Convergence Diagnostics

Bayesian Mixed Effects Models with brms for Linguists

Job Schepens

2026-01-06

Table of contents

1 The Problem: Can We Trust These Estimates?

1.1 Research Scenario

You've fitted a Bayesian model with `brms`. The code ran without errors. You got parameter estimates and credible intervals. But before you interpret or report these results, you need to ask:

"Did the MCMC sampler actually converge to the posterior distribution?"

If the chains haven't converged, your estimates are **unreliable** — no matter how plausible they look!

1.2 Why Convergence Matters

MCMC (Markov Chain Monte Carlo) is a sampling algorithm that explores the posterior distribution. Think of it like:

- **Goal**: Map out a mountain range (the posterior)
- **Method**: Send out 4 hikers (chains) who wander randomly but tend toward high peaks
- **Success**: After enough time, all hikers explore the same regions proportionally to altitude
- **Failure**: Hikers get stuck in different valleys, never finding the true peaks

Convergence means all chains have "forgotten" their starting points and are sampling from the same distribution.

1.3 What Can Go Wrong?

1.3.1 Example Problems

1. **Chains haven't mixed**: Each chain explores a different part of parameter space
2. **Chains stuck in local modes**: Model has multiple peaks, chains haven't found the global one
3. **High autocorrelation**: Chains move slowly, need many more iterations
4. **Insufficient information**: Data too sparse, posterior poorly defined
5. **Model misspecification**: Priors or likelihood don't match the data structure

1.4 Preview: Our Diagnostic Tools

Today we'll learn to check convergence using:

1. **Trace plots**: Visual inspection of chain behavior over iterations
2. **R-hat statistic**: Quantitative measure of between-chain vs within-chain variance
3. **Effective sample size (ESS)**: How many independent samples you really have
4. **Posterior predictive distributions**: Do parameter values make substantive sense?
5. **Autocorrelation plots**: How correlated are consecutive samples?
6. **Iteration doubling test**: Does inference change with more samples?

2 Where We Are in the Analysis Workflow

2.1 The Bayesian Workflow So Far

```
Module 01-02: Build model + set priors
            ↓
Module 03: Check model fit (posterior predictive)
            ↓
Module 04: Test prior sensitivity
            ↓
Module 05: Compare models (LOO-CV for prediction)
            ↓
Module 06: Practical significance (ROPE, emmeans, marginaleffects)
            ↓
Module 07: Hypothesis comparison (Bayes Factors)
            ↓
Module 08 (TODAY): Convergence diagnostics
            ↓
         Report results!
```

2.2 When to Check Convergence?

Answer: ALWAYS, before interpreting any results!

Typical workflow:

1. Fit model with default settings (4 chains, 2000 iterations)
2. **Check convergence diagnostics** ← TODAY
3. If problems detected:
   - Increase iterations
   - Adjust adapt_delta (for divergent transitions)
   - Reparameterize model
   - Check for coding errors or model misspecification
4. Once converged → proceed to inference

3 The Five Essential Checks

3.1 Overview: What We'll Check

For each model, we'll ask:

1. **Trace plots**: Do chains mix well and explore the same space?
2. **R-hat < 1.01**: Are between-chain and within-chain variances similar?
3. **ESS > 400** (bulk and tail): Do we have enough effective samples?
4. **Low autocorrelation**: Are consecutive samples reasonably independent?
5. **Substantive sense**: Do parameter values match what we know about the phenomenon?

We'll also test: 6. **Stability test**: Does doubling iterations change our conclusions?

Let's work through these with real examples.

4 Setup and Data

4.1 Load Packages

4.2 Simulate Reaction Time Data

We'll use the same data structure as Modules 06 and 07: a psycholinguistic experiment with reaction times.

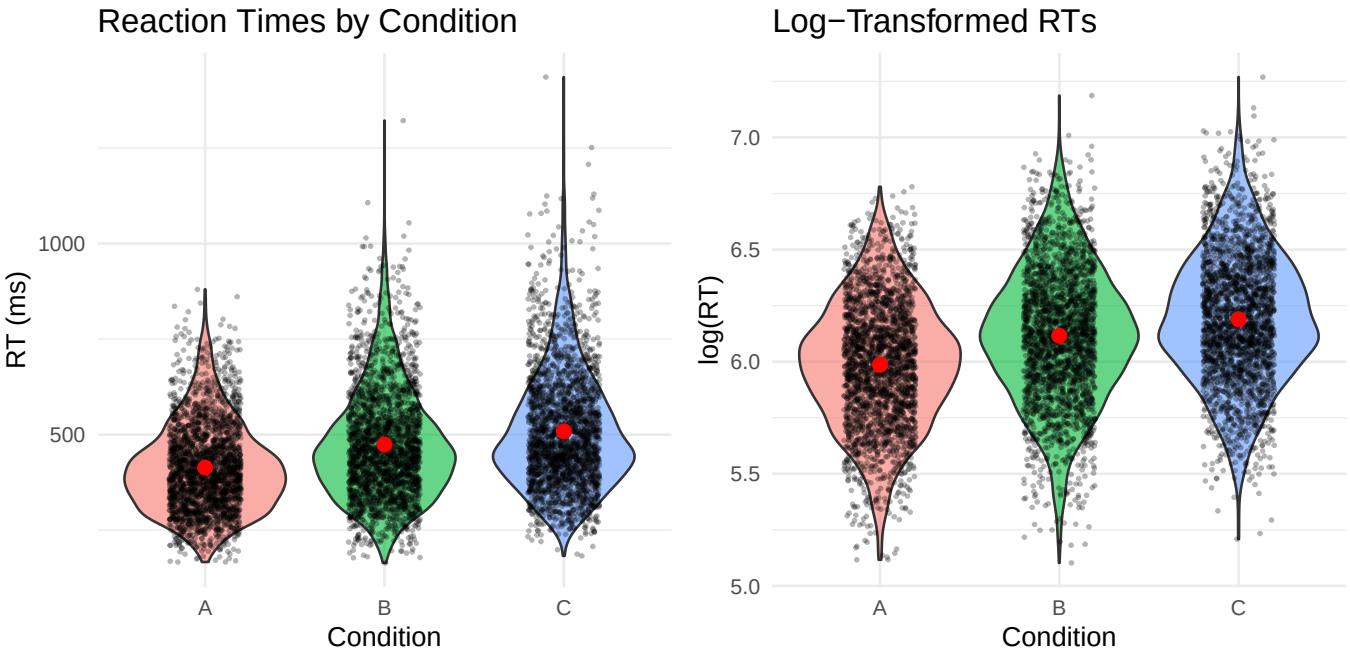
```
=== DATA SUMMARY ===
```

```
# A tibble: 3 x 6
  condition     n mean_rt median_rt sd_rt mean_log_rt
  <fct>     <int>   <dbl>     <dbl> <dbl>       <dbl>
1 A          2400     414       400   117        5.99
2 B          2400     474       454   145        6.12
3 C          2400     509       480   154        6.19
```

```
Condition B - A effect (log): 0.13
```

```
Condition C - A effect (log): 0.203
```

4.3 Visualize the Data



5 Check 1: Trace Plots

5.1 Fit a Well-Behaved Model

Let's start with a model that should converge well.

5.2 Trace Plots: Visual Inspection

A **trace plot** shows parameter values across MCMC iterations for each chain.

Good trace plot (well-mixed chains): - All chains overlap completely - Look like "hairy caterpillars" - No trends or drift over time - Chains quickly forget their starting points

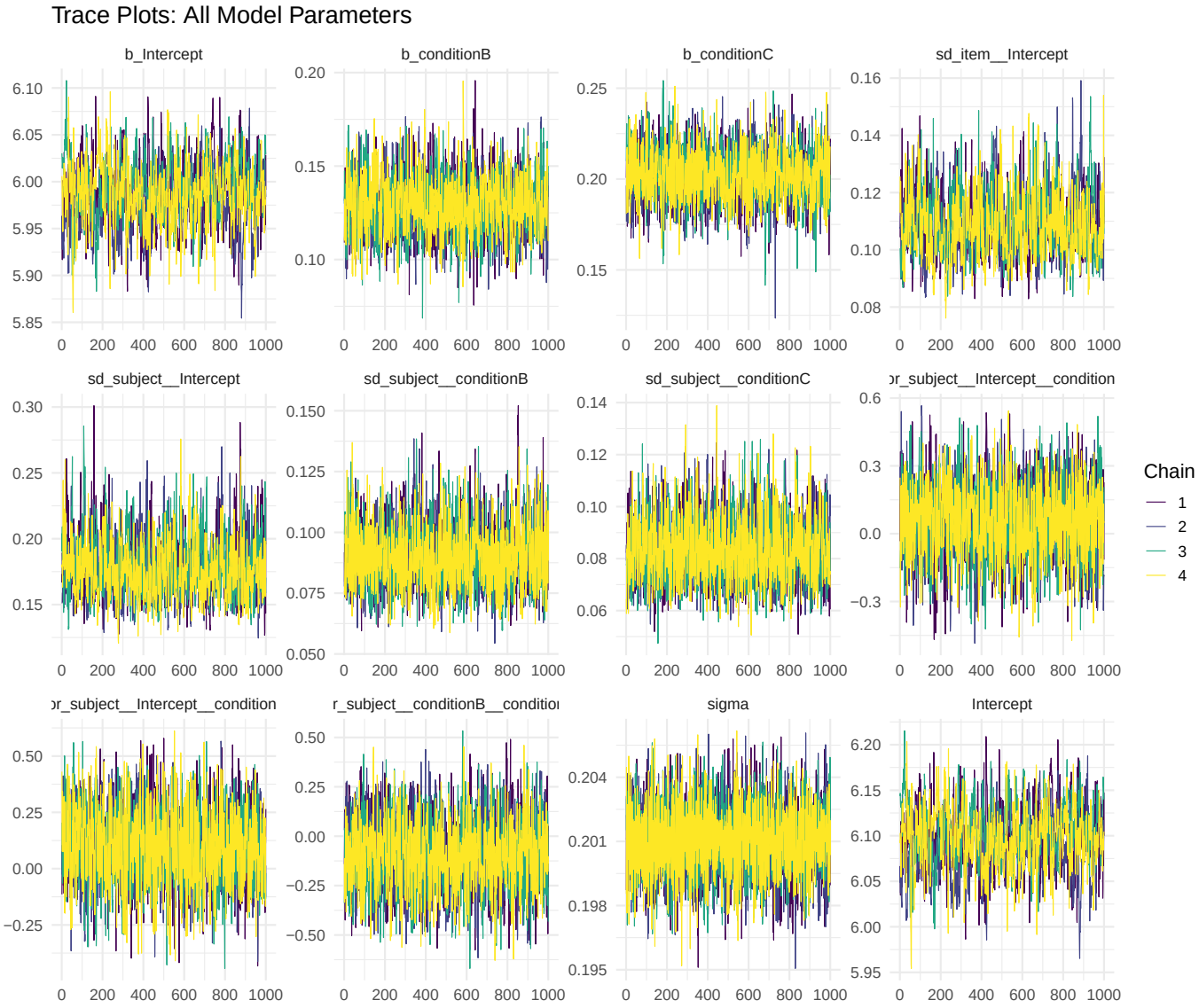
Bad trace plot (convergence problems): - Chains explore different regions - Systematic trends over time - One chain stuck while others move - Slow wandering (high autocorrelation)

