

Package ‘gp3bayes’

August 23, 2026

Title Contract-First Bayesian Workflows for Hierarchical Behavioural Data

Version 0.5.0

Description Provides transparent, contract-first infrastructure for Bayesian analysis of repeated-measures and hierarchical behavioural data. It supports approved Bernoulli-logit, positive lognormal duration, and governed Gaussian dynamic-pupillometry workflows with strict readiness auditing, deterministic simulation, explicit preparation and transformation replay, inspectable scale-aware priors, prior and posterior predictive checks, restricted optional fitting through 'brms' with either 'rstan' or 'cmdstanr', sampling and temporal diagnostics, explicit posterior estimands, sensitivity analysis, target-specific predictive validation, simulation-based calibration, and conservative reporting. Core contracts and validation remain backend-independent. Version 0.5 adds governed robust and distributional dynamic pupillometry, bounded ARMA residual structures, Gaussian-process trajectories, explicit measurement uncertainty and missing-data models, joint binocular analysis, predictive model comparison, functional posterior estimands, and experimental nonlinear response-shape models while preserving explicit scientific and computational governance boundaries.

License MIT + file LICENSE

URL <https://stefanosbalaskas.github.io/gp3bayes/>,
<https://github.com/stefanosbalaskas/gp3bayes>

BugReports <https://github.com/stefanosbalaskas/gp3bayes/issues>

Encoding UTF-8

RoxygenNote 8.0.0

Imports withr, stats

Suggests ggplot2, SBC, bayesplot, brms, cmdstanr, detectseparation, knitr, loo, posterior, priorsense, rmarkdown, rstan, testthat
(>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

Additional_repositories <https://stan-dev.r-universe.dev>,
<https://r-multiverse.r-universe.dev>

NeedsCompilation no

Author Stefanos Balaskas [aut, cre, cph] (ORCID:
<https://orcid.org/0000-0003-2444-9796>)

Maintainer Stefanos Balaskas <s.balaskas@ac.upatras.gr>

Repository CRAN

Date/Publication 2026-08-23 10:40:44 UTC

Contents

advanced_pupil_trajectory_table	12
analysis_bundle_table	12
analysis_manifest_table	13
apply_transformation_recipe	13
as.data.frame.gp3bayes_pupil_contract	14
as.data.frame.gp3bayes_pupil_estimand	14
assess_binary_prior_sensitivity	15
assess_duration_prior_sensitivity	16
assess_powerscaled_sensitivity	17
as_pupil_prediction_draws	17
audit_advanced_pupil_identifiability	18
audit_backend_parity	19
audit_binocular_pupil_readiness	20
audit_design_support	20
audit_duration_boundaries	21
audit_duration_unit_invariance	22
audit_estimand_invariance	22
audit_fixed_effect_design	23
audit_missingness_structure	24
audit_model_readiness	25
audit_model_readiness_strict	26
audit_prediction_support	27
audit_pupil_computational_budget	28
audit_pupil_measurement_context	28
audit_pupil_measurement_model	29
audit_pupil_missingness	29
audit_pupil_predictive_calibration	30
audit_pupil_readiness	30
audit_pupil_temporal_dependence	31
audit_random_effects_support	32
backend_capabilities	32
backend_environment_table	33
backend_parity_table	33
bayesian_backend_capabilities	34
binary_calibration_error	34

binary_calibration_table	35
binary_calibration_uncertainty	35
binary_calibration_uncertainty_table	36
binary_confusion_table	37
binary_group_calibration	37
binary_precision_recall_curve	38
binary_prediction_scores	38
binary_roc_curve	39
binary_threshold_metrics	40
build_model_formula	40
capture_gp3bayes_schema	41
check_advanced_pupil_prior_predictive	42
check_binary_posterior_predictive	43
check_binary_ppc_details	44
check_binary_prior_predictive	44
check_cmdstan_backend	46
check_duration_posterior_predictive	47
check_duration_ppc_details	48
check_duration_prior_predictive	48
check_model_ppc	49
check_pupil_posterior_predictive	50
check_pupil_prior_predictive	51
collect_model_evidence	52
compare_analysis_manifests	53
compare_estimand_sensitivity	53
compare_gp3bayes_schemas	54
compare_psis_loo	54
compare_pupil_autocorrelation	55
compare_pupil_lfo	55
compare_pupil_models	56
compare_pupil_sensitivity_estimands	56
compute_kfold_cv	57
compute_loo_model_weights	58
compute_psis_loo	58
compute_psis_loo_from_log_lik	59
create_advanced_pupil_prior_specification	59
create_analysis_bundle	60
create_analysis_figure_set	61
create_analysis_manifest	61
create_binary_model_report	63
create_brms_sbc_plan	64
create_complete_evidence_inventory	65
create_contrast_coding_sensitivity_specification	65
create_custom_sbc_plan	66
create_diagnostic_dashboard	67
create_diagnostic_dashboard_figures	68
create_duration_model_report	68
create_duration_unit_sensitivity_specification	69

create_figure_set	70
create_group_deletion_sensitivity_plan	70
create_loo_influence_atlas	71
create_model_card	72
create_model_contract	72
create_model_evidence_report	74
create_model_specification	75
create_prediction_contrast_profile	75
create_prediction_grid	76
create_prediction_profile	77
create_prediction_surface	78
create_predictive_distribution_atlas	79
create_predictor_scaling_sensitivity_specification	80
create_prior_specification	81
create_publication_registry	82
create_publication_table_set	83
create_pupil_advanced_sensitivity_suite	83
create_pupil_arma_spec	84
create_pupil_contract	84
create_pupil_gp_spec	87
create_pupil_lfo_plan	88
create_pupil_measurement_model	88
create_pupil_missingness_spec	89
create_pupil_model_set	90
create_pupil_sensitivity_suite	90
create_pupil_validation_plan	92
create_random_slope_sensitivity_plan	93
create_reporting_checklist	93
create_sensitivity_suite_plan	94
create_transformation_recipe	95
design_support_table	95
detect_binary_separation	96
diagnose_advanced_pupil_fit	96
diagnose_binary_fit	97
diagnose_duration_fit	98
diagnose_model_fit	99
diagnose_pupil_fit	99
diagnostic_dashboard_table	100
duration_pit_table	100
duration_prediction_scores	101
duration_qq_table	101
duration_quantile_calibration	102
duration_tail_check	102
estimand_sensitivity_table	103
estimate_binocular_pupil_trajectory	103
estimate_model_estimands	104
estimate_pupil_auc	104
estimate_pupil_dynamic_contrast	105

estimate_pupil_peak	106
estimate_pupil_peak_latency	106
estimate_pupil_residual_scale	107
estimate_pupil_response_parameters	108
estimate_pupil_threshold_duration	108
estimate_pupil_trajectory	109
estimate_pupil_trajectory_derivative	110
estimate_pupil_window	110
estimate_standardized_duration_estimands	111
estimate_standardized_probability_contrast	112
evaluate_pathological_simulation	112
evidence_inventory_table	113
extract_expected_predictions	113
extract_linear_predictions	114
extract_log_likelihood	115
extract_posterior_draws	115
extract_posterior_predictions	116
extract_sampler_diagnostics	117
fit_advanced_pupil_model	117
fit_advanced_pupil_model_backend	118
fit_advanced_pupil_model_cmdstanr	118
fit_binary_model	119
fit_binary_model_backend	121
fit_binocular_pupil_model	122
fit_duration_model	123
fit_duration_model_backend	124
fit_pupil_model	125
fit_pupil_model_backend	126
fit_pupil_model_cmdstanr	127
fit_pupil_response_shape_model	128
freeze_analysis_manifest	128
freeze_gp3bayes_schema	129
gazeplot_pupil_mapping_table	129
gp3bayes_specification_traceability	130
grouped_prediction_check	130
group_deletion_sensitivity_table	131
group_effect_draws_table	131
group_effect_rank_probability_table	132
group_effect_table	133
group_prediction_summary	134
identify_identifier_like_predictors	134
identify_loo_influential_observations	135
identify_mcmc_issues	136
inspect_gazeplot_pupil_schema	137
interaction_prior_summary	138
invert_transformation_recipe	138
loo_diagnostic_table	139
loo_flagged_data	139

loo_group_influence_table	140
loo_influence_atlas_table	140
loo_influence_summary	141
loo_pointwise_table	141
loo_summary_table	142
manifest_comparison_table	142
materialize_pupil_advanced_sensitivity_scenario	143
materialize_pupil_sensitivity_scenario	143
mcmc_diagnostic_table	144
missingness_audit_table	144
model_card_table	145
model_comparison_table	145
model_evidence_table	146
model_weights_table	146
model_workflow_status	147
plot_gp3bayes_kfold_cv	147
plot_gp3bayes_strict_readiness_audit	148
plot_advanced_pupil_simulation	148
plot_advanced_pupil_trajectory	149
plot_autocorrelation	149
plot_backend_environment_gg	150
plot_backend_parity_gg	150
plot_binary_calibration	151
plot_binary_calibration_uncertainty	151
plot_binary_group_calibration	152
plot_binary_precision_recall	152
plot_binary_roc	153
plot_binary_threshold_metrics	153
plot_binocular_pupil_trajectory	154
plot_design_support_gg	154
plot_diagnostic_dashboard	155
plot_duration_pit	155
plot_duration_qq	156
plot_duration_quantile_calibration	156
plot_duration_tail	157
plot_estimand_intervals	157
plot_estimand_sensitivity_gg	158
plot_exceedance_probability	158
plot_grouped_prediction_check	159
plot_group_deletion_sensitivity	159
plot_group_effects	160
plot_group_effect_distribution	160
plot_group_effect_rank_probability	161
plot_group_predictions	161
plot_loo_group_elpd	162
plot_loo_group_influence	162
plot_loo_influence	163
plot_loo_influence_rank	163

