

# Package ‘fibr’

June 29, 2026

**Type** Package

**Title** Prior-Fraction Diagnostics for Hierarchical Models

**Version** 0.1.1

**Description** Computes the prior fraction, the per-group pooling or shrinkage factor, for hierarchical models, including directly from 'brms' fits. For each group-level coefficient the prior fraction is the share of the posterior precision contributed by the shrinkage prior relative to the likelihood; values near one indicate a coefficient that is prior-dominated (the centring/non-centring funnel regime), values near zero indicate a likelihood-dominated coefficient that is well identified from the data. These quantities are invisible to standard convergence diagnostics such as R-hat and effective sample size, and they indicate where a non-centred reparameterisation is likely to help. A companion advisor reports the same decomposition for changepoint random effects fitted with 'smoothbp'. The underlying geometry (the Fisher-metric connection on the base-fiber split, for which this connection is flat so the obstruction is statistical rather than geometric) is described in Bindoff (2026) <[doi:10.5281/zenodo.20724550](https://doi.org/10.5281/zenodo.20724550)>; code reproducing the paper is in the package's source repository.

**License** MIT + file LICENSE

**URL** <https://github.com/ABindoff/fibr>,  
<https://doi.org/10.5281/zenodo.20724550>

**BugReports** <https://github.com/ABindoff/fibr/issues>

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Depends** R (>= 4.1.0)

**Imports** posterior (>= 1.4.0), ggplot2

**Suggests** brms, smoothbp, cmdstanr, testthat (>= 3.0.0)

**Additional\_repositories** <https://stan-dev.r-universe.dev>

**Config/testthat/edition** 3

**Config/rextendr/version** 0.5.0

**SystemRequirements** Cargo (Rust's package manager), rustc >= 1.65.0, xz

**NeedsCompilation** no

**Author** Aidan D Bindoff [aut, cre] (ORCID:  
<<https://orcid.org/0000-0002-0943-2702>>)

**Maintainer** Aidan D Bindoff <aidan.bindoff@utas.edu.au>

**Repository** CRAN

**Date/Publication** 2026-06-29 12:50:19 UTC

## Contents

prior_fraction . . . . .	2
smoothbp_advisor . . . . .	4
<b>Index</b>	<b>6</b>

---

prior_fraction	<i>Per-coordinate prior fraction (shrinkage / pooling factor)</i>
----------------	---

---

## Description

For each group-level (random-effect) coordinate  $\alpha_j$  of a fitted hierarchical model, the *prior fraction*

$$\pi_j = \frac{\text{prior precision}}{\text{prior precision} + \text{likelihood information}} = \frac{1/\sigma^2}{G_{FF,j}}$$

is the share of that coordinate's posterior precision contributed by the prior rather than by its own data. It is the classical shrinkage / pooling factor (Gelman and Pardoe, 2006) and the per-group information ratio of Betancourt and Girolami (2015).

**Interpretation.**  $\pi_j \approx 1$  means the coordinate is *prior-dominated*: its posterior is essentially the prior pushed through shrinkage, so the estimate is mostly regularisation toward the population and should not be over-interpreted unless the prior is one you would defend.  $\pi_j \approx 0$  means the data speak. This is a *prior-influence* report, not a convergence diagnostic, and it is read-only: nothing is reparameterised or refit.

**Scope and limits.** The estimate is exact for the common GLM families (gaussian, bernoulli, binomial, poisson, negbinomial) with the standard (. . . | g) random-effect structure. For *correlated* random effects it reports the per-marginal fraction (using each coefficient's own sd); the full story there is the eigenvalues of a matrix pooling factor, and a message is emitted. Coordinates with no data (n\_obs == 0) are flagged with  $\pi = 1$ . Smooths and GP terms have correlated coordinates and should be read with that caveat. The diagnostic says nothing about multimodality, aliasing, or likelihood mis-specification.

**Usage**

```
prior_fraction(x, ...)

## Default S3 method:
prior_fraction(x, lik_information, labels = NULL, ...)

## S3 method for class 'brmsfit'
prior_fraction(x, ndraws = 200L, ...)
```

**Arguments**

x	A fitted model. Methods are provided for brmsfit objects and a default method taking the prior precision directly (see Details).
...	Passed to methods.
lik_information	Numeric vector of per-coordinate likelihood information.
labels	Optional data frame of label columns (recycled / bound to output).
ndraws	Number of posterior draws to subsample when forming the posterior-mean linear predictor (for speed). Default 200.

**Value**

A data frame of class `fibr_prior_fraction` with one row per coordinate and columns `group`, `coef`, `level`, `n_obs`, `prior_sd`, `lik_info`, `pi`. Has `print` and `plot` methods.

**Methods (by class)**

- `prior_fraction(default)`: Manual path for any model. Supply the per-coordinate prior precision  $\times (1/\sigma^2)$  and the per-coordinate likelihood information `lik_information` ( $G_{FF,j} - 1/\sigma^2$ ); optionally a `labels` data frame to carry through. Use this to validate against the closed-form GLMM or to handle Stan fits this package does not parse.
- `prior_fraction(brmsfit)`: Adapter for **brms** fits. Extracts the random-effect structure, per-coordinate prior SDs, and the family information at the posterior mean, and returns the per-coordinate prior fraction. Requires **brms**.