What to Look For in Trace Plots

Good signs: - All 4 chains completely overlap - No systematic drift up or down - "Fuzzy caterpillar" appearance - Chains explore same range of values

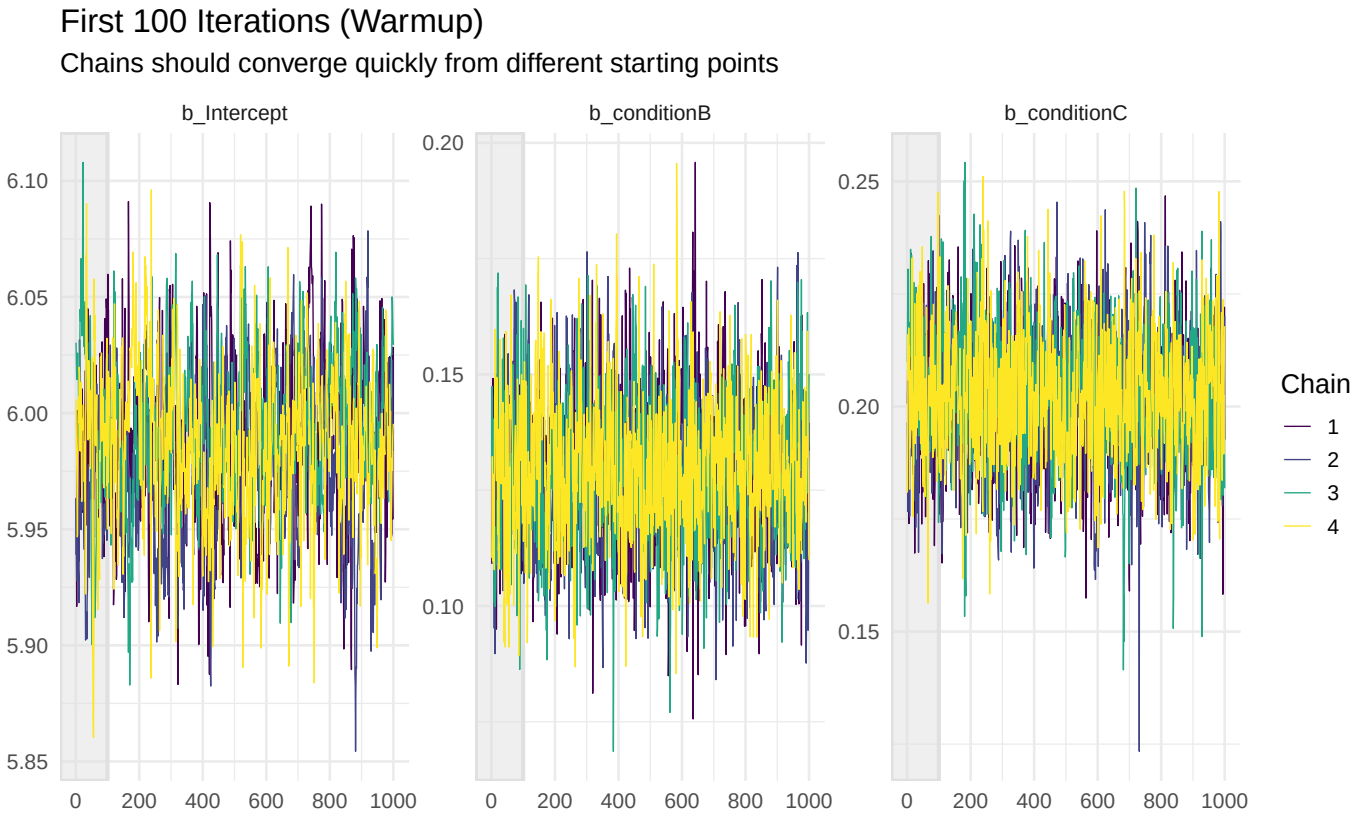
Warning signs: - Chains separated vertically (exploring different modes) - Trends over time (non-stationarity) - One chain behaves differently - Slow, snake-like wandering

5.3 Trace Plot for All Parameters



5.4 Check Starting Points

Did chains "forget" their starting points quickly?



Warmup Period

The first `warmup` iterations (default 1000) are discarded. During warmup: - Stan tunes the sampler's step size and mass matrix - Chains move from starting points toward the typical set - Once adapted, chains should sample from the stationary distribution
We only use post-warmup samples for inference.

6 Check 2: R-hat Statistic

6.1 What is R-hat?

R-hat (Gelman-Rubin statistic) compares: - **Between-chain variance**: How different are the chain means? - **Within-chain variance**: How much do values vary within each chain?

Formula (simplified): $\hat{R}$ = sqrt(between-chain variance / within-chain variance)

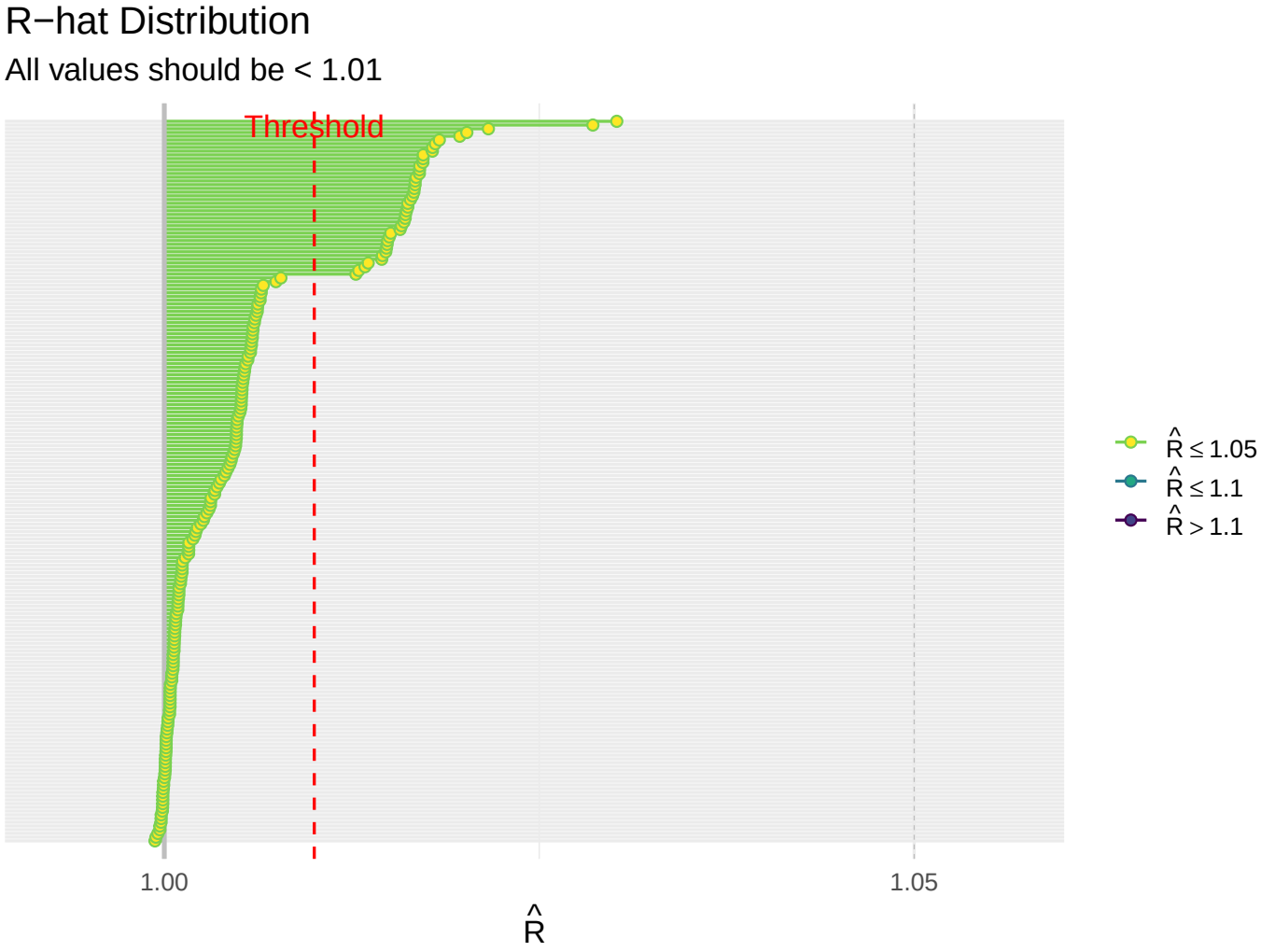
Interpretation: - $\hat{R} = 1.00$: Perfect convergence, chains identical - $\hat{R} < 1.01$: Acceptable (modern threshold, stricter than old 1.1) - $\hat{R} > 1.01$: Chains haven't converged, don't trust estimates!