plot_loo_pareto_vs_elpd	164
plot_loo_pointwise_elpd	164
plot_manifest_comparison_gg	165
plot_mcmc_quality	165
plot_missingness_gg	166
plot_model_comparison	166
plot_model_evidence_gg	167
plot_model_weights	167
plot_posterior_areas	168
plot_posterior_correlations	168
plot_posterior_density	169
plot_posterior_intervals	169
plot_posterior_pairs	170
plot_powerscale_sensitivity_gg	171
plot_ppc_statistic	171
plot_prediction_contrast_profile	172
plot_prediction_draws	172
plot_prediction_gradient	173
plot_prediction_intervals	173
plot_prediction_interval_width	174
plot_prediction_profile	174
plot_prediction_rank_probabilities	175
plot_prediction_score_uncertainty	175
plot_prediction_support	176
plot_prediction_surface	176
plot_prediction_surface_uncertainty	177
plot_predictive_atlas_statistics	177
plot_predictive_coverage	178
plot_predictive_quantile_envelope	178
plot_predictive_residuals	179
plot_prior_posterior_contraction	179
plot_prior_posterior_density	180
plot_prior_posterior_intervals	180
plot_prior_posterior_shift	181
plot_prior_sensitivity	181
plot_prior_sensitivity_scenarios	182
plot_pupil_autocorrelation_comparison	182
plot_pupil_dynamic_contrast	183
plot_pupil_estimand	183
plot_pupil_gp_hyperparameters	184
plot_pupil_identifiability_audit	184
plot_pupil_lfo	185
plot_pupil_measurement_audit	185
plot_pupil_measurement_uncertainty	186
plot_pupil_missingness	186
plot_pupil_model_comparison	187
plot_pupil_model_complexity	187
plot_pupil_observed_trajectory	188

plot_pupil_posterior_trajectory	188
plot_pupil_ppc	189
plot_pupil_predictive_calibration	189
plot_pupil_readiness	190
plot_pupil_residual_acf	190
plot_pupil_residual_scale	191
plot_pupil_residual_spectrum	191
plot_pupil_response_parameters	192
plot_pupil_sensitivity	192
plot_pupil_temporal_dependence	193
plot_pupil_trajectory_derivative	193
plot_pupil_validation	194
plot_random_intercept_variance_partition	194
plot_random_slope_sensitivity	195
plot_rank_diagnostics	195
plot_recovery_bias	196
plot_recovery_coverage	196
plot_recovery_estimates	197
plot_recovery_fit_status	197
plot_recovery_rmse	198
plot_reporting_checklist	198
plot_sampler_diagnostics	199
plot_sampling_diagnostics	199
plot_sbc_coverage_gg	200
plot_sbc_ecdf_gg	200
plot_sbc_rank_gg	201
plot_sbc_simulated_vs_estimated_gg	201
plot_schema_comparison_gg	202
plot_sensitivity_suite_gg	202
plot_uncertainty_decomposition	203
plot_variance_components	203
posterior_correlation_table	204
posterior_interval_table	204
posterior_predictive_statistic	205
posterior_predictive_summary_table	206
posterior_probability_table	206
powerscale_sensitivity_table	207
powerscale_sequence_for_fit	207
ppc_statistic_table	208
prediction_contrast	208
prediction_contrast_profile_table	209
prediction_draws_long	210
prediction_exceedance_probability	210
prediction_gradient_table	211
prediction_interval_width	211
prediction_pairwise_contrasts	212
prediction_profile_table	212
prediction_rank_probabilities	213

prediction_score_uncertainty	213
prediction_score_uncertainty_table	214
prediction_support_table	215
prediction_surface_table	215
prediction_table	216
prediction_uncertainty_decomposition	216
predictive_coverage_table	217
predictive_distribution_atlas_table	218
predictive_quantile_envelope	218
predictive_residuals	219
predict_advanced_pupil_trajectory	220
predict_binary_probability	221
predict_duration	221
predict_model	222
predict_pupil_trajectory	223
preflight_model_specification	224
prepare_binocular_pupil_timecourse	225
prepare_hierarchical_binary_data	226
prepare_hierarchical_duration_data	227
prepare_pupil_timecourse	228
prior_posterior_bridge	230
prior_posterior_distance_table	231
prior_posterior_draws_long	231
prior_posterior_summary_table	232
prior_sensitivity_scenario_table	232
prior_sensitivity_table	233
prior_specification_table	233
publication_registry_table	234
pupil_advanced_capabilities	234
pupil_advanced_compatibility_table	235
pupil_advanced_mapping_table	235
pupil_advanced_specification_table	236
pupil_autocorrelation_table	236
pupil_binocular_agreement_table	237
pupil_binocular_correlation	237
pupil_binocular_difference	238
pupil_condition_contrast	238
pupil_distribution_table	239
pupil_dynamic_contrast_table	239
pupil_gp_hyperparameters	240
pupil_gp_table	240
pupil_identifiability_table	241
pupil_measurement_audit_table	241
pupil_measurement_uncertainty_table	242
pupil_missingness_table	242
pupil_model_card	243
pupil_model_card_table	243
pupil_model_comparison_table	244

pupil_model_weights	244
pupil_ppc_table	245
pupil_readiness_table	245
pupil_residual_acf	246
pupil_residual_scale_table	246
pupil_residual_spectrum	247
pupil_response_parameter_table	247
pupil_sensitivity_table	248
pupil_specification_table	248
pupil_trajectory_derivative_table	249
pupil_trajectory_table	249
pupil_validation_table	250
random_intercept_variance_partition	250
random_intercept_variance_partition_table	251
random_slope_sensitivity_table	251
read_analysis_manifest	252
read_gp3bayes_schema	252
recovery_estimate_table	253
recovery_fit_status_table	253
recovery_parameter_table	254
register_publication_figure	254
register_publication_table	255
review_duration_extremes	255
run_binary_recovery	256
run_duration_recovery	257
run_group_deletion_sensitivity	259
run_random_slope_sensitivity	260
run_sbc_plan	261
run_sensitivity_suite	261
sampler_diagnostic_table	262
save_figure_set	263
save_publication_registry_figures	263
sbc_overview_table	264
sbc_stats_table	265
schema_comparison_table	265
score_pupil_predictions	266
sensitivity_suite_table	266
simulate_advanced_pupil_timecourse	267
simulate_binary_pathology	268
simulate_binocular_pupil_timecourse	269
simulate_declared_prior_draws	269
simulate_duration_pathology	270
simulate_hierarchical_binary_data	271
simulate_hierarchical_duration_data	272
simulate_pupil_response_shape	274
simulate_pupil_timecourse	275
specify_advanced_pupil_timecourse_model	277
specify_binary_model	279

specify_binary_model_with_interaction_prior	280
specify_binocular_pupil_model	281
specify_duration_model	282
specify_duration_model_with_interaction_prior	283
specify_pupil_distribution	284
specify_pupil_response_shape_model	285
specify_pupil_timecourse_model	286
summarise_binary_group_variation	287
summarise_binary_posterior	288
summarise_condition_balance	288
summarise_duration_posterior	289
summarise_estimand_draws	290
summarise_mcmc_quality	290
summarise_model_posterior	291
summarise_pupil_posterior	291
summarise_sbc_result	292
summarise_sensitivity_suite	292
theme_gp3bayes	293
translate_advanced_pupil_model_to_brms	293
translate_binary_model_to_brms	294
translate_binary_model_with_interaction_prior	295
translate_binocular_pupil_model_to_brms	295
translate_duration_model_to_brms	296
translate_duration_model_with_interaction_prior	296
translate_pupil_model_to_brms	297
translate_pupil_response_shape_to_brms	297
validate_analysis_manifest	298
validate_backend_environment	298
validate_gp3bayes_object	299
validate_gp3bayes_schema	300
validate_prior_specification	300
validate_publication_registry	301
validate_pupil_leave_future_out	301
validate_pupil_model	302
validate_transformation_replay	303
variance_component_table	303
write_analysis_bundle_report	304
write_diagnostic_dashboard_report	304
write_model_card	305
write_publication_registry	305
write_reproducibility_report	306

advanced_pupil_trajectory_table
<i>Summarise an advanced pupil trajectory</i>

Description

Summarise an advanced pupil trajectory

Usage

advanced_pupil_trajectory_table(prediction, probability = 0.95)

Arguments

- prediction An advanced trajectory prediction.
- probability Central posterior interval probability.

Value

A data frame.

analysis_bundle_table	<i>Analysis-Bundle Status Table</i>
-----------------------	-------------------------------------

Description

Analysis-Bundle Status Table

Usage

analysis_bundle_table(x)

Arguments

- x A gp3bayes_analysis_bundle.

Value

Component availability and captured errors.

`analysis_manifest_table`*Summarise Manifest Components*

Description

Summarise Manifest Components

Usage

```
analysis_manifest_table(manifest)
```

Arguments

`manifest` An analysis manifest.

Value

A data frame of key provenance components.

`apply_transformation_recipe`*Apply a Recorded Transformation Recipe*

Description

Apply a Recorded Transformation Recipe

Usage

```
apply_transformation_recipe(  
  new_data,  
  recipe,  
  input_scale = c("raw", "prepared"),  
  require_outcome = FALSE,  
  input_unit = NULL  
)
```

Arguments

`new_data` New data to transform.

`recipe` A `gp3bayes_transformation_recipe` or prepared `gp3bayes` object.

