



MODEL SPECIFICATION

<code>drm_formula()</code> / <code>bf()</code>	Combine one formula per distributional parameter. <code>bf()</code> is the short alias.
<code>drmTMB()</code>	Fit a univariate or bivariate distributional model. The native TMB engine is the default.
<code>drm_control()</code>	Set optimizer controls and fitted-object storage or large-data options.
<code>meta_V()</code>	Supply known sampling variance or covariance for Gaussian meta-analytic models.
<code>meta_vcov_bivariate()</code>	Assemble the row-paired known covariance matrix for two-response meta-analysis.
<code>miss_control()</code>	Choose the documented missing-response or missing-predictor policy.
<code>mi()</code>	Mark a predictor whose missing values are modelled jointly with the response.
<code>impute_model()</code>	Specify the family and formula for a modelled missing predictor.

PAIRED-OUTCOME ASSOCIATION

<code>biv_associate()</code>	Fit two margins for a pair class listed in the association guide, then construct their frozen-margin association.
<code>associate_pairs()</code>	Estimate within-row association from two fitted, aligned margins on a route listed in the association guide.
<code>latent_normal()</code>	Declare the Gaussian-copula kernel used by the paired-outcome association stage.
<code>association()</code>	Extract latent-normal eta from a separate <code>drm_pair_association</code> object. It is not residual <code>rho12</code> or a correlation from <code>corpairs()</code> .

OPTIONAL JULIA EXTRACTOR

<code>rho_latent()</code>	Extract the latent/link-scale correlation from a compatible legacy Julia-engine object. Julia fitting is not a current route for a new analysis.
---------------------------	--

RESPONSE AND PREDICTOR FAMILIES

<code>student()</code>	Student-t location-scale-shape models for heavy-tailed continuous responses.
<code>skew_normal()</code>	Skew-normal location-scale-shape models.
<code>lognormal()</code>	Positive continuous responses on the lognormal scale.
<code>tweedie()</code>	Non-negative semicontinuous responses that may include exact zeroes.
<code>beta()</code>	Strict proportions inside (0, 1).
<code>zero_one_beta()</code>	Continuous proportions with structural exact zeroes or ones.
<code>beta_binomial()</code>	Success counts with a known denominator and extra-binomial variation.
<code>categorical()</code>	An unordered categorical missing-predictor model in <code>impute_model()</code> .
<code>cumulative_logit()</code>	Ordered categorical responses or ordered missing-predictor models.
<code>nbinom2()</code>	Negative-binomial count models with quadratic mean-variance scaling.
<code>truncated_nbinom2()</code>	Positive or hurdle negative-binomial count models.
<code>biv_gaussian()</code>	Two Gaussian responses with separate means, scales, and residual <code>rho12</code> .
<code>biv_lognormal()</code>	Two lognormal responses with residual <code>rho12</code> on the log-response scale.
<code>biv_student()</code>	Two heavy-tailed Student-t responses with a shared degrees-of-freedom parameter.
<code>gaussian()</code> , <code>Gamma()</code> , <code>binomial()</code> , <code>poisson()</code>	Base-R families for documented Gaussian, positive, event-count, and count-response routes.

RANDOM AND STRUCTURED EFFECTS

<code>animal()</code>	Pedigree or supplied animal-model precision/covariance structure.
<code>phylo()</code>	A structured effect derived from one phylogenetic tree.
<code>phylo_interaction()</code>	A pair-level effect derived from two phylogenetic trees.
<code>spatial()</code>	Coordinate-based spatial structured effects.
<code>spatial_coords()</code>	Convert longitude and latitude to an explicitly chosen projected coordinate system.
<code>make_mesh()</code>	Build the fixed-kappa SPDE mesh and projection inputs used by mesh spatial effects.
<code>relmat()</code>	A structured effect from a user-supplied relationship or precision matrix.
<code>corpair()</code>	Mark one current <code>q=2</code> labelled ordinary or phylogenetic location-location correlation formula. Keep this distinct from residual <code>rho12 ~ ...</code>
<code>sd(group) ~ ..., sd1(group) ~ ..., sd2(group) ~ ...</code>	Formula-only syntax for Gaussian location random-effect SDs. These are not replacements for <code>stats::sd()</code> .
<code>drm_phylo_penalty()</code>	Define the documented penalized/MAP prior for weakly identified phylogenetic SDs or correlations. Penalized fits are MAP fits, not ordinary ML fits for LR tests or AIC comparisons.
<code>drm_phylo_penalty_sweep()</code>	Check whether a penalized phylogenetic coupling is data-informed or prior-shaped.

DISTRIBUTIONAL ASSESSMENT

<code>fitted_distribution()</code>	Access fitted per-row density, CDF, and quantile functions.
<code>exceedance()</code>	Compute conditional probability above or below a threshold.
<code>centile_chart()</code>	Plot conditional response centiles over one covariate.
<code>qq_plot()</code>	Plot randomized quantile residuals against normal quantiles.
<code>worm_plot()</code>	Detrend the QQ plot to reveal systematic departures.