6.2 Check R-hat for Our Model

```
=== R-HAT SUMMARY ===

Total parameters: 194

Max R-hat: 1.0302
```

```
Parameters with R-hat > 1.01: 42

Top 10 R-hat values:
            b_Intercept                 Intercept  r_subject[2,Intercept]
                 1.0302                    1.0286                  1.0216
r_subject[40,Intercept] r_subject[11,Intercept]  r_subject[4,Intercept]
                 1.0202                    1.0197                  1.0183
r_subject[26,Intercept] r_subject[24,Intercept] r_subject[12,Intercept]
                 1.0181                    1.0179                  1.0179
 r_subject[3,Intercept]
                 1.0173
```

6.3 Visual R-hat Diagnostic



! R-hat Decision Rules

- **$\hat{R} < 1.01$ for all parameters**: Chains converged, proceed with inference
- **$\hat{R} > 1.01$ for any parameter**: Convergence failure!

- Increase iterations (`iter = 4000` or more)
- Check for model misspecification
- Inspect trace plots for the problematic parameters
- Consider reparameterizing the model

7 Check 3: Effective Sample Size (ESS)

7.1 What is ESS?

MCMC samples are **autocorrelated** — consecutive samples are similar. This means: - You have 4000 total samples (4 chains × 1000 post-warmup) - But they don't contain as much information as 4000 independent samples - **Effective sample size (ESS)** estimates the equivalent number of independent samples

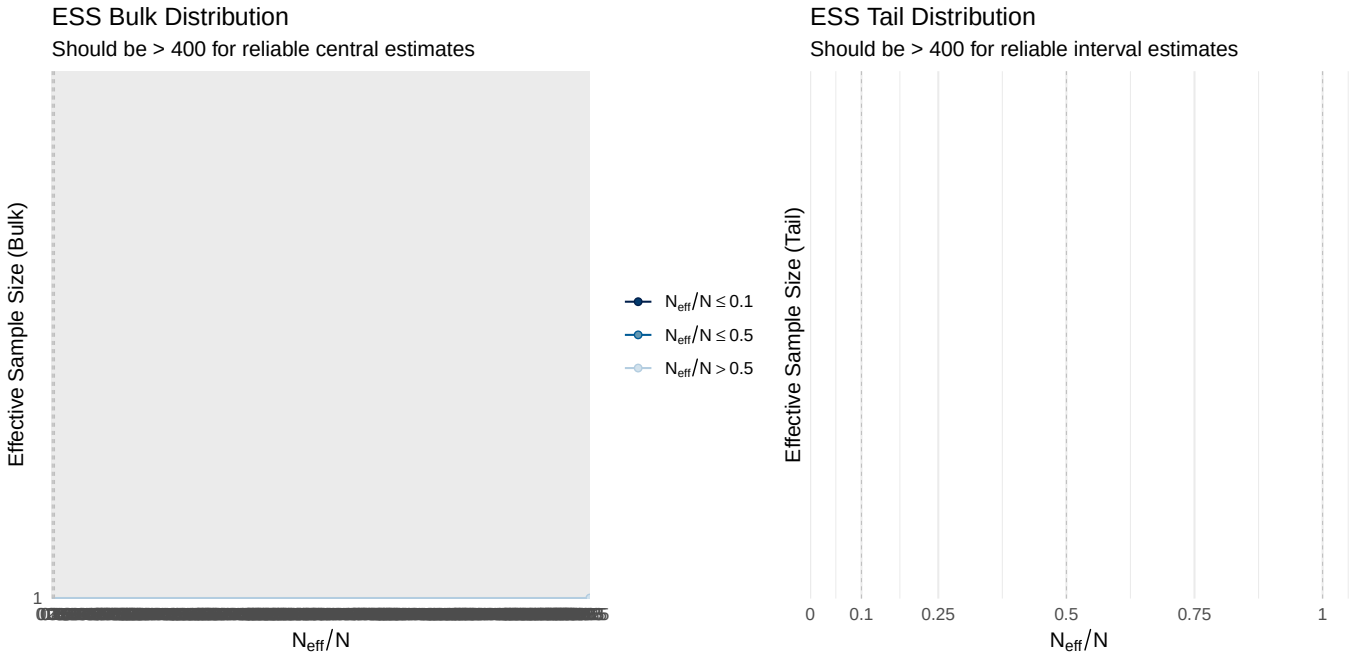
Two types: 1. **ESS Bulk**: For the central 80% of the distribution (for means, medians) 2. **ESS Tail**: For the extreme 5% tails (for quantiles, credible intervals)

Rule of thumb: - **ESS $>$ 400** (both bulk and tail): Reliable estimates - **ESS $<$ 400**: Increase iterations or check for convergence problems

7.2 Check ESS for Our Model

```
=== ESS BULK SUMMARY ===

Min ESS Bulk: 213

Median ESS Bulk: 213

Parameters with ESS Bulk < 400: 1

=== ESS TAIL SUMMARY ===

Min ESS Tail: Inf

Median ESS Tail: NA

Parameters with ESS Tail < 400: 0

Bottom 10 ESS Bulk values:

 [1] 213  NA  NA  NA  NA  NA  NA  NA  NA  NA
```

7.3 Visual ESS Diagnostic



7.4 ESS as Proportion of Total Samples

```
ESS Ratio (ESS / 4000 samples):

  Median Bulk: 0.05

  Median Tail: NA

Interpretation:

  Ratio = 1.0: No autocorrelation (every sample independent)

  Ratio = 0.5: Half as much info as independent samples

  Ratio = 0.1: Highly autocorrelated, need 10× more iterations
```

ESS Decision Rules

For each parameter: - **ESS Bulk $>$ 400 AND ESS Tail $>$ 400**: Sufficient samples - **ESS $<$ 400**: Increase iterations - If ESS 200: Double iterations (`iter = 4000`) - If ESS $<$ 100: Quadruple iterations or investigate model issues
Quick fix: Increase iterations proportionally to achieve ESS $>$ 400:

```
new_iter = current_iter * (400 / min_ess)
```

8 Check 4: Autocorrelation

8.1 What is Autocorrelation?

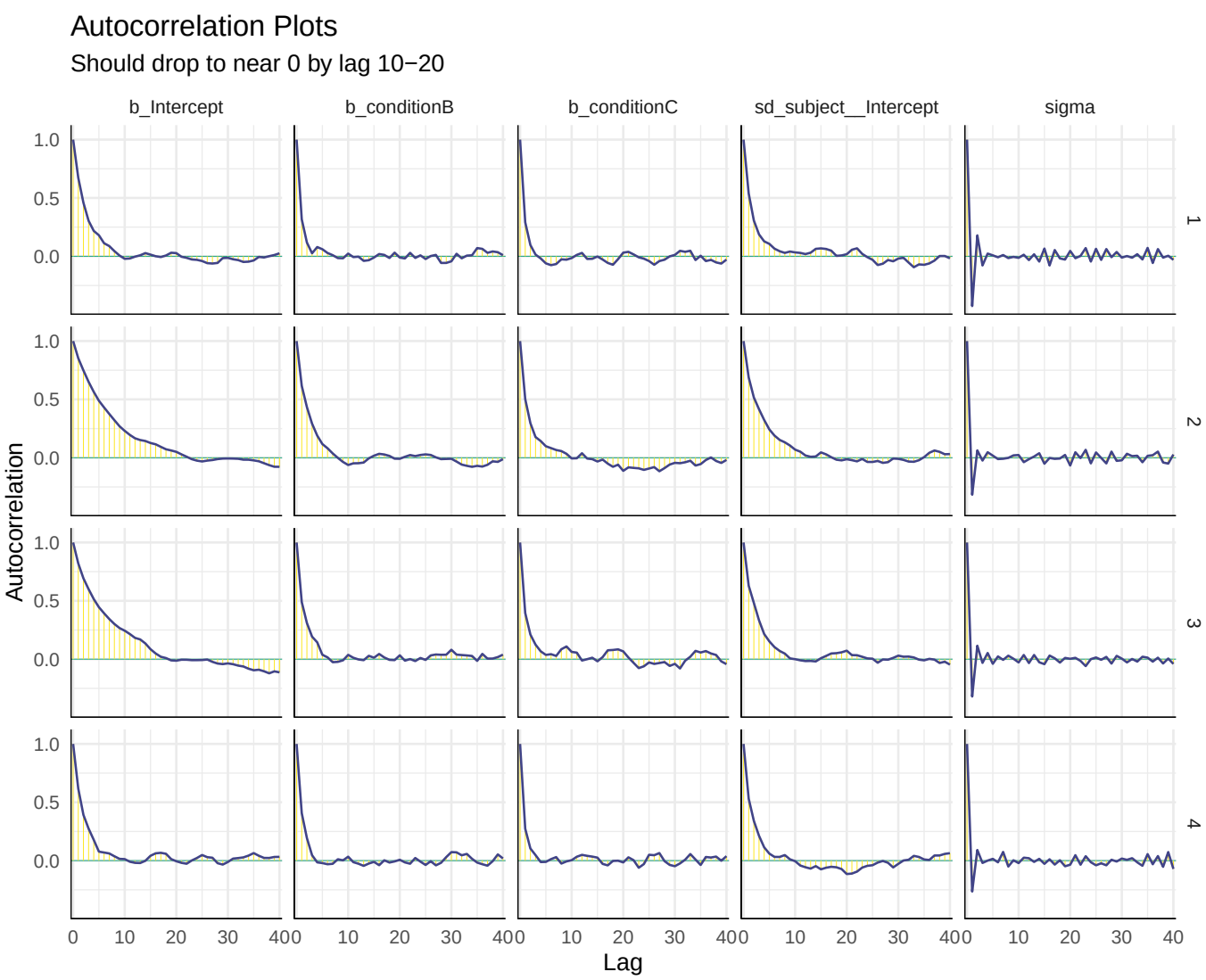
Autocorrelation measures how similar consecutive MCMC samples are.