`input_scale` Either "raw" or "prepared".

`require_outcome` Whether the outcome column must be present.

`input_unit` Optional source duration unit used to guard duration replay.

Value

A transformed data frame with a recipe attribute.

```
as.data.frame.gp3bayes_pupil_contract
```

Convert a pupil contract to a transparent table

Description

Convert a pupil contract to a transparent table

Usage

```
## S3 method for class 'gp3bayes_pupil_contract'
as.data.frame(x, ...)
```

Arguments

x	A gp3bayes_pupil_contract.
...	Ignored.

Value

A data frame of contract fields.

```
as.data.frame.gp3bayes_pupil_estimand
```

Convert a pupil estimand to a data frame

Description

Convert a pupil estimand to a data frame

Usage

```
## S3 method for class 'gp3bayes_pupil_estimand'
as.data.frame(x, ...)
```

Arguments

x	A gp3bayes_pupil_estimand.
...	Ignored.

Value

The estimand table.

assess_binary_prior_sensitivity
Assess Binary Prior Sensitivity

Description

Refits the approved binary model under prespecified tighter and wider prior scales and compares population-level posterior medians.

Usage

```
assess_binary_prior_sensitivity(
  fit,
  scale_multipliers = c(tighter = 0.5, wider = 2),
  chains = fit$sampling$chains,
  iter = fit$sampling$iter,
  warmup = fit$sampling$warmup,
  cores = fit$sampling$cores,
  seed = fit$sampling$seed + 1000L,
  adapt_delta = fit$sampling$adapt_delta,
  max_treedepth = fit$sampling$max_treedepth,
  refresh = 0L,
  maximum_standardized_shift = 0.25,
  review_standardized_shift = 0.5,
  retain_fits = FALSE
)
```

Arguments

<code>fit</code>	A <code>gp3bayes_binary_fit</code> .
<code>scale_multipliers</code>	Named positive numeric multipliers applied to the intercept, coefficient, and group-scale priors.
<code>chains</code> , <code>iter</code> , <code>warmup</code> , <code>cores</code> , <code>seed</code> , <code>adapt_delta</code> , <code>max_treedepth</code> , <code>refresh</code>	Restricted sampling controls passed to <code>fit_binary_model()</code> .
<code>maximum_standardized_shift</code>	Maximum absolute posterior-median shift, divided by the reference posterior standard deviation, for a pass.
<code>review_standardized_shift</code>	Maximum standardized shift for review.
<code>retain_fits</code>	Whether alternative fitted objects should be retained.

Details

This function is computationally expensive. Sensitivity status describes stability under the declared scale changes only; it does not prove prior robustness under all defensible priors.

Value

A gp3bayes_binary_prior_sensitivity object.

```
assess_duration_prior_sensitivity
```

Assess Duration Prior Sensitivity

Description

Refits the approved duration model under prespecified tighter and wider prior scales and compares population-level and residual posterior medians.

Usage

```
assess_duration_prior_sensitivity(
  fit,
  scale_multipliers = c(tighter = 0.5, wider = 2),
  chains = fit$sampling$chains,
  iter = fit$sampling$iter,
  warmup = fit$sampling$warmup,
  cores = fit$sampling$cores,
  seed = fit$sampling$seed + 2000L,
  adapt_delta = fit$sampling$adapt_delta,
  max_treedepth = fit$sampling$max_treedepth,
  refresh = 0L,
  maximum_standardized_shift = 0.25,
  review_standardized_shift = 0.5,
  retain_fits = FALSE
)
```

Arguments

```
fit
```

A gp3bayes_duration_fit.

```
scale_multipliers
```

Named positive prior-scale multipliers.

```
chains, iter, warmup, cores, seed, adapt_delta, max_treedepth, refresh
```

Restricted sampling controls passed to [fit_duration_model\(\)](#).

```
maximum_standardized_shift
```

Maximum standardized median shift for pass.

```
review_standardized_shift
```

Maximum standardized median shift for review.

```
retain_fits
```

Whether alternative fits are retained.

Value

A gp3bayes_duration_prior_sensitivity.

assess_powerscaled_sensitivity

Assess Power-Scaled Prior and Likelihood Sensitivity

Description

Delegates to priorsense while retaining a conservative review status. Low local sensitivity does not prove that prior choices are irrelevant or that a model is robust.

Usage

```
assess_powerscaled_sensitivity(
  fit,
  variable = NULL,
  prior_selection = NULL,
  likelihood_selection = NULL
)
```

Arguments

fit	A gp3bayes fit or brmsfit.
variable	Optional posterior variables to inspect.
prior_selection	Optional tagged priors to perturb.
likelihood_selection	Optional likelihood subset.

Value

A gp3bayes_powerscale_sensitivity.

as_pupil_prediction_draws

Create a lightweight pupil prediction object from frozen draws

Description

Wraps already-computed posterior prediction draws for examples, reporting, and reproducible post-fit analysis without pretending that fitting occurred in the current session.

Usage

```
as_pupil_prediction_draws(
  draws,
  grid,
  unit,
  type = c("expected", "posterior_predictive", "linear"),
  max_cells = 5000000L
)
```

Arguments

<code>draws</code>	Numeric matrix with posterior draws in rows and grid points in columns.
<code>grid</code>	Data frame with one row per draw column. It should contain <code>.event_time</code> and may contain <code>.condition</code> , <code>.participant</code> , and <code>.item</code> .
<code>unit</code>	Declared pupil unit.
<code>type</code>	Prediction type label.
<code>max_cells</code>	Maximum draw-by-grid cells.

Value

A `gp3bayes_pupil_prediction`.

Examples

```
grid <- expand.grid(
  .event_time = seq(0, 1, length.out = 5),
  .condition = factor(c("control", "treatment"))
)
draws <- matrix(rnorm(1000), nrow = 100, ncol = nrow(grid))
prediction <- as_pupil_prediction_draws(
  draws, grid, unit = "millimetres"
)
```

`audit_advanced_pupil_identifiability`

Audit empirical support for an advanced pupil model specification

Description

This audit is deliberately heuristic. It identifies weakly supported design configurations before sampling but does not certify model identifiability or posterior adequacy.

Usage

```
audit_advanced_pupil_identifiability(specification)
```

Arguments

specification An advanced pupil specification.

Value

A gp3bayes_pupil_identifiability_audit object.

audit_backend_parity *Audit Posterior Parity Across rstan and cmdstanr*

Description

Compares posterior summaries from two independently sampled fits. Mean differences are evaluated relative to the combined Monte Carlo standard error rather than requiring identical draws. Standard-deviation differences are reported separately. A parity pass is a computational consistency check, not evidence that either model is statistically or substantively adequate.

Usage

```
audit_backend_parity(
  rstan_fit,
  cmdstanr_fit,
  variables = NULL,
  mcse_multiplier = 3,
  absolute_tolerance = 0,
  relative_sd_tolerance = 0.1
)
```

Arguments

rstan_fit	A gp3bayes rstan fit, or a compatible posterior-summary data frame for testing/auditing.
cmdstanr_fit	A gp3bayes cmdstanr fit, or a compatible summary table.
variables	Optional parameter names to compare. When omitted, the package's approved population/group-scale parameter set is used.
mcse_multiplier	Multiplier applied to the combined MCSE of posterior means to define a sampling-noise comparison band.
absolute_tolerance	Minimum absolute tolerance for mean differences.
relative_sd_tolerance	Review threshold for relative posterior-SD differences.

Value

A gp3bayes_backend_parity_audit.

Examples

```

rstan_summary <- data.frame(
  variable = c("b_Intercept", "b_conditiontreatment"),
  mean = c(-0.6, 0.4), sd = c(0.20, 0.15),
  mcse_mean = c(0.01, 0.01)
)
cmdstanr_summary <- data.frame(
  variable = c("b_Intercept", "b_conditiontreatment"),
  mean = c(-0.59, 0.41), sd = c(0.21, 0.15),
  mcse_mean = c(0.01, 0.01)
)
audit_backend_parity(rstan_summary, cmdstanr_summary)

```

```
audit_binocular_pupil_readiness
```

Audit binocular pupil availability and agreement descriptively

Description

Audit binocular pupil availability and agreement descriptively

Usage

```
audit_binocular_pupil_readiness(prepared)
```

Arguments

prepared A binocular prepared object.

Value

A gp3bayes_binocular_pupil_audit object.

```
audit_design_support    Audit Overall Design Support
```

Description

Combines missingness, fixed-effects design, random-effects support, standard readiness, and optional binary separation screening into one pre-fit audit.

Usage

```

audit_design_support(
  x,
  contract = NULL,
  separation = FALSE,
  strict_readiness = TRUE
)

```

Arguments

x	A data frame, prepared object, model specification, or fit.
contract	Required when x is a raw data frame.
separation	Whether to run <code>detect_binary_separation()</code> when the contract is binary and <code>detectseparation</code> is installed.
strict_readiness	Whether to include <code>audit_model_readiness_strict()</code> .

Value

A `gp3bayes_design_support_audit`.

audit_duration_boundaries

Audit Duration Range and Censoring Boundaries

Description

Checks an explicitly declared plausible measurement range and an explicitly supplied censoring indicator. Candidate censoring-like column names are only reported when no indicator is supplied; they are not interpreted silently.

Usage

```
audit_duration_boundaries(
  data,
  contract,
  allowed_range = NULL,
  censor_col = NULL,
  detect_candidate_columns = TRUE
)
```

Arguments

data	A data frame.
contract	An approved duration contract.
allowed_range	Optional positive lower and upper bounds in the contract's recorded outcome unit.
censor_col	Optional censoring-indicator column.
detect_candidate_columns	Whether common censoring/truncation names should be reported for review.