**References**

Gelman and Pardoe (2006), *Technometrics* 48(2):241–251. Betancourt and Girolami (2015), in *Current Trends in Bayesian Methodology with Applications*.

**Examples**

```
## Manual path (no model fit needed): supply the per-coordinate prior
## precision (1/sigma^2) and likelihood information (sum of per-observation
## Fisher information). This is the closed-form GLMM prior fraction.
sigma <- 1.5
lik <- c(0.2, 1.0, 5.0) # e.g. sum p(1-p) for three groups
prior_fraction(1 / sigma^2, lik_information = lik)
```

```
## brms path: which group-level estimates are prior-dominated?
if (requireNamespace("brms", quietly = TRUE)) {
  set.seed(42)
  dat <- data.frame(y = rpois(30, 5), site = rep(letters[1:10], 3))
  fit <- brms::brm(y ~ 1 + (1 | site), data = dat,
                  family = stats::poisson(), iter = 500, chains = 1,
                  refresh = 0)
  pf <- prior_fraction(fit)
  pf                                     # summary: how many coordinates have pi > 0.8
  plot(pf)                               # pi vs. number of observations
}
```

---

smoothbp\_advisor

*Parameterisation advisor for smoothbp changepoint random effects*


---

## Description

For each random effect on a changepoint location ( $\omega_{k,g}$  = per-group deviation from the population changepoint at breakpoint  $k$ ), computes the **Fisher information decomposition** at a subsample of posterior draws.

The key quantity is `prior_frac`:

$$\text{prior\_frac}_{k,g} = \frac{G_{\text{prior}}}{G_{\text{prior}} + G_{\text{lik}}}$$

where

$$G_{\text{prior}} = \sigma_{\text{re},k}^{-2}, \quad G_{\text{lik}} = \sigma^{-2} \sum_{i:\text{group}=g} \left( \frac{\partial \mu_i}{\partial \omega_{k,i}} \right)^2$$

- `prior_frac` → 1: prior dominates – group changepoints are poorly identified from data relative to the shrinkage prior. The sampler is in the funnel regime and non-centred reparameterisation would help.
- `prior_frac` → 0: likelihood dominates – centred parameterisation is efficient and mixing should be adequate.
- Mixed: flag individual groups for attention.

## What to do with the results:

When `prior_frac` is high, re-fit with `reparameterise = "omega"`:

```
fit_nc <- smoothbp(..., reparameterise = "omega")
fit_nc_ss <- smoothbp_ss(..., reparameterise = "omega")
```

This activates the non-centred HMC parameterisation in the Rust sampler:  $z[j] = \text{beta\_om}[j] / \text{sigma\_re\_om}[k]$  is sampled with an  $N(0, 1)$  prior, and  $\text{beta\_om}[j] = z[j] * \text{sigma\_re\_om}[k]$  is reconstructed automatically. The  $\text{sigma\_re\_om}$  Gibbs step and the stored draws are unchanged – output is in the original (centred) parameterisation for easy interpretation.

Additional options if `reparameterise = "omega"` is insufficient:

1. Increase `warmup` and `iter`.
2. Check `fit$n_divergent` – many divergences confirm a remaining funnel.
3. Fix the changepoint for poorly-identified groups using `omega = list(fixed(value))`.

The `prior_frac` values quantify the severity: values above 0.8 indicate a serious funnel; 0.6-0.8 suggests moderate difficulty worth addressing.

The gradient is computed analytically from the sigmoid smooth-transition likelihood:

$$\frac{\partial \mu_i}{\partial \omega_{k,i}} = - \left[ \delta_{k,i} \sigma_{ki} (1 + d_{ki} \rho_{ki} (1 - \sigma_{ki})) + b_{1,i} \mathbf{1}_{k=1} \right]$$

where  $d_{ki} = \tau_i - \omega_{k,i}$  and  $\sigma_{ki} = \text{logistic}(d_{ki} \rho_{ki})$ .

## Usage

```
smoothbp_advisor(fit, n_draws = 200L, threshold_nc = 0.6, threshold_c = 0.4)
```

## Arguments

<code>fit</code>	A <code>smoothbp_fit</code> from <code>smoothbp::smoothbp()</code> or <code>smoothbp::smoothbp_ss()</code> , with at least one <code>omega</code> random effect ( <code>omega = list(~ 1 + (1   group))</code> ).
<code>n_draws</code>	Number of posterior draws to evaluate the metric at (default 200; subsampled uniformly).
<code>threshold_nc</code>	Prior fraction above which non-centred is recommended (default 0.60).
<code>threshold_c</code>	Prior fraction below which centred is safe (default 0.40).

## Value

An S3 object of class `fibr_smoothbp_advice`. Contains one list element per breakpoint that has `omega` random effects, each with `prior_frac_mean`, `prior_frac_q05`, `prior_frac_q95`, `recommendation`, and `delta` (one entry per group). `delta[j]` is the recommended per-group NC mixing fraction  $\delta_j = \sqrt{\text{prior\_frac}_j}$ . Print and plot methods included.

# Index

`prior_fraction`, [2](#)

`smoothbp_advisor`, [4](#)