- **Lag 1 autocorrelation**: Correlation between sample t and sample $t + 1$
- **Lag k autocorrelation**: Correlation between sample t and sample $t + k$

Why it matters: - High autocorrelation → samples contain redundant information - Low autocorrelation → chains explore efficiently - Affects ESS: higher autocorrelation → lower ESS

What we want: - Autocorrelation drops to near 0 by lag 10-20 - If still high at lag 50+: chains are mixing poorly

8.2 Autocorrelation Plots

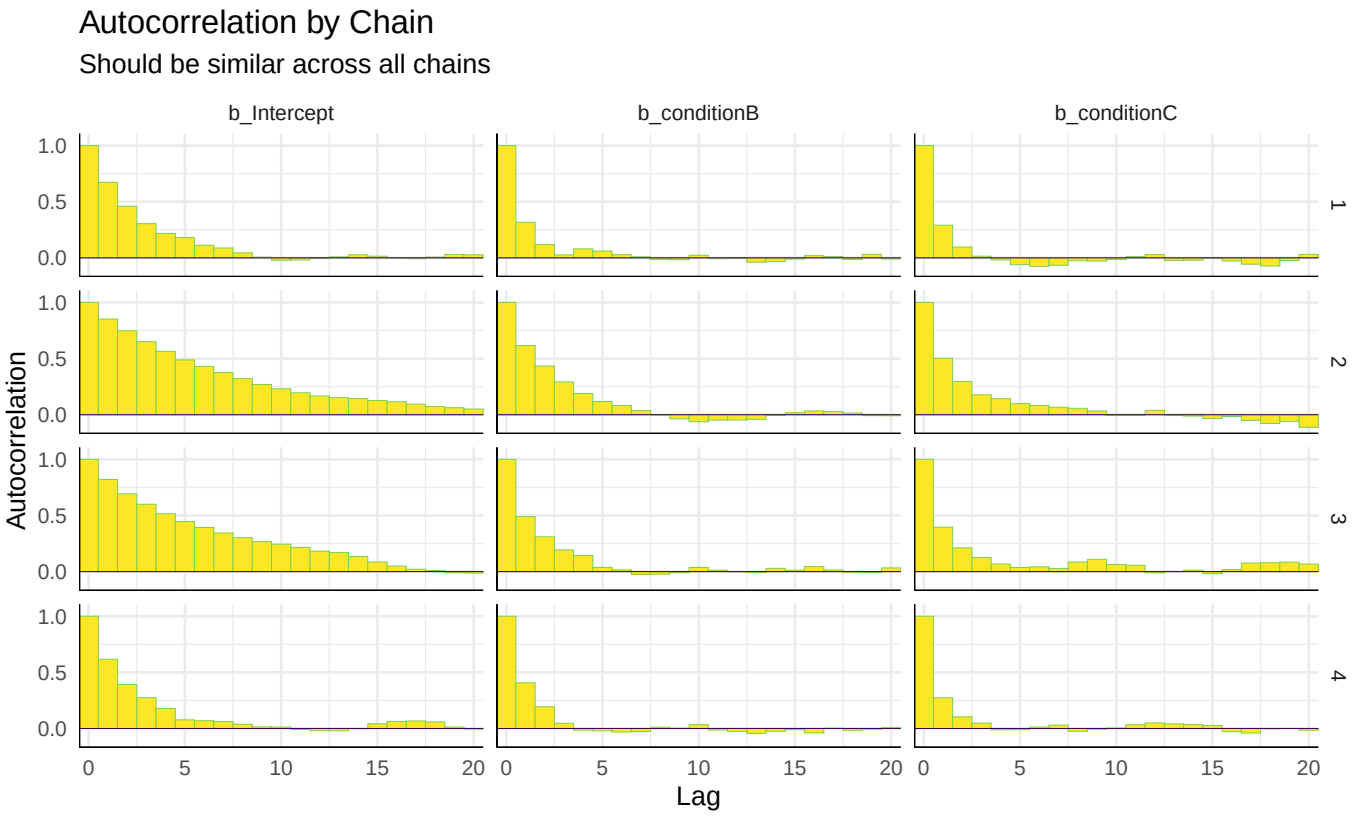


Interpreting ACF Plots

Good (low autocorrelation): - Drops below 0.1 by lag 5-10 - Fluctuates around 0 for higher lags - No systematic patterns
Problematic (high autocorrelation): - Stays above 0.3 beyond lag 20 - Slow exponential decay - ESS will be low

Solutions: - Increase iterations (more samples compensate for correlation) - Thin chains (keep every k-th sample, though less efficient than more iterations) - Reparameterize model (e.g., center predictors)

8.3 Autocorrelation by Chain



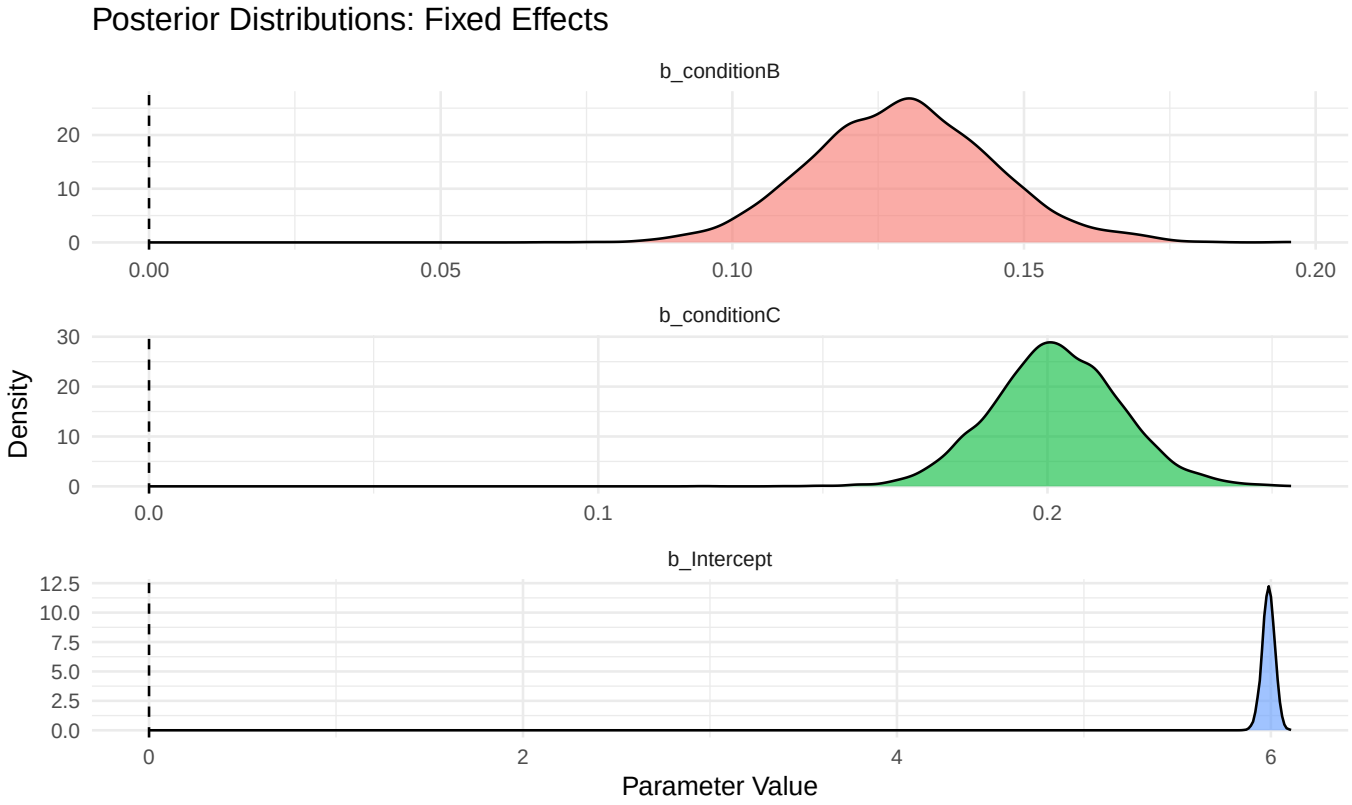
9 Check 5: Substantive Sense

9.1 Do Parameter Values Make Sense?

Beyond statistical convergence, parameters should make **substantive sense** based on:

1. **Domain knowledge**: Do values align with what we know about the phenomenon?
2. **Scale expectations**: Are effects plausible given measurement scale?
3. **Sign expectations**: Are effects in the expected direction?
4. **Relative magnitudes**: Are effect sizes reasonable compared to variability?