Value

A `gp3bayes_duration_boundary_audit` object.

 audit_duration_unit_invariance

Audit Duration-Unit Invariance

Description

Checks the unit-free median and predictive-quantile ratios and the expected scaling of absolute predictive quantities after a known unit conversion.

Usage

```
audit_duration_unit_invariance(
  reference,
  converted,
  multiplier,
  tolerance = 0.02
)
```

Arguments

reference, converted	Comparable duration estimands.
multiplier	Conversion factor applied to the outcome unit.
tolerance	Absolute tolerance for unit-free ratios and relative tolerance for scaled absolute quantities.

Value

A gp3bayes_duration_unit_invariance_audit.

 audit_estimand_invariance

Audit Estimand Invariance Against a Declared Tolerance

Description

Audit Estimand Invariance Against a Declared Tolerance

Usage

```
audit_estimand_invariance(
  reference,
  alternative,
  quantity = reference$primary_quantity,
  tolerance
)
```

Arguments

reference, alternative	Comparable gp3bayes estimands.
quantity	Quantity to compare.
tolerance	Maximum absolute median difference considered invariant for the declared scientific use.

Value

A gp3bayes_estimand_invariance_audit.

audit_fixed_effect_design

Audit the Fixed-Effects Design Matrix

Description

Checks rank, singular values, condition number, invariant columns and extreme leverage before fitting Stan. The audit never rewrites a formula or drops a predictor automatically.

Usage

```
audit_fixed_effect_design(
  x,
  contract = NULL,
  condition_number_review = 30,
  condition_number_fail = 100,
  leverage_multiplier = 3
)
```

Arguments

x	A data frame, prepared object, model specification, or fit.
contract	Required when x is a raw data frame.
condition_number_review	Condition number triggering review.
condition_number_fail	Condition number triggering fail.
leverage_multiplier	Review observations whose hat value is above $\text{leverage_multiplier} * p / n$.

Value

A gp3bayes_fixed_effect_design_audit.

audit_missingness_structure

Audit Missingness Structure Before Model Fitting

Description

Summarises missing values in declared analysis columns and, where possible, by participant, item, and condition. This is a reporting audit; no rows are dropped or imputed.

Usage

```
audit_missingness_structure(
  x,
  contract = NULL,
  review_fraction = 0.05,
  fail_fraction = 0.2
)
```

Arguments

x	A data frame, prepared object, model specification, or fit.
contract	Required when x is a raw data frame.
review_fraction	Column-level missing fraction above which a component is marked for review.
fail_fraction	Column-level missing fraction above which a component is marked fail. A fail does not automatically exclude data.

Value

A gp3bayes_missingness_audit.

Examples

```
data <- data.frame(
  participant_id = rep(c("p1", "p2"), each = 4),
  trial_id = rep(1:4, 2),
  condition = rep(c("control", "treatment"), 4),
  selected = c(0, 1, NA, 1, 1, 0, 1, 0)
)
contract <- create_model_contract(
  "binary", "selected", "participant_id",
  trial_col = "trial_id", condition_col = "condition"
)
audit_missingness_structure(data, contract)
```

audit_model_readiness *Audit Data Readiness for an Approved Model Contract*

Description

Audits whether a data frame satisfies the observable data requirements of an existing model contract created by `create_model_contract()`. The audit is backend-independent and does not construct a formula, define executable priors, fit a model, or establish model adequacy.

Usage

```
audit_model_readiness(data, contract)
```

Arguments

data	A data frame containing the declared outcome, grouping identifiers, predictors, and optional design columns.
contract	A gp3bayes_model_contract created by <code>create_model_contract()</code> .

Details

A readiness audit evaluates observable data properties only. Passing the audit does not establish convergence, model adequacy, predictive validity, causal identification, or substantive validity.

Failures block progression to a later model-building gate. Warnings identify weak or unusual structures that require review but do not automatically block progression.

Binary outcomes must be logical or numeric values encoded exclusively as zero and one, with both classes observed. Duration outcomes must be numeric, finite, strictly positive, uncensored, and variable.

Value

An object of class `gp3bayes_readiness_audit`. The object records:

- whether the data are ready to proceed to a later model-building gate;
- pass, warning, and failure counts;
- one structured row per readiness check;
- declared and observed column summaries; and
- the audited model contract.

The input data are not retained in the returned object.

Interpretation boundaries

Readiness checks cannot determine whether a model is scientifically justified. Behavioural measurements must not be interpreted as direct measures of latent psychological or protected attributes.

Examples

```

binary_data <- data.frame(
  participant_id = rep(c("p1", "p2"), each = 4),
  trial_id = rep(1:4, times = 2),
  condition = rep(c("control", "treatment"), times = 4),
  selected = c(0, 1, 0, 1, 1, 0, 1, 0)
)

binary_contract <- create_model_contract(
  family = "binary",
  outcome_col = "selected",
  participant_col = "participant_id",
  trial_col = "trial_id",
  condition_col = "condition"
)

audit_model_readiness(binary_data, binary_contract)

```

audit_model_readiness_strict

Run a Strict Model-Readiness Audit

Description

Extends `audit_model_readiness()` with explicit overall condition balance, binary within-group outcome variation, identifier-like predictor review, fixed-effects rank, duration extremes and boundaries, and optional binary separation screening.

Usage

```

audit_model_readiness_strict(
  data,
  contract,
  condition_warning_fraction = 0.1,
  condition_failure_fraction = 0.02,
  identifier_unique_fraction = 0.9,
  duration_allowed_range = NULL,
  censor_col = NULL,
  run_separation = TRUE
)

```

Arguments

<code>data</code>	A data frame.
<code>contract</code>	An approved model contract.
<code>condition_warning_fraction</code> , <code>condition_failure_fraction</code>	Condition balance thresholds.

identifier_unique_fraction	Identifier-like uniqueness threshold.
duration_allowed_range	Optional positive duration bounds.
censor_col	Optional duration censoring indicator.
run_separation	Whether to run the optional fixed-effects separation screen when the family is binary.

Value

A gp3bayes_strict_readiness_audit object.

audit_prediction_support
<i>Audit Prediction Support</i>

Description

Compares requested prediction rows with the observed model-building support. It reports extrapolation and novel levels but never removes prediction rows.

Usage

```
audit_prediction_support(fit, newdata)
```

Arguments

fit	A fitted gp3bayes_fit.
newdata	Data to audit.

Value

A gp3bayes_prediction_support object.

`audit_pupil_computational_budget`*Audit computational complexity before fitting an advanced pupil model*

Description

Audit computational complexity before fitting an advanced pupil model

Usage

```
audit_pupil_computational_budget(x)
```

Arguments

`x` An advanced specification.

Value

A `gp3bayes_pupil_complexity_audit` object.

`audit_pupil_measurement_context`*Audit pupil measurement and confound context*

Description

Summarises declared blink/interpolation, baseline, PFE/gaze, luminance, contrast, and sampling context without correcting or excluding observations.

Usage

```
audit_pupil_measurement_context(x)
```

Arguments

`x` A prepared pupil object.

Value

A `gp3bayes_pupil_measurement_audit`.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

`audit_pupil_measurement_model`*Audit a declared measurement model against data*

Description

Audit a declared measurement model against data

Usage

```
audit_pupil_measurement_model(specification)
```

Arguments

`specification` An advanced specification containing a measurement model.

Value

A `gp3bayes_pupil_measurement_audit_05` object.

`audit_pupil_missingness`*Audit missingness in an advanced pupil specification*

Description

Audit missingness in an advanced pupil specification

Usage

```
audit_pupil_missingness(specification)
```

Arguments

`specification` An advanced specification.

Value

A `gp3bayes_pupil_missingness_audit` object.

 audit_pupil_predictive_calibration

Audit posterior predictive calibration on explicit evaluation data

Description

Audit posterior predictive calibration on explicit evaluation data

Usage

```
audit_pupil_predictive_calibration(
  fit,
  newdata,
  ndraws = 500L,
  probability = 0.9,
  population_only = FALSE,
  allow_new_levels = FALSE
)
```

Arguments

fit	An advanced fitted model.
newdata	Evaluation data containing the pupil response.
ndraws	Number of posterior predictive draws.
probability	Interval probability.
population_only	Exclude group-level effects if TRUE.
allow_new_levels	Passed to brms prediction.

Value

A predictive-score object with evaluation metadata.

 audit_pupil_readiness *Audit pupil-timecourse readiness*

Description

Produces observable evidence about hierarchy, sampling, missingness, baseline support, measurement flags, gaze/PFE context, luminance, and preprocessing provenance. Review signals are not exclusion decisions.

Usage

```
audit_pupil_readiness(x, contract = NULL)
```

Arguments

x	A gp3bayes_pupil_prepared object or raw data with contract.
contract	Required only when x is raw data.

Value

A gp3bayes_pupil_readiness object with summary and stratified tables.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

```
audit_pupil_temporal_dependence
```

Audit empirical temporal dependence before model fitting

Description

Computes descriptive within-series autocorrelation and spacing diagnostics. The audit is diagnostic only and does not select an ARMA order.

Usage

```
audit_pupil_temporal_dependence(x, max_lag = 10L)
```

Arguments

x	A prepared pupil object, data frame, or advanced specification.
max_lag	Maximum lag for descriptive ACF summaries.

Value

A gp3bayes_pupil_temporal_dependence_audit object.

audit_random_effects_support
<i>Audit Random-Effects Support</i>

Description

Audits participant repetition, item crossing, and within-participant condition support for a requested random slope.