CHECK AND EXTRACT - CORE

<code>check_drm()</code>	Run the model-aware convergence and identifiability checks. Start here after fitting.
<code>is_converged()</code>	Return one logical convergence summary for programmatic workflows.
<code>summary()</code>	Print coefficients, fit information, and optional intervals.
<code>coef()</code> / <code>fixef()</code>	Extract fixed-effect coefficient blocks; <code>fixef()</code> makes the mixed-model meaning explicit.
<code>ranef()</code>	Extract conditional modes for ordinary or structured random effects.
<code>sigma()</code>	Extract fitted residual or family-specific scale values.
<code>rho12()</code>	Extract within-observation residual correlation for bivariate Gaussian fits.
<code>corpairs()</code>	Extract labelled ordinary or structured latent-correlation summaries.
<code>structured_effects()</code>	List fitted structured-effect terms and their model roles.

FORMULA ANATOMY

<code>drm_formula(mu = trait ~ temp, sigma = ~ temp)</code>	Model the expected trait and its residual scale with separate linear predictors.
<code>nu = ~ 1, zi = ~ habitat, hu = ~ habitat</code>	Add a shape, zero-inflation, or hurdle component where the chosen family supports it.
<code>rho12 = ~ environment</code>	Model coscale: the within-observation residual correlation of a bivariate Gaussian fit.
<code>sd(group) ~ x, sd1(group) ~ x, sd2(group) ~ x</code>	Model Gaussian location random-effect SDs; use level = "phylogenetic" for the documented structured route.
<code>corpair(label) ~ x</code>	Formula-marker syntax for a current q=2 labelled ordinary or phylogenetic location-location correlation block; predictors must be constant within group.

CHECK AND EXTRACT - FIT

<code>imputed()</code>	Summarize fitted values for a modelled missing predictor.
<code>fitted()</code>	Extract fitted response means for the original fitted rows.
<code>residuals()</code>	Compute the documented response, Pearson, or randomized-quantile residuals.
<code>weights()</code>	Recover fitted likelihood weights.
<code>logLik()</code>	Extract the unpenalized fitted log likelihood.
<code>AIC()</code> / <code>BIC()</code>	Compare appropriate maximum-likelihood fits using information criteria.
<code>vcov()</code>	Extract the fixed-effect covariance matrix when the Hessian is usable.
<code>nobs()</code> / <code>df.residual()</code> / <code>deviance()</code>	Extract fitted-row count, residual degrees of freedom, or absolute negative twice log likelihood.

QUICK ECOLOGICAL RECIPES

<code>drmTMB(drm_formula(trait ~ temp, sigma = ~ temp), family = gaussian(), data = dat)</code>	Ask whether both the expected trait and residual variability change with temperature.
<code>drmTMB(count ~ habitat, family = nbinom2(), data = dat)</code>	Model overdispersed abundance counts across habitats.
<code>drmTMB(cbind(success, failure) ~ treatment, family = beta_binomial(), data = dat)</code>	Model success probability with extra-binomial variation.
<code>check_drm(fit); predict(fit, newdata = grid)</code>	Check the fitted object before interpreting predictions on a controlled grid.
<code>profile_targets(fit); confint(fit, method = "profile", parm = target)</code>	Discover a direct target before requesting a profile-likelihood interval.

INTERVALS AND PROFILES

<code>profile_targets()</code>	Discover direct fitted targets and their profile names before requesting an interval.
<code>confint()</code>	Compute available Wald, profile-likelihood, or parametric-bootstrap intervals.
<code>profile()</code>	Compute likelihood-ratio profiles for explicit targets.
<code>plot()</code> for a profile	Plot the likelihood-ratio curve, cutoff, estimate, and interval endpoints.

PREDICT, SIMULATE, PLOT

<code>prediction_grid()</code>	Build a new-data grid that varies selected predictors while controlling the others.
<code>predict()</code>	Predict the response or a named distributional parameter for original or new data.
<code>predict_parameters()</code>	Return several fitted distributional parameters in one tidy long table.
<code>marginal_parameters()</code>	Compute unweighted plug-in averages of fitted parameters on their moment-appropriate reporting scales; this is not an interval or contrast.
<code>simulate()</code>	Simulate new responses from a fitted model.
<code>plot_cpairs()</code>	Plot a table returned by <code>corpairs()</code> .
<code>plot_parameter_surface()</code>	Plot a table returned by <code>predict_parameters()</code> or a compatible parameter-surface table.

INTERPRETATION GUARDRAILS

<code>student()</code> scale	Student-t sigma is a scale parameter, not the response standard deviation.
<code>drm_phylo_penalty()</code> fits	These are penalized/MAP fits; <code>logLik()</code> reports the unpenalized data likelihood, and ordinary LR/AIC comparisons are not standard.
Capability matrix	A visible function does not imply every family-by-structure interval route is validated.