9.2 Examine Posterior Distributions



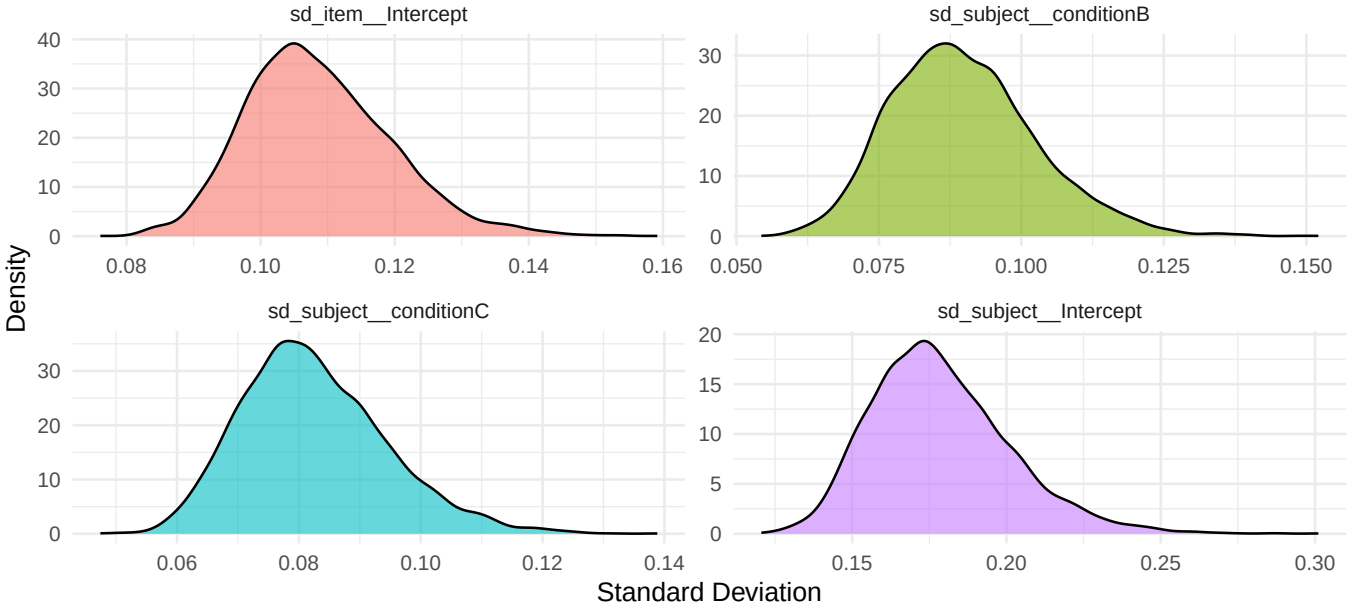
9.3 Interpret Fixed Effects

```
            Estimate  Est.Error       Q2.5     Q97.5
Intercept  5.9870814 0.03338817 5.91997791 6.0514296
conditionB 0.1291542 0.01548296 0.09987102 0.1607197
conditionC 0.2019316 0.01435443 0.17409168 0.2305139

=== SUBSTANTIVE INTERPRETATION ===

1. Intercept (Condition A baseline):

   Estimate: 5.987 log-ms → exp( 5.99 )  398 ms

    Reasonable RT for linguistic stimuli (300-600ms typical)

2. Condition B effect:

   Estimate: 0.129 log units

   →  13.8 % change in RT

    Moderate effect, typical for experimental manipulations

3. Condition C effect:

   Estimate: 0.202 log units

   →  22.4 % change in RT
```

```
Larger than B, consistent with hypothesized ordering
```

9.4 Examine Random Effects Variability

Posterior Distributions: Random Effects SDs




```
=== RANDOM EFFECTS VARIABILITY ===

$item
$item$sd
           Estimate  Est.Error       Q2.5     Q97.5
Intercept 0.1086299 0.01079036 0.09043753 0.1323893


$subject
$subject$sd
             Estimate  Est.Error       Q2.5     Q97.5
Intercept  0.17862168 0.02246552 0.14231358 0.2294753
conditionB 0.08987458 0.01269141 0.06861088 0.1177892
conditionC 0.08242443 0.01185874 0.06259936 0.1091452

$subject$cor
, , Intercept

             Estimate Est.Error       Q2.5     Q97.5
Intercept  1.00000000 0.0000000  1.0000000 1.0000000
conditionB 0.06041412 0.1655031 -0.2805700 0.3747177
conditionC 0.10294729 0.1661161 -0.2213175 0.4220262

, , conditionB
```

```
              Estimate Est.Error      Q2.5     Q97.5
Intercept   0.06041412 0.1655031 -0.280570 0.3747177
conditionB  1.00000000 0.0000000  1.000000 1.0000000
conditionC -0.08781520 0.1815400 -0.441408 0.2595498

, , conditionC

             Estimate Est.Error       Q2.5     Q97.5
Intercept   0.1029473 0.1661161 -0.2213175 0.4220262
conditionB -0.0878152 0.1815400 -0.4414080 0.2595498
conditionC  1.0000000 0.0000000  1.0000000 1.0000000


$subject$cov
, , Intercept

               Estimate   Est.Error          Q2.5       Q97.5
Intercept  0.0324102780 0.008416199  0.020253155 0.052658915
conditionB 0.0009210763 0.002855242 -0.005079781 0.006526851
conditionC 0.0014989473 0.002631428 -0.003461325 0.006965401

, , conditionB

                Estimate   Est.Error          Q2.5       Q97.5
Intercept   0.0009210763 0.002855242 -0.005079781 0.006526851
conditionB  0.0082384725 0.002380799  0.004707453 0.013874286
conditionC -0.0006100925 0.001427285 -0.003496026 0.002183864

, , conditionC

                Estimate   Est.Error          Q2.5       Q97.5
Intercept   0.0014989473 0.002631428 -0.003461325 0.006965401
conditionB -0.0006100925 0.001427285 -0.003496026 0.002183864
conditionC  0.0069343818 0.002044330  0.003918680 0.011912665



$residual__
$residual__$sd
  Estimate   Est.Error      Q2.5    Q97.5
 0.2010077 0.001683724 0.1977314 0.204301
```

Table 1: Random Effects Variance Components

Component	Estimate	95% CI Lower	95% CI Upper	Interpretation
Subject: Intercept SD	0.179	0.142	0.229	Between-subject variation in baseline RT
Subject: Condition slope SD	NA	NA	NA	Between-subject variation in condition effect
Item: Intercept SD	0.109	0.090	0.132	Between-item variation in difficulty
Residual SD	0.201	0.198	0.204	Within-subject/item unexplained variation

```
All SDs are positive and substantively reasonable
```

Substantive Sense Checklist

For your domain (psycholinguistics, phonetics, etc.), ask:

1. **Scale**: Are values in the right ballpark?
   - RTs: 200-2000ms typical
   - Log-RTs: ~5.3-7.6 (exp scale)
   - Accuracy: 0-1 (binomial), logit scale -5 to +5
2. **Sign**: Are effects in the expected direction?
   - More difficult → slower/less accurate
   - Priming → faster RTs
3. **Magnitude**: Are effects plausible?
   - Strong manipulations: 10-30% RT change
   - Subtle effects: 2-10% change
   - Random effects: Usually smaller than residual SD
4. **Relative sizes**: Do comparisons make sense?
   - Fixed effect < random effect variation → individual differences dominate
   - SD(items) « SD(subjects) → consistent across items

If estimates seem unreasonable, check: - Model specification (wrong family, missing predictors) - Priors (too strong, pulling estimates away from data) - Data coding errors (e.g., RT in seconds instead of milliseconds)

10 Check 6: Doubling Iterations Test

10.1 Why Double Iterations?

If chains have truly converged: - Doubling iterations should give nearly identical parameter estimates - Credible intervals should have similar widths - Conclusions shouldn't change

If estimates change substantially: - Original model hadn't fully converged - Need even more iterations

10.2 Fit Model with Doubled Iterations

10.3 Compare Parameter Estimates

```
            Parameter Original_Est Doubled_Est Original_SE Doubled_SE
Intercept   Intercept    5.9870814   5.9862924  0.03338817 0.03178091
conditionB conditionB    0.1291542   0.1292009  0.01548296 0.01554502
conditionC conditionC    0.2019316   0.2026220  0.01435443 0.01423280
                Est_Diff Est_Diff_Pct  SE_Ratio
Intercept  -7.889914e-04   0.01317823 0.9518614
conditionB  4.665406e-05   0.03612275 1.0040083
conditionC  6.904251e-04   0.34191032 0.9915269
```