Usage

```
audit_random_effects_support(  
  x,  
  contract = NULL,  
  minimum_repeated_rows = 2L,  
  minimum_group_levels = 2L,  
  minimum_condition_cell_rows = 2L  
)
```

Arguments

- x A data frame, prepared object, model specification, or fit.
- contract Required when x is a raw data frame.
- minimum_repeated_rows Minimum observations per participant.
- minimum_group_levels Minimum participant/item levels.
- minimum_condition_cell_rows Minimum rows in each observed participant-condition cell when a random slope is requested.

Value

A gp3bayes_random_effects_support_audit.

backend_capabilities	<i>Report Bayesian Backend Capabilities</i>
----------------------	---

Description

Provides a stable 0.2.0-facing capability table for the two approved Stan backends. It augments the existing package capability report with package versions, CmdStan installation information, and explicit readiness fields. No model is compiled or fitted.

Usage

backend_capabilities()

Value

A gp3bayes_backend_capabilities_v2 data frame.

Examples

backend_capabilities()

backend_environment_table	<i>Backend-Environment Table</i>
---------------------------	----------------------------------

Description

Backend-Environment Table

Usage

backend_environment_table(x)

Arguments

x A gp3bayes_backend_environment.

Value

Backend environment checks.

backend_parity_table	<i>Backend-Parity Table</i>
----------------------	-----------------------------

Description

Backend-Parity Table

Usage

backend_parity_table(x)

Arguments

x A gp3bayes_backend_parity_audit.

Value

The parameter-level backend-parity table.

bayesian_backend_capabilities
<i>Report Optional Bayesian Backend Capabilities</i>

Description

Audits package availability and, where possible, runtime usability for the optional Bayesian extensions.

Usage

bayesian_backend_capabilities()

Value

A gp3bayes_backend_capabilities data frame.

binary_calibration_error
<i>Binary Calibration Error</i>

Description

Binary Calibration Error

Usage

binary_calibration_error(x, observed = NULL, bins = 10L)

Arguments

- | | |
|----------|--|
| x | A binary expected prediction or numeric probabilities. |
| observed | Optional observed outcomes. |
| bins | Number of equal-frequency bins. |

Value

A one-row table with expected and maximum absolute calibration error.

Examples

binary_calibration_error(c(0.1, 0.8, 0.7, 0.2), c(0, 1, 1, 0), bins = 2)

binary_calibration_table

Binary Calibration Table

Description

Binary Calibration Table

Usage

```
binary_calibration_table(x, bins = 10L, probs = c(0.025, 0.5, 0.975))
```

Arguments

x	A binary expected-response gp3bayes_prediction.
bins	Number of equal-frequency calibration bins.
probs	Posterior interval probabilities.

Value

A data frame comparing observed event rates with posterior mean event probabilities by bin.

binary_calibration_uncertainty

Binary Calibration Uncertainty

Description

Equal-width bins are defined from posterior-mean predicted probabilities.

Usage

```
binary_calibration_uncertainty(
  fit,
  newdata = NULL,
  bins = 10L,
  include_group_effects = FALSE,
  ndraws = 1000L,
  probs = c(0.025, 0.5, 0.975)
)
```

Arguments

fit	A fitted binary gp3bayes_fit.
newdata	Optional data containing observed outcomes.
bins	Number of equal-width probability bins.
include_group_effects	Whether fitted group effects are included.
ndraws	Expected-probability posterior draws.
probs	Three interval probabilities.

Value

A gp3bayes_binary_calibration_uncertainty.

binary_calibration_uncertainty_table

Binary Calibration-Uncertainty Table

Description

Binary Calibration-Uncertainty Table

Usage

```
binary_calibration_uncertainty_table(x)
```

Arguments

x	A binary calibration-uncertainty object.
---	--

Value

Bin-level summaries.

`binary_confusion_table`*Binary Confusion Table*

Description

Binary Confusion Table

Usage

```
binary_confusion_table(x, observed = NULL, threshold = 0.5)
```

Arguments

x	A binary expected prediction or numeric probabilities.
observed	Optional observed binary outcomes.
threshold	Classification threshold from 0 to 1.

Value

A four-row confusion table plus rates as attributes.

Examples

```
binary_confusion_table(c(0.1, 0.8, 0.7, 0.2), c(0, 1, 1, 0))
```

`binary_group_calibration`*Grouped Binary Calibration*

Description

Grouped Binary Calibration

Usage

```
binary_group_calibration(x, group)
```

Arguments

x	A binary expected-response gp3bayes_prediction with observed outcomes.
group	Name of a column in x\$newdata.

Value

A group-level calibration summary.

`binary_precision_recall_curve`*Binary Precision-Recall Curve*

Description

Binary Precision-Recall Curve

Usage

```
binary_precision_recall_curve(x, observed = NULL, thresholds = NULL)
```

Arguments

<code>x</code>	A binary expected prediction or numeric probabilities.
<code>observed</code>	Optional observed binary outcomes.
<code>thresholds</code>	Optional thresholds. By default all finite empirical breakpoints are used.

Value

A data frame containing recall and precision.

Examples

```
binary_precision_recall_curve(  
  c(0.1, 0.8, 0.7, 0.2),  
  c(0, 1, 1, 0)  
)
```

`binary_prediction_scores`*Binary Prediction Scores*

Description

Binary Prediction Scores

Usage

```
binary_prediction_scores(x, observed = NULL, threshold = 0.5, epsilon = 1e-12)
```

Arguments

x	A binary expected-response gp3bayes_prediction or numeric event probabilities.
observed	Optional binary outcomes when x is numeric.
threshold	Classification threshold used only for threshold summaries.
epsilon	Probability truncation used for finite log loss.

Value

A one-row data frame of descriptive predictive scores.

Examples

```
binary_prediction_scores(c(0.1, 0.8, 0.7, 0.2), c(0, 1, 1, 0))
```

binary_roc_curve	<i>Binary ROC Curve</i>
------------------	-------------------------

Description

Binary ROC Curve

Usage

```
binary_roc_curve(x, observed = NULL, thresholds = NULL)
```

Arguments

x	A binary expected prediction or numeric probabilities.
observed	Optional observed binary outcomes.
thresholds	Optional thresholds. By default all finite empirical breakpoints are used.

Value

A data frame containing false-positive and true-positive rates.

Examples

```
binary_roc_curve(c(0.1, 0.8, 0.7, 0.2), c(0, 1, 1, 0))
```

 binary_threshold_metrics

Binary Threshold-Metric Curve

Description

Binary Threshold-Metric Curve

Usage

```
binary_threshold_metrics(
  x,
  observed = NULL,
  thresholds = seq(0.1, 0.9, by = 0.05)
)
```

Arguments

x	A binary expected-response prediction or numeric probabilities.
observed	Optional observed binary outcomes.
thresholds	Numeric thresholds between 0 and 1, inclusive.

Value

A data frame with accuracy, sensitivity, specificity, and balanced accuracy over the supplied thresholds.

Examples

```
binary_threshold_metrics(
  c(0.1, 0.8, 0.7, 0.2),
  c(0, 1, 1, 0),
  thresholds = c(0.3, 0.5, 0.7)
)
```

 build_model_formula

Build an Approved Model Formula

Description

Constructs a backend-independent R formula from a `create_model_contract()` result. The formula records the approved fixed effects, one optional interaction, the participant grouping structure, an optional participant-level random slope, and an optional crossed item intercept.

Usage

```
build_model_formula(contract)
```

Arguments

contract A gp3bayes_model_contract created by `create_model_contract()`.

Details

The participant random intercept is always included. When `contract$random_slope` is TRUE, it is replaced by a correlated participant intercept-and-condition-slope term. A declared item identifier adds a crossed item random intercept.

The trial identifier is treated as a row key and is not added as a predictor or grouping factor. A declared time column is included as a linear population-level term only.

Value

An R formula. The formula is a specification only and has not been translated to, validated by, or fitted with a Bayesian backend.

Examples

```
contract <- create_model_contract(
  family = "binary",
  outcome_col = "selected",
  participant_col = "participant_id",
  item_col = "stimulus_id",
  condition_col = "condition",
  predictors = "age_z"
)

build_model_formula(contract)
```

```
capture_gp3bayes_schema
```

Capture the Structural Schema of a gp3bayes Object

Description

Captures classes, types, lengths, and field names without storing the object's values. The result is intended for release compatibility auditing, not for validating numerical or statistical equivalence.

Usage

```
capture_gp3bayes_schema(x, max_depth = 3L)
```

Arguments

`x` A gp3bayes object.

`max_depth` Maximum nested list depth to record.

Value

A gp3bayes_object_schema.

Examples

```
contract <- create_model_contract(
  family = "binary",
  outcome_col = "selected",
  participant_col = "participant_id",
  condition_col = "condition"
)
capture_gp3bayes_schema(contract)
```

check_advanced_pupil_prior_predictive

Check advanced pupil priors through prior-only simulation

Description

This optional backend gate samples only from priors using brms. It does not establish model adequacy.

Usage

```
check_advanced_pupil_prior_predictive(
  specification,
  backend = c("rstan", "cmdstanr"),
  chains = 2L,
  iter = 800L,
  warmup = 400L,
  cores = min(2L, chains),
  seed = 2026
)
```

Arguments

`specification` An advanced specification.

`backend` "rstan" or "cmdstanr".

`chains, iter, warmup, cores, seed` Sampling controls.

Value

A gp3bayes_pupil_advanced_prior_predictive object.

check_binary_posterior_predictive

Check Binary Posterior Predictive Behaviour

Description

Compares observed binary summaries with replicated outcomes from the fitted posterior predictive distribution.

