled estimates.

```
fixef_lmer ← fixef(fit_lmer_rs)
fixef_lmer
```

10.39.9 5. Report

> On simulated hierarchical data (20 centres × 15 patients), a random-slope `lmer` model recovered an average dose effect of `round(fixef_lmer["dose"], 2)` (SE from `lmer`), with substantial centre-to-centre variation. The `brms` sketch shown in the accompanying code would fit the same structure with explicit priors, and `loo_compare` would rank the random-slope model against the random-intercept model.

In practice, the Bayesian and frequentist fits should agree closely when priors are weakly informative and n is large; the Bayesian route adds explicit prior sensitivity and full posterior uncertainty for downstream prediction.

10.39.10 Common pitfalls

- Fitting a flat prior in a small-sample hierarchical model and getting a divergent posterior on variance components.
- Comparing LOO elpd differences smaller than about 4 SE without noting the uncertainty.

- Forgetting to scale continuous predictors; default priors assume normalised inputs.

10.39.11 Further reading

- Bürkner P-C (2017), *brms: An R package for Bayesian multilevel models using Stan*.
- Vehtari A, Gelman A, Gabry J (2017), *Practical Bayesian model evaluation using leave-one-out cross-validation*.

10.39.12 Session info

```
sessionInfo()
```

10.39.13 See also — chapter index

- Methods in this chapter
- Find your method (Chapter 0)