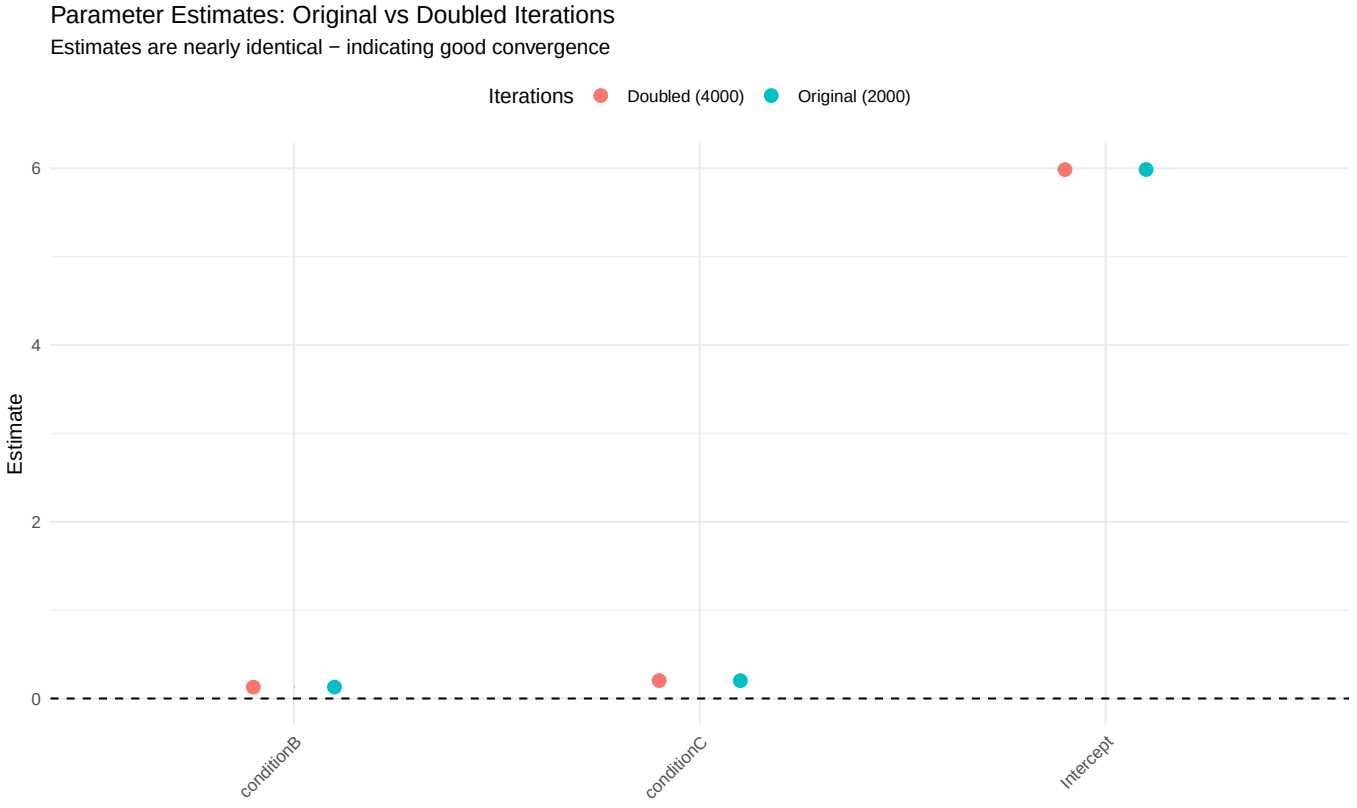
```
=== STABILITY ASSESSMENT ===
```

```
Max absolute difference in estimates: 8e-04
```

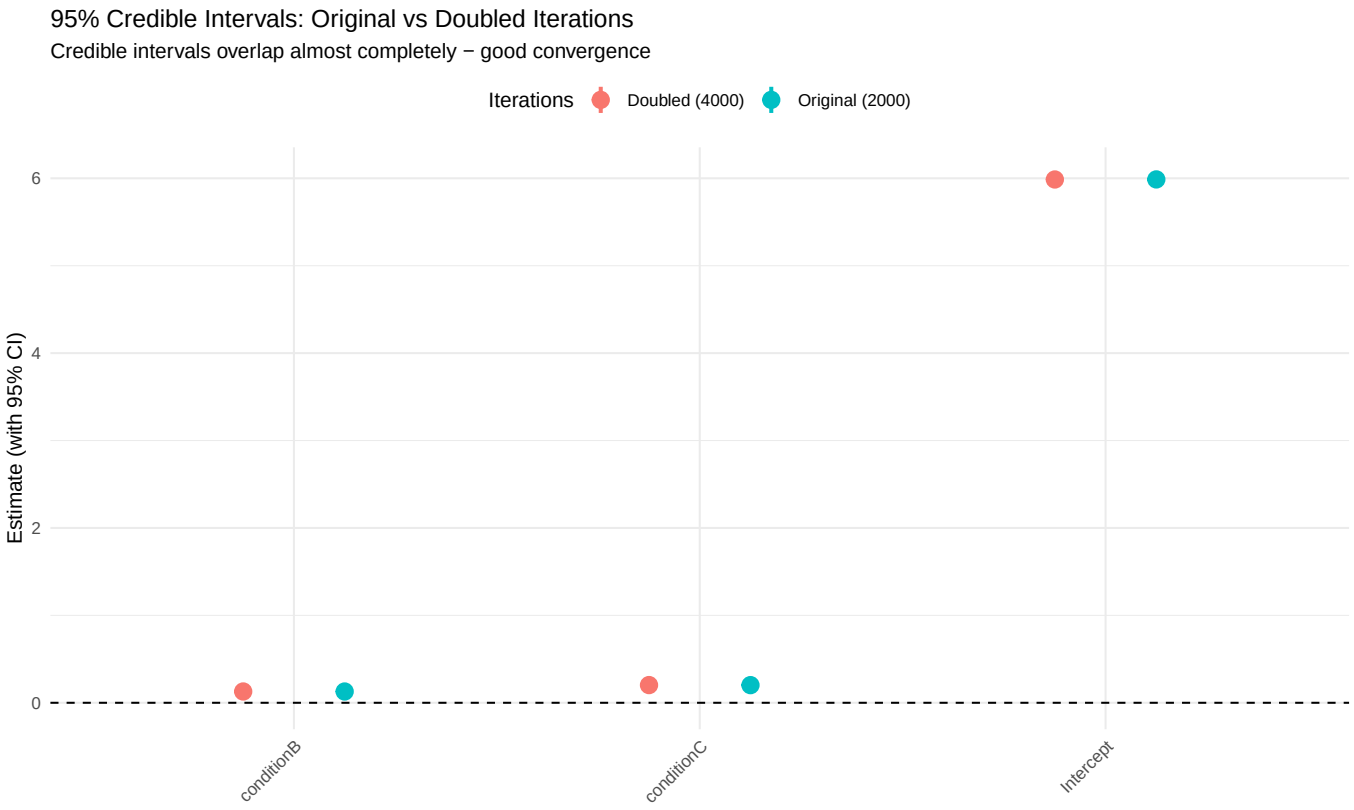
```
Max percentage change: 0.34 %
```

```
Median SE ratio (should be ~0.7 for 2× samples): 0.99
```

10.4 Visual Comparison



10.5 Compare Credible Intervals



! Iteration Doubling Decision Rules

If parameter estimates change by: - **< 1%**: Excellent stability, original iterations sufficient - **1-5%**: Acceptable, convergence achieved - **5-10%**: Marginal, consider using doubled model - **> 10%**: Not converged, use even more iterations or investigate
Standard error ratio (SE_doubled / SE_original): - **Expected**: ~0.71 (= $1/\sqrt{2}$, from doubling sample size) - **Actual < 0.71**: Even better precision than expected - **Actual > 0.71**: Less efficient sampling (high autocorrelation)

11 Example: Poorly Converged Model

11.1 Create a Convergence Problem

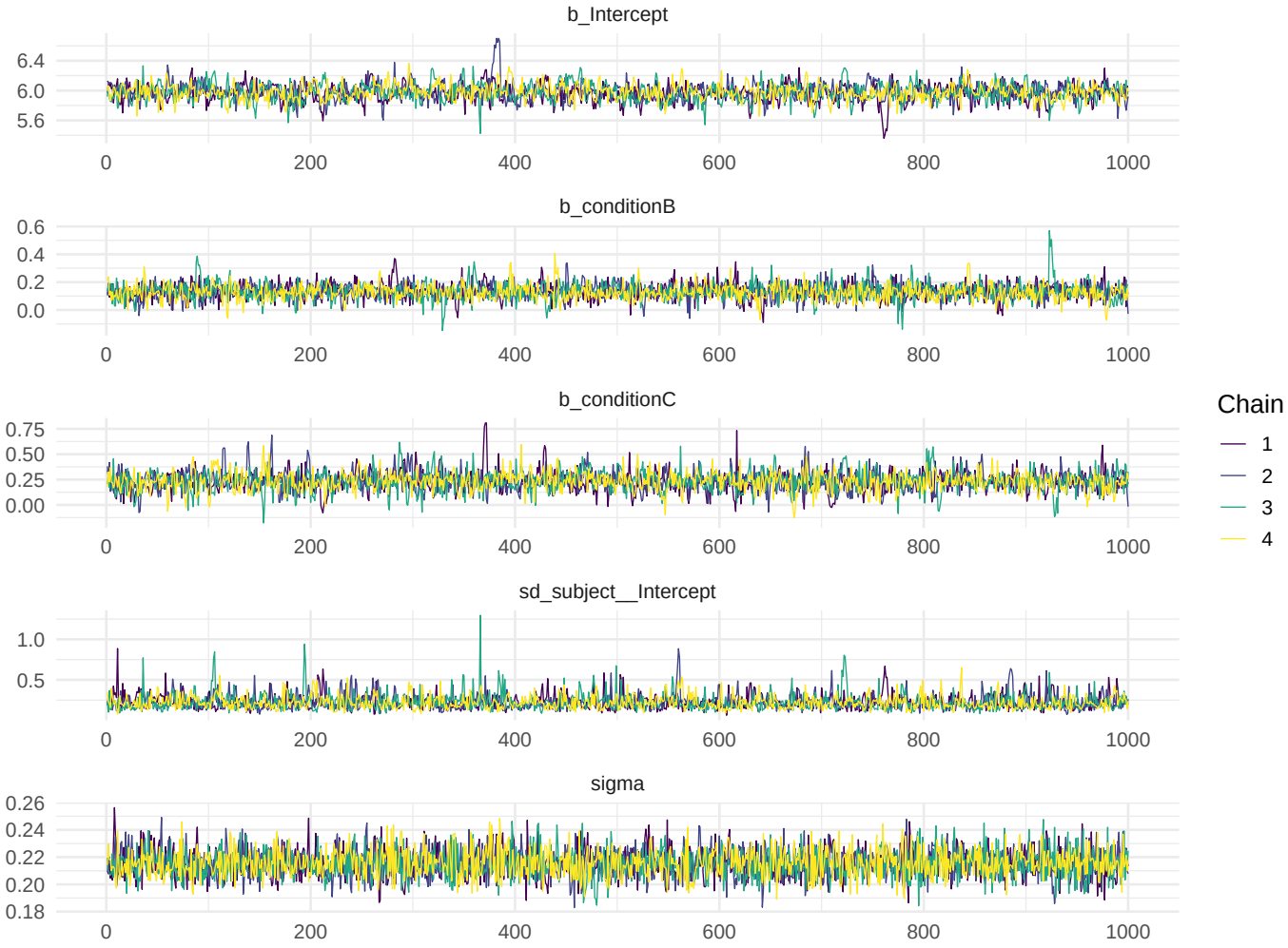
Let's deliberately create a model with convergence issues by using: - Very small sample size (insufficient data) - Weak priors (letting model wander) - Complex random effects structure (many parameters to estimate)

```
Small dataset: n = 240
```

```
Subjects: 8
```

```
Items: 10
```