Usage

```
check_binary_posterior_predictive(
  fit,
  draws = 500,
  seed = 1,
  pass_probability = 0.8,
  review_probability = 0.95
)
```

Arguments

fit	A gp3bayes_binary_fit.
draws	Number of posterior predictive replications.
seed	Non-negative integer seed used to select predictive draws.
pass_probability	Central predictive interval used for a pass.
review_probability	Wider central predictive interval used for review.

Details

The check evaluates prespecified descriptive summaries. It does not prove that the likelihood, link, random-effects structure, or substantive model is adequate.

Value

A gp3bayes_binary_posterior_predictive_check.

```
check_binary_ppc_details
```

Run Detailed Binary Posterior Predictive Checks

Description

Adds calibration bins, participant/item rate checks, focal-condition rates, sparse participant-condition cells, and all-zero/all-one participant patterns to the existing binary PPC workflow.

Usage

```
check_binary_ppc_details(
  fit,
  draws = 300L,
  seed = 1L,
  calibration_bins = 10L,
  sparse_cell_min = 3L
)
```

Arguments

fit	Approved binary fit.
draws	Number of posterior predictive draws.
seed	Random seed.
calibration_bins	Number of probability calibration bins.
sparse_cell_min	Cell size below which participant-condition cells are reported as sparse.

Value

A gp3bayes_binary_ppc_detail.

```
check_binary_prior_predictive
```

Check Binary Prior Predictive Behaviour

Description

Simulates replicated binary outcomes from the declared prior specification and prepared design without calling a Bayesian fitting backend.

Usage

```
check_binary_prior_predictive(
  specification,
  draws = 500,
  seed = 1,
  plausible_rate = c(0.01, 0.99),
  boundary_probability = c(0.01, 0.99),
  extreme_contrast = 0.8,
  maximum_degenerate_participant_fraction = 0.5,
  maximum_boundary_mass = 0.5,
  maximum_extreme_probability = 0.25
)
```

Arguments

<code>specification</code>	A <code>gp3bayes_binary_model_specification</code> .
<code>draws</code>	Number of prior predictive data sets.
<code>seed</code>	Non-negative integer random-number seed.
<code>plausible_rate</code>	Increasing lower and upper limits for plausible overall and condition-specific event rates.
<code>boundary_probability</code>	Probability thresholds used to identify prior mass close to zero and one.
<code>extreme_contrast</code>	Absolute probability-scale condition contrast considered extreme.
<code>maximum_degenerate_participant_fraction</code>	Maximum participant fraction allowed to have all-zero or all-one replicated outcomes in a draw.
<code>maximum_boundary_mass</code>	Maximum fraction of row probabilities allowed beyond the declared boundary thresholds in a draw.
<code>maximum_extreme_probability</code>	Maximum acceptable fraction of prior predictive draws violating each criterion.

Details

Failure does not select or alter priors automatically. It indicates that the declared priors and design generate outcomes that require substantive review. This check assesses prior implications, not posterior adequacy or model fit.

Value

A `gp3bayes_binary_prior_predictive_check` containing replicated summaries, structured checks, thresholds, and the seed.

Examples

```

simulation <- simulate_hierarchical_binary_data(
  n_participants = 12,
  trials_per_participant = 8,
  seed = 2026
)

contract <- create_model_contract(
  family = "binary",
  outcome_col = "selected",
  participant_col = "participant_id",
  item_col = "item_id",
  trial_col = "trial_id",
  condition_col = "condition"
)

prepared <- prepare_hierarchical_binary_data(
  simulation$data,
  contract,
  condition_levels = c("control", "treatment")
)

specification <- specify_binary_model(
  prepared,
  baseline = 0.35
)

check_binary_prior_predictive(
  specification,
  draws = 100,
  seed = 2027
)

```

check_cmdstan_backend *Check the CmdStanR Backend*

Description

Checks package availability, the C++ toolchain, and a configured CmdStan installation. No installation or repair is performed automatically.

Usage

```
check_cmdstan_backend(strict = FALSE)
```

Arguments

strict Whether to stop instead of returning a failed audit.

Value

A list with status, version, path, and diagnostic detail.

check_duration_posterior_predictive

Check Duration Posterior Predictive Behaviour

Description

Compares observed positive-duration summaries with replicated outcomes from the fitted posterior predictive distribution.

Usage

```
check_duration_posterior_predictive(
  fit,
  draws = 500L,
  seed = 1L,
  pass_probability = 0.8,
  review_probability = 0.95
)
```

Arguments

fit	A gp3bayes_duration_fit.
draws	Number of posterior predictive data sets.
seed	Non-negative integer seed.
pass_probability	Central predictive interval used for pass.
review_probability	Wider predictive interval used for review.

Details

The check covers median, mean, upper-tail, dispersion, condition-ratio, and grouping summaries. It does not prove global model adequacy.

Value

A gp3bayes_duration_posterior_predictive_check.

```
check_duration_ppc_details
```

Run Detailed Duration Posterior Predictive Checks

Description

Adds raw/log-scale distributions, median and upper-quantile summaries, tail exceedance, group medians, and within-participant focal-condition median ratios.

Usage

```
check_duration_ppc_details(
  fit,
  draws = 300L,
  seed = 1L,
  quantiles = c(0.5, 0.9, 0.95),
  tail_threshold = NULL
)
```

Arguments

<code>fit</code>	Approved duration fit.
<code>draws</code>	Number of posterior predictive draws.
<code>seed</code>	Random seed.
<code>quantiles</code>	Predictive quantiles to report.
<code>tail_threshold</code>	Optional substantive tail threshold in the analysis unit. If omitted, the observed 95th percentile is used descriptively.

Value

A `gp3bayes_duration_ppc_detail`.

```
check_duration_prior_predictive
```

Check Duration Prior Predictive Behaviour

Description

Simulates positive-duration data from the declared prior specification and prepared design without fitting a model.

Usage

```
check_duration_prior_predictive(
  specification,
  draws = 500L,
  seed = 1L,
  plausible_median = NULL,
  maximum_q99 = NULL,
  maximum_cv = 5,
  maximum_condition_ratio = 10,
  maximum_extreme_probability = 0.25
)
```

Arguments

specification	A gp3bayes_duration_model_specification.
draws	Number of prior predictive data sets.
seed	Non-negative integer seed.
plausible_median	Optional increasing pair for plausible overall medians in the prepared outcome unit.
maximum_q99	Maximum plausible 99th percentile.
maximum_cv	Maximum plausible coefficient of variation.
maximum_condition_ratio	Maximum plausible ratio between condition medians in either direction.
maximum_extreme_probability	Maximum fraction of prior predictive draws allowed to violate each criterion.

Details

Failure requests substantive prior review; it does not select or alter priors automatically.

Value

A gp3bayes_duration_prior_predictive_check.

check_model_ppc

Check Posterior Predictive Behaviour

Description

Family-neutral wrapper around the approved family-specific posterior predictive checks. Passing this check is not a global adequacy claim.

Usage

```
check_model_ppc(fit, ...)
```

Arguments

<code>fit</code>	A <code>gp3bayes_fit</code> .
<code>...</code>	Family-specific diagnostic arguments.

Value

A family-specific `gp3bayes_posterior_predictive_check`.

`check_pupil_posterior_predictive`

Run pupil-specific posterior predictive checks

Description

Compares observed and replicated trajectories, distributional features, whole-support peak/latency and AUC, optional declared-window response, lag-one serial structure, participant/trial heterogeneity, residual trajectories, and blink/interpolation context. The object reports evidence and never declares a model adequate.

Usage

```
check_pupil_posterior_predictive(
  fit,
  ndraws = 200L,
  probability = 0.9,
  window = NULL,
  max_cells = 3000000L
)
```

Arguments

<code>fit</code>	A fitted pupil model.
<code>ndraws</code>	Posterior predictive draws.
<code>probability</code>	Predictive envelope probability.
<code>window</code>	Optional user-declared event-time window in canonical seconds.
<code>max_cells</code>	Maximum draw-by-observation cells.

Value

A `gp3bayes_pupil_ppc`.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

 check_pupil_prior_predictive

Check the approved pupil priors predictively

Description

Creates a governed prior-predictive plan by default. With `execute = TRUE`, draws from the prior-only approved Gaussian brms model and compares replicated pupil values with the observed model-scale range. The check reports evidence only and never changes priors automatically.

Usage

```
check_pupil_prior_predictive(
  specification,
  execute = FALSE,
  backend = c("rstan", "cmdstanr"),
  draws = 200L,
  chains = 2L,
  iter = 1000L,
  warmup = 500L,
  cores = min(2L, chains),
  seed = 2026,
  probability = 0.95,
  max_cells = 3000000L
)
```

Arguments

<code>specification</code>	Approved pupil model specification.
<code>execute</code>	Whether to run prior-only MCMC. Defaults to <code>FALSE</code> .
<code>backend</code>	Approved backend, <code>"rstan"</code> or <code>"cmdstanr"</code> .
<code>draws</code>	Number of prior predictive replicated draws to retain.
<code>chains, iter, warmup, cores, seed</code>	Sampling controls. Package-controlled cores are capped at two.
<code>probability</code>	Central predictive interval probability.
<code>max_cells</code>	Maximum retained draw-by-observation cells.

Value

A `gp3bayes_pupil_prior_predictive` evidence object.

Governance boundary

This operation does not tune priors, select a favourable prior scale, or certify a model as scientifically adequate. `execute = FALSE` performs no compilation or fitting.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

```
collect_model_evidence
```

Collect Model Evidence Without Declaring Model Adequacy

Description

Collects already-computed design, diagnostics, posterior summaries, posterior predictive checks, estimands, predictive validation, sensitivity, and reproducibility provenance into a single review object.

Usage

```
collect_model_evidence(
  fit = NULL,
  design = NULL,
  diagnostics = NULL,
  posterior = NULL,
  ppc = NULL,
  estimands = NULL,
  loo = NULL,
  kfold = NULL,
  sensitivity = NULL,
  manifest = NULL,
  compute = character()
)
```

Arguments

<code>fit</code>	Optional gp3bayes fit.
<code>design</code> , <code>diagnostics</code> , <code>posterior</code> , <code>ppc</code> , <code>estimands</code> , <code>loo</code> , <code>kfold</code> , <code>sensitivity</code>	Optional evidence components.
<code>manifest</code>	Optional gp3bayes_analysis_manifest containing reproducibility provenance for the analysis.
<code>compute</code>	Character vector selecting inexpensive components to compute from fit when not supplied. Supported values are "diagnostics", "posterior", and "estimands". Posterior predictive checks and refitting sensitivity are intentionally not automatic.

Value

A gp3bayes_model_evidence.

`compare_analysis_manifests`*Compare Analysis Manifests*

Description

Compares analysis-defining fields without interpreting any difference as automatically problematic.

Usage

```
compare_analysis_manifests(x, y)
```

Arguments

`x, y` Analysis manifests.

Value

A `gp3bayes_manifest_comparison`.

`compare_estimand_sensitivity`*Compare Estimand Sensitivity Across Alternative Fits*

Description

Compare Estimand Sensitivity Across Alternative Fits

Usage

```
compare_estimand_sensitivity(  
  reference,  
  alternatives,  
  quantity = reference$primary_quantity  
)
```

Arguments

`reference` Reference `gp3bayes_estimand`.
`alternatives` Named list of alternative estimands.
`quantity` Quantity to compare; defaults to the reference primary quantity.

Value

A `gp3bayes_estimand_sensitivity`.

compare_gp3bayes_schemas

Compare gp3bayes Object Schemas

Description

Compare gp3bayes Object Schemas

Usage

```
compare_gp3bayes_schemas(x, y, compare_lengths = FALSE)
```

Arguments

x	A gp3bayes object or captured schema.
y	A gp3bayes object or captured schema.
compare_lengths	Whether vector/list lengths are part of the structural compatibility rule. The default is FALSE because analysis-specific cardinalities such as numbers of predictors or groups may legitimately differ while the serialized object contract remains compatible.

Value

A gp3bayes_schema_comparison.

compare_psis_loo

Compare Models with PSIS-LOO

Description

Compare Models with PSIS-LOO

Usage

```
compare_psis_loo(models, moment_match = FALSE, relloo = FALSE, cores = 1L)
```

Arguments

models	Named list of gp3bayes fits, brmsfit objects, or gp3bayes_psis_loo objects.
moment_match	Whether to request moment matching.
relloo	Whether to request exact refits for problematic observations.
cores	Number of cores.

Value

A gp3bayes_loo_comparison. The result never selects a model.

`compare_pupil_autocorrelation`*Compare residual autocorrelation across fitted advanced models*

Description

This function does not choose a winner. It summarises lag-specific residual dependence after subtracting posterior expected means.

Usage

```
compare_pupil_autocorrelation(..., max_lag = 10L, ndraws = 300L)
```

Arguments

<code>...</code>	Two or more named advanced fits, or one named list.
<code>max_lag</code>	Maximum residual ACF lag.
<code>ndraws</code>	Draws used for posterior expected means.

Value

A `gp3bayes_pupil_autocorrelation_comparison` object.

`compare_pupil_lfo`*Compare executed leave-future-out validations*

Description

Compare executed leave-future-out validations

Usage

```
compare_pupil_lfo(...)
```

Arguments

<code>...</code>	Two or more named executed LFO validation objects, or one named list.
------------------	---

Value

A `gp3bayes_pupil_lfo_comparison` object.

compare_pupil_models *Compare fitted pupil models predictively*

Description

Compare fitted pupil models predictively

Usage

```
compare_pupil_models(
  model_set,
  criterion = c("loo", "kfold"),
  K = 10L,
  group = NULL,
  moment_match = FALSE,
  save_psis = TRUE
)
```

Arguments

model_set	A named model set.
criterion	"loo" or exact "kfold". Leave-future-out is handled by explicit plans via create_pupil_lfo_plan() and validate_pupil_leave_future_out() .
K	Number of folds for exact K-fold CV.
group	Optional grouping column passed to brms K-fold.
moment_match	Use brms/loo moment matching for PSIS-LOO where supported.
save_psis	Save PSIS objects.

Value

A gp3bayes_pupil_model_comparison object.

compare_pupil_sensitivity_estimands
 Compare declared pupil sensitivity estimands

Description

Combines already-computed pupil estimands by named scenario. No scenario is ranked or selected.

Usage

```
compare_pupil_sensitivity_estimands(results)
```


Arguments

results Named list of gp3bayes_pupil_estimand objects.

Value

A gp3bayes_pupil_sensitivity_comparison.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

compute_kfold_cv	<i>Compute Governed Exact K-Fold Cross-Validation</i>
------------------	---

Description

Uses `brms::kfold()` as an explicit fallback or complement to PSIS-LOO. Grouped folds may use only the declared participant or item grouping column. No model is selected automatically.

Usage

```
compute_kfold_cv(
  fit,
  K = 10L,
  folds = c("random", "stratified", "grouped"),
  group = NULL,
  joint = c("obs", "fold", "group"),
  save_fits = FALSE,
  seed = 1L
)
```

Arguments

fit Approved gp3bayes fit.

K Number of folds for random or stratified splitting.

folds One of "random", "stratified", or "grouped".

group Optional declared grouping column used by stratified/grouped splitting.

joint One of "obs", "fold", or "group".

save_fits Whether cross-validation refits are retained.

seed Random seed.

Value

A gp3bayes_kfold_cv.

 compute_loo_model_weights

Compute LOO Model-Averaging Weights

Description

Compute LOO Model-Averaging Weights

Usage

```
compute_loo_model_weights(x, method = c("stacking", "pseudobma"), cores = 1L)
```

Arguments

x	A gp3bayes_loo_comparison or named list of PSIS-LOO objects.
method	Either stacking or pseudo-BMA.
cores	Number of cores.

Value

A gp3bayes_loo_weights object.

compute_psis_loo

Compute PSIS-LOO for a gp3bayes Fit

Description

Compute PSIS-LOO for a gp3bayes Fit

Usage

```
compute_psis_loo(
  fit,
  moment_match = FALSE,
  relloo = FALSE,
  cores = 1L,
  save_psis = TRUE
)
```

Arguments

fit	A gp3bayes fit or brmsfit.
moment_match	Whether to request moment matching.
relloo	Whether to request exact refits for problematic observations.
cores	Number of cores.
save_psis	Whether to retain the PSIS object.

Value

A conservative gp3bayes_psis_loo result.

```
compute_psis_loo_from_log_lik
```

Compute PSIS-LOO from a Log-Likelihood Matrix

Description

This function supports deterministic tests and advanced workflows that already possess pointwise log-likelihood draws.

Usage

```
compute_psis_loo_from_log_lik(
  log_lik,
  chain_id = NULL,
  cores = 1L,
  save_psis = TRUE
)
```

Arguments

log_lik	Matrix with posterior draws in rows and observations in columns.
chain_id	Optional chain identifier for each row.
cores	Number of cores.
save_psis	Whether to retain the PSIS object.

Value

A gp3bayes_psis_loo.

```
create_advanced_pupil_prior_specification
```

Create an advanced pupil prior specification

Description

Constructs the governed default prior specification used when translating an advanced pupil time-course model to brms. This function defines prior scales only; it does not compile or fit a model and does not establish model adequacy.

Usage

```
create_advanced_pupil_prior_specification(specification)
```

Arguments

specification A gp3bayes_pupil_advanced_specification created by [specify_advanced_pupil_timecourse_mo](#)

Value

A gp3bayes_pupil_advanced_prior_specification object.

create_analysis_bundle

Create a Structured Post-Fit Analysis Bundle

Description

Collects reusable posterior, diagnostic, prediction, calibration, scoring, and optionally PSIS-LOO tables without making an automatic adequacy or model selection decision.

Usage

```
create_analysis_bundle(
  fit,
  newdata = NULL,
  ndraws = 1000L,
  include_group_effects = FALSE,
  include_loo = FALSE
)
```

Arguments

fit	A fitted gp3bayes_fit.
newdata	Optional prediction data.
ndraws	Posterior draws used for prediction-facing components.
include_group_effects	Whether prediction summaries include recorded group-level effects.
include_loo	Whether PSIS-LOO is computed.

Value

A gp3bayes_analysis_bundle.

`create_analysis_figure_set`*Create Publication Figures from an Analysis Bundle*

Description

Produces only figures supported by available bundle components.

Usage

```
create_analysis_figure_set(x)
```

Arguments

`x` A gp3bayes_analysis_bundle.

Value

A gp3bayes_figure_set.

`create_analysis_manifest`*Create an Analysis Manifest*

Description

Records the declared analysis contract, transformations, estimands, sensitivity plan, random seed, data fingerprint, package versions, and optional sampling specification in one backend-independent provenance object. The manifest stores a data fingerprint rather than a duplicate copy of the analysis data.