11.2 Trace Plots: Poor Convergence



> ⚠ Warning Signs in These Traces
>
> Look for: - Chains exploring different regions (not overlapping) - Slow mixing (snake-like patterns) - One or more chains "stuck" at different values - Trends over time (non-stationarity)

11.3 Check R-hat and ESS

```
=== R-HAT (BAD MODEL) ===

Max R-hat: 1.0062

Parameters with R-hat > 1.01: 0

Parameters with highest R-hat:

r_subject[1,Intercept]           b_Intercept             Intercept
                1.0062                1.0059                1.0056
```

```
r_subject[3,Intercept] r_subject[8,Intercept] r_subject[7,Intercept]
                1.0051                 1.0050                 1.0042
r_subject[4,Intercept] r_subject[6,Intercept]                   lp__
                1.0040                 1.0040                 1.0036
r_subject[2,Intercept]
                1.0035
```

```
=== ESS (BAD MODEL) ===
```

```
Min ESS Bulk: 62
```

```
Min ESS Tail: Inf
```

```
Parameters with ESS < 400: 1
```

11.4 What Went Wrong?

```
=== DIAGNOSIS ===
```

```
1. Insufficient Data:
   - Only 240 observations
   - Trying to estimate 48 parameters
   - Ratio: 5 observations per parameter
   → PROBLEM: Not enough data to estimate all random effects precisely
2. Weak Priors:
   - Very vague priors don't constrain parameter space
   - With limited data, posterior poorly defined
   → SOLUTION: Use stronger, more informative priors
3. Complex Model Structure:
   - Random slopes for 2 conditions + correlations
   - Many variance components with little data
   → SOLUTION: Simplify to random intercepts only
```

11.5 Test: Double Iterations on Bad Model

Let's see what happens when we double iterations on the poorly converged model:

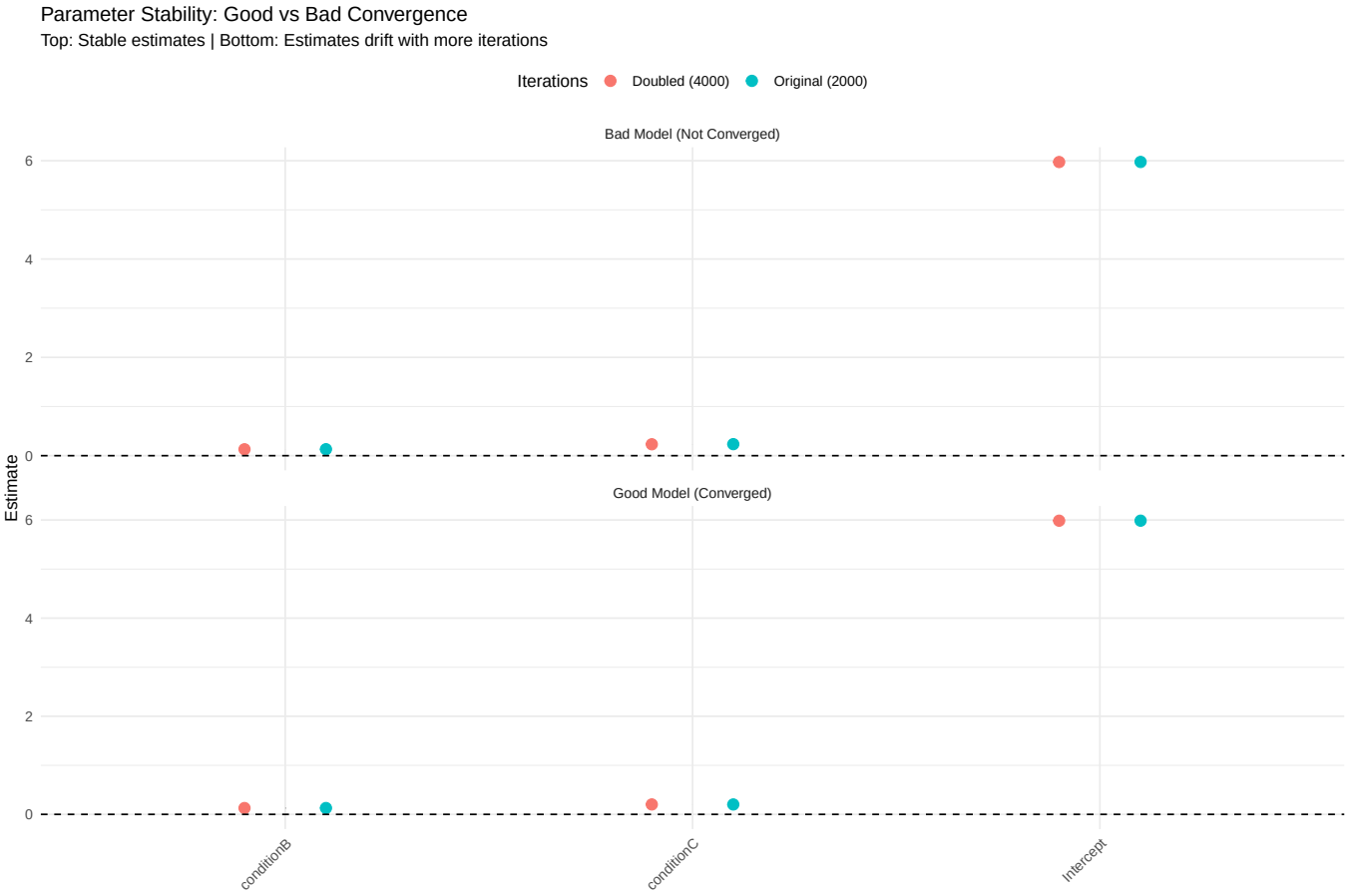
```
=== BAD MODEL: STABILITY ASSESSMENT ===
```

```
            Parameter Original_Est Doubled_Est Original_SE Doubled_SE
Intercept   Intercept    5.9748176   5.9733090  0.11456063 0.11241800
conditionB conditionB    0.1313419   0.1299990  0.05751411 0.05455722
conditionC conditionC    0.2342640   0.2321079  0.09600195 0.09238752
              Est_Diff Est_Diff_Pct  SE_Ratio
```

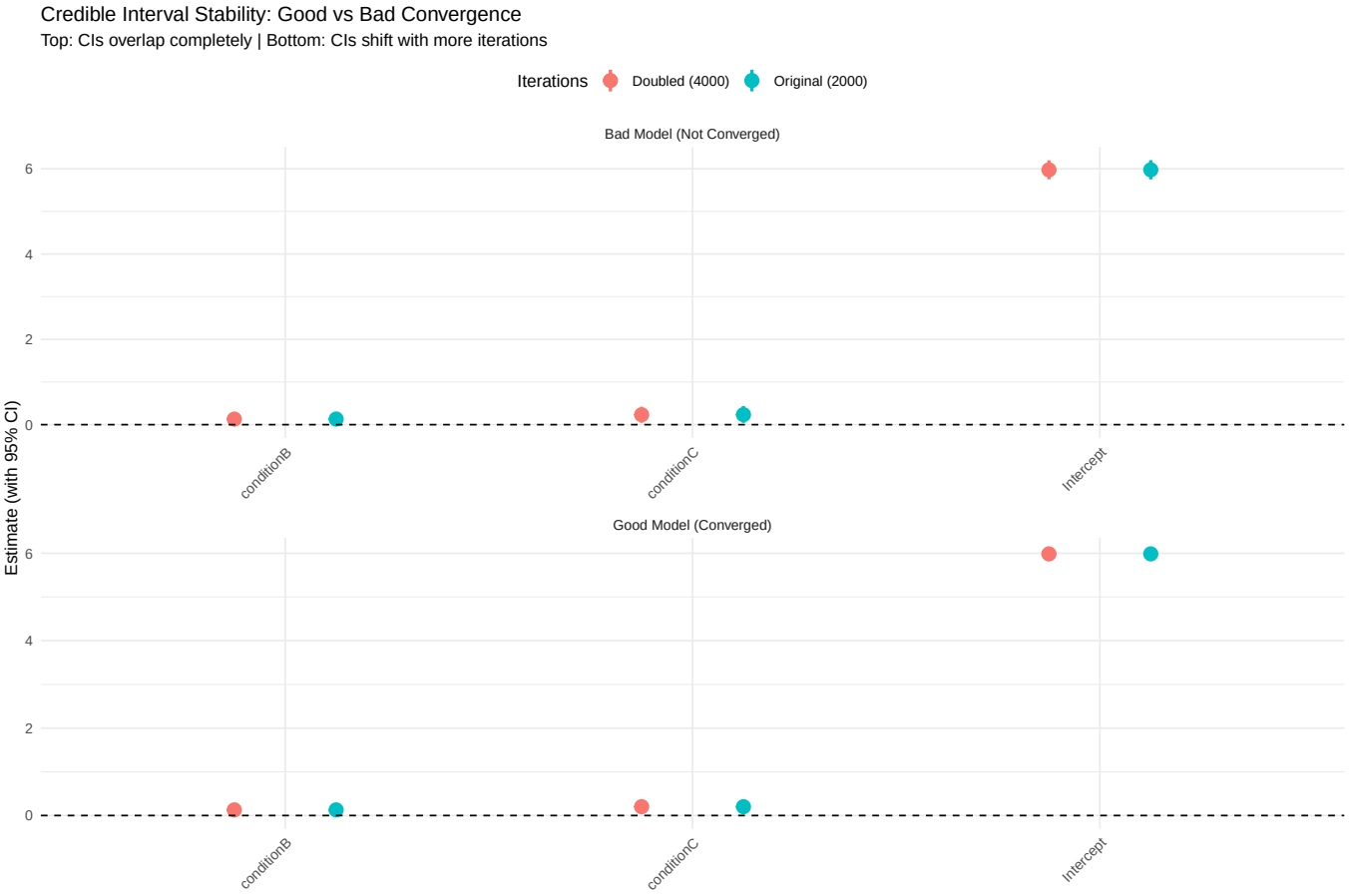
```
Intercept  -0.001508573   0.02524885 0.9812970
conditionB -0.001342911   1.02245400 0.9485884
conditionC -0.002156060   0.92035509 0.9623504

Max absolute difference: 0.0022

Max percentage change: 1.02 %

  Notice large percentage changes - model has NOT fully converged!
```

11.6 Visual Comparison: Good vs Bad Model

Now let's compare how the good (converged) and bad (non-converged) models behave when we double iterations:



11.7 Credible Intervals: Good vs Bad Model



! Key Takeaway from Comparison

Good model (top panels): - Estimates barely change when doubling iterations - Credible intervals overlap almost completely - Indicates chains have converged
Bad model (bottom panels): - Estimates shift substantially with more iterations - Credible intervals move to different locations - Indicates chains have NOT converged - need to fix the underlying issues
Lesson: Doubling iterations alone won't fix a fundamentally problematic model. Address root causes (data, priors, model structure) first.

Fixing Convergence Problems

Common causes and solutions:

1. **Insufficient data**
   - Simplify model (remove random slopes, interactions)
   - Use stronger priors to regularize
   - Collect more data
2. **Weak/improper priors**
   - Use informative priors based on domain knowledge

- Check prior predictive distributions
- Avoid completely flat priors

3. **Model misspecification**
   - Check data coding (units, factors, centering)
   - Verify distributional assumptions (family)
   - Consider transformations (log, logit, etc.)
4. **Sampler struggles (divergent transitions)**
   - Increase `adapt_delta` (e.g., 0.95 or 0.99)
   - Reparameterize (non-centered random effects)
   - Increase `max_treedepth`
5. **Just needs more time**
   - Double or quadruple iterations
   - Increase warmup period
   - Run longer chains (but check other issues first!)

12 Summary: Complete Convergence Checklist

12.1 Step-by-Step Workflow

After fitting any Bayesian model, work through this checklist:

12.1.1 1. Automatic Warnings

```
# brms automatically prints warnings
summary(model)
```

Look for: - "The largest R-hat is..." - "Bulk/Tail ESS is too low..." - "There were X divergent transitions..." - "Rhat > 1.01 for..."

12.1.2 2. Trace Plots

```
plot(model, variable = c("b_", "sd_"), regex = TRUE)
# Or with bayesplot:
mcmc_trace(as_draws_array(model), pars = c("b_Intercept", ...))
```

Check: - Do all chains overlap completely? - No drift or trends over time? - "Hairy caterpillar" appearance?

12.1.3 3. R-hat < 1.01

```
max(rhat(model))
# Should be < 1.01 for ALL parameters
```

12.1.4 4. ESS > 400

```
min(ess_bulk(model))
min(ess_tail(model))
# Both should be > 400 for ALL parameters
```

12.1.5 5. Low Autocorrelation

```
mcmc_acf(as_draws_array(model), pars = c(...))
# Should drop to near 0 by lag 10-20
```

12.1.6 6. Substantive Sense

```
summary(model)
fixef(model)
ranef(model)
```

Ask: - Are parameter values in reasonable ranges? - Do effect signs match expectations? - Are magnitudes plausible?

12.1.7 7. Stability Test (Optional)

```
# Refit with doubled iterations
model_doubled <- update(model, iter = 4000, warmup = 2000)
# Compare estimates - should be < 5% difference
```

12.2 Decision Matrix

Check	Status	Action
Warnings	None	Proceed
Warnings	Divergent transitions	Increase `adapt_delta = 0.95+`
Warnings	Max treedepth	Increase `max_treedepth = 12+`
Trace plots	Well mixed	Proceed
Trace plots	Poor mixing	Increase iterations, check model
R-hat	All < 1.01	Proceed
R-hat	Any > 1.01	Don't trust results! Increase iterations
ESS	All > 400	Proceed
ESS	Any < 400	Increase iterations proportionally
Autocorrelation	Drops quickly	Proceed
Autocorrelation	Stays high	Increase iterations or reparameterize
Substantive	Makes sense	Proceed
Substantive	Implausible	Check model specification and data
Doubled iterations	$< 5\%$ change	Stable, proceed
Doubled iterations	$> 10\%$ change	Not converged, use more iterations

12.3 Quick Reference: Fixing Common Problems

12.3.1 Problem: High R-hat, low ESS

```
# Solution 1: More iterations
model <- brm(..., iter = 4000, warmup = 2000)

# Solution 2: Check for divergent transitions
# If yes, increase adapt_delta
model <- brm(..., control = list(adapt_delta = 0.95))
```

12.3.2 Problem: Divergent transitions

```
# Increase adapt_delta (makes sampler more careful)
model <- brm(..., control = list(adapt_delta = 0.95))

# If still divergent, try 0.99
model <- brm(..., control = list(adapt_delta = 0.99))

# Consider non-centered parameterization for random effects
# (Advanced: modify model formula or use custom Stan code)
```

12.3.3 Problem: Max treedepth warnings

```
# Increase max treedepth
model <- brm(..., control = list(max_treedepth = 12))
```

12.3.4 Problem: Implausible parameter values

```
# 1. Check data coding
summary(data)
str(data)

# 2. Use more informative priors
priors <- c(
  prior(normal(6, 1), class = Intercept),  # Stronger prior
  prior(normal(0, 0.5), class = b)          # More regularization
)

# 3. Simplify model
model <- brm(y ~ x + (1 | subject),  # Remove random slopes
             ...)
```

12.4 When Everything Looks Good

Once all checks pass:

You can trust your posterior estimates!

Proceed to: - ROPE analysis (Module 06) - Effect estimation with emmeans/marginaleffects (Module 06) - Bayes Factors (Module 07) - Report results in your manuscript

12.5 Final Thoughts

Convergence diagnostics are not optional!

- Always check convergence before interpreting results
- If in doubt, run longer chains
- Document your diagnostics in supplementary materials
- Make convergence checks part of your standard workflow

Remember: > "Garbage in, garbage out" applies to MCMC too. If your chains haven't converged, your inferences are meaningless—no matter how sophisticated your analysis!

13 Exercises

13.1 Exercise 1: Check Your Own Model

Take a model you've fitted in previous modules:

1. Run all convergence diagnostics
2. Create a trace plot for key parameters
3. Check R-hat and ESS values
4. Verify parameters make substantive sense
5. Document your findings

13.2 Exercise 2: Fix a Convergence Problem

Use the poorly converged model from this module:

1. Identify the specific problems
2. Propose 2-3 solutions
3. Implement one solution and recheck diagnostics
4. Compare results before/after

13.3 Exercise 3: Sensitivity to Iterations

Fit the same model with different iteration settings: - 1000 iterations (500 warmup) - 2000 iterations (1000 warmup) - 4000 iterations (2000 warmup)

Compare: - Parameter estimates - Credible interval widths - ESS values - Computation time

When do estimates stabilize?

14 Literature and Resources

14.1 Essential Reading

Convergence diagnostics: - Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., & Bürkner, P. C. (2021). Rank-normalization, folding, and localization: An improved $\hat{R}$ for assessing convergence of MCMC. *Bayesian Analysis*, 16(2), 667-718. - Modern R-hat diagnostic (current standard)

- Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4), 457-472.
  - Original R-hat diagnostic

Effective sample size: - Vehtari et al. (2021) [same as above] - ESS bulk and tail diagnostics

General MCMC diagnostics: - Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). *Bayesian Data Analysis* (3rd ed.). CRC Press. - Chapter 11: Basics of Markov Chain Simulation - Gold standard textbook

Practical guidance: - Gabry, J., Simpson, D., Vehtari, A., Betancourt, M., & Gelman, A. (2019). Visualization in Bayesian workflow. *Journal of the Royal Statistical Society A*, 182(2), 389-402. - Comprehensive workflow including convergence checks

14.2 Software Documentation

- **brms**: `help("brms-package")`, especially MCMC settings
- **bayesplot**: `vignette("visual-mcmc-diagnostics")`
- **posterior**: `vignette("posterior")`
- **Stan**: https://mc-stan.org/docs/reference-manual/

14.3 Advanced Topics

- **Non-centered parameterizations**: Betancourt, M., & Girolami, M. (2015). Hamiltonian Monte Carlo for hierarchical models.
- **Divergent transitions**: https://mc-stan.org/docs/2_27/reference-manual/divergent-transitions.html

---

Next Module: With converged models in hand, you're ready to report your results!