Usage

```
create_analysis_manifest(  
  specification = NULL,  
  fit = NULL,  
  data = NULL,  
  estimands = character(),  
  sensitivity_plan = NULL,  
  seed = NULL,  
  label = NULL,  
  notes = character()  
)
```

Arguments

specification	Optional approved gp3bayes model specification.
fit	Optional gp3bayes fit. When supplied, the specification and prepared data are derived from the fit unless explicitly supplied.
data	Optional analysis data frame. When omitted it is derived from the specification or fit where possible.
estimands	Character vector or structured list describing the prespecified estimands.
sensitivity_plan	Optional sensitivity-plan object or list.
seed	Optional non-negative integer seed. When omitted and fit is supplied, the recorded fitting seed is used.
label	Optional human-readable analysis label.
notes	Optional character notes.

Value

A gp3bayes_analysis_manifest.

Examples

```
simulation <- simulate_hierarchical_binary_data(
  n_participants = 8,
  trials_per_participant = 6,
  n_items = 4,
  random_slope_sd = 0,
  seed = 2026
)
contract <- create_model_contract(
  family = "binary",
  outcome_col = "selected",
  participant_col = "participant_id",
  item_col = "item_id",
  trial_col = "trial_id",
  condition_col = "condition"
)
prepared <- prepare_hierarchical_binary_data(
  simulation$data,
  contract,
  condition_levels = c("control", "treatment")
)
specification <- specify_binary_model(prepared, baseline = 0.35)
manifest <- create_analysis_manifest(
  specification = specification,
  estimands = "standardized_probability_contrast",
  seed = 2026
)
manifest
```

`create_binary_model_report`*Create a Structured Binary Model Report*

Description

Writes a conservative Markdown report for an approved fitted binary model.

Usage

```
create_binary_model_report(  
  fit,  
  diagnostics = NULL,  
  posterior_summary = NULL,  
  posterior_predictive = NULL,  
  prior_sensitivity = NULL,  
  recovery = NULL,  
  file,  
  overwrite = FALSE  
)
```

Arguments

<code>fit</code>	A <code>gp3bayes_binary_fit</code> .
<code>diagnostics</code>	Optional result from diagnose_binary_fit() .
<code>posterior_summary</code>	Optional result from summarise_binary_posterior() .
<code>posterior_predictive</code>	Optional result from check_binary_posterior_predictive() .
<code>prior_sensitivity</code>	Optional result from assess_binary_prior_sensitivity() .
<code>recovery</code>	Optional result from run_binary_recovery() .
<code>file</code>	Explicit output Markdown path. The caller must supply the destination; the function has no default output path.
<code>overwrite</code>	Whether an existing file may be replaced.

Details

The report never converts diagnostic or predictive statuses into an automatic statement that the model converged or is substantively valid.

Value

A `gp3bayes_binary_model_report` containing the normalized path and section-status registry.

create_brms_sbc_plan *Create a brms-Based Simulation-Based Calibration Plan*

Description

Constructs a complete SBC generator/backend pair using the restricted gp3bayes brms translation. Because the generator and backend share brms implementation code, this plan is mainly a computational-calibration check; custom independent generators remain preferable for detecting shared implementation errors.

Usage

```
create_brms_sbc_plan(
  specification,
  n_sims = 20L,
  backend = c("rstan", "cmdstanr"),
  chains = 2L,
  iter = 1000L,
  warmup = 500L,
  thin = 1L,
  seed = 1L,
  generator_iter = 3000L,
  generator_warmup = 2000L
)
```

Arguments

specification	An approved binary or duration specification.
n_sims	Number of simulated datasets.
backend	Either rstan or cmdstanr.
chains	Number of chains used per SBC fit.
iter	Total iterations per SBC fit.
warmup	Warmup iterations.
thin	Thinning interval.
seed	Seed used when datasets are generated.
generator_iter	Total prior-only generator iterations.
generator_warmup	Prior-only generator warmup.

Value

A gp3bayes_sbc_plan.

```
create_complete_evidence_inventory
```

Create a Complete Evidence Inventory

Description

Create a Complete Evidence Inventory

Usage

```
create_complete_evidence_inventory(..., label = NULL)
```

Arguments

```
...      Named evidence objects.
label    Optional label.
```

Value

A gp3bayes_evidence_inventory.

```
create_contrast_coding_sensitivity_specification
```

Create a Contrast-Coding Sensitivity Specification

Description

Replays the prepared data back to the recorded raw scale, applies an alternative two-level condition coding, and rebuilds the approved specification. Because the intercept meaning changes with coding, an explicit new baseline is required.

Usage

```
create_contrast_coding_sensitivity_specification(
  specification,
  condition_coding,
  baseline
)
```

Arguments

```
specification  An approved model specification.
condition_coding Two distinct numeric condition codes.
baseline       Explicit baseline probability or median under the new coding.
```

Value

An approved alternative specification.

```
create_custom_sbc_plan
```

Create an Expert Custom SBC Plan

Description

Wraps an independently coded generator function and a user-supplied SBC backend. The generator must return variables and generated elements as required by SBC.

Usage

```
create_custom_sbc_plan(
  generator_function,
  backend,
  n_sims = 20L,
  generator_args = list(),
  seed = 1L
)
```

Arguments

generator_function	Function producing one SBC dataset.
backend	A valid SBC backend object.
n_sims	Number of datasets.
generator_args	Named list passed to the generator constructor.
seed	Seed used for dataset generation.

Value

A gp3bayes_sbc_plan.

`create_diagnostic_dashboard`*Create a Diagnostic Dashboard Object*

Description

Expensive analyses are never launched implicitly. Supply already-computed evidence objects.

Usage

```
create_diagnostic_dashboard(  
  fit = NULL,  
  analysis_bundle = NULL,  
  model_card = NULL,  
  loo = NULL,  
  prior_posterior = NULL,  
  sensitivity = NULL,  
  recovery = NULL,  
  sbc = NULL,  
  label = NULL  
)
```

Arguments

<code>fit</code>	Optional fitted model.
<code>analysis_bundle</code>	Optional analysis bundle.
<code>model_card</code>	Optional model card.
<code>loo</code>	Optional PSIS-LOO or LOO influence atlas.
<code>prior_posterior</code>	Optional prior-posterior bridge.
<code>sensitivity</code>	Optional sensitivity result.
<code>recovery</code>	Optional recovery result.
<code>sbc</code>	Optional SBC result.
<code>label</code>	Optional label.

Value

A `gp3bayes_diagnostic_dashboard`.

`create_diagnostic_dashboard_figures`*Create Diagnostic Dashboard Figures*

Description

Create Diagnostic Dashboard Figures

Usage

```
create_diagnostic_dashboard_figures(x)
```

Arguments

`x` A diagnostic dashboard.

Value

A gp3bayes_figure_set for supported evidence components.

`create_duration_model_report`*Create a Structured Duration Model Report*

Description

Writes a conservative Markdown report for an approved fitted lognormal duration model.

Usage

```
create_duration_model_report(  
  fit,  
  diagnostics = NULL,  
  posterior_summary = NULL,  
  posterior_predictive = NULL,  
  prior_sensitivity = NULL,  
  recovery = NULL,  
  file,  
  overwrite = FALSE  
)
```

Arguments

fit	A gp3bayes_duration_fit.
diagnostics	Optional result from diagnose_duration_fit() .
posterior_summary	Optional result from summarise_duration_posterior() .
posterior_predictive	Optional result from check_duration_posterior_predictive() .
prior_sensitivity	Optional result from assess_duration_prior_sensitivity() .
recovery	Optional result from run_duration_recovery() .
file	Explicit output Markdown path. The caller must supply the destination; the function has no default output path.
overwrite	Whether an existing file may be replaced.

Value

A gp3bayes_duration_model_report.

```
create_duration_unit_sensitivity_specification
```

Create a Duration-Unit Sensitivity Specification

Description

Re-expresses an approved prepared duration outcome in a new unit by a positive multiplicative conversion, shifts the baseline median accordingly, and retains all dimensionless prior scales.

Usage

```
create_duration_unit_sensitivity_specification(
  specification,
  multiplier,
  new_unit
)
```

Arguments

specification	An approved duration specification.
multiplier	Positive conversion factor from the current analysis unit to the new unit.
new_unit	New non-empty unit label.

Value

An approved duration sensitivity specification.

create_figure_set	<i>Create a Named Figure Set</i>
-------------------	----------------------------------

Description

Create a Named Figure Set

Usage

```
create_figure_set(..., title = "gp3bayes figure set")
```

Arguments

...	Named plot objects.
title	Optional figure-set title.

Value

A gp3bayes_figure_set.

Examples

```
if (requireNamespace("ggplot2", quietly = TRUE)) {
  p <- ggplot2::ggplot(data.frame(x = 1:3, y = 1:3), ggplot2::aes(x, y)) +
    ggplot2::geom_point()
  create_figure_set(example = p)
}
```

create_group_deletion_sensitivity_plan	<i>Create a Group-Deletion Sensitivity Plan</i>
--	---

Description

Create a Group-Deletion Sensitivity Plan

Usage

```
create_group_deletion_sensitivity_plan(
  specification,
  group = c("participant", "item"),
  units = NULL,
  max_units = 20L
)
```

Arguments

specification	An approved binary or duration specification.
group	Either participant or item.
units	Optional explicit group levels. If omitted all levels are used only when their count does not exceed max_units.
max_units	Maximum automatic number of omission fits.

Value

A gp3bayes_group_deletion_sensitivity_plan.

create_loo_influence_atlas

Create a LOO Influence Atlas

Description

Create a LOO Influence Atlas

Usage

```
create_loo_influence_atlas(x, data = NULL, threshold = 0.7)
```

Arguments

x	A gp3bayes PSIS-LOO or raw loo object.
data	Optional observation-level data.
threshold	Pareto-k threshold used for the flagged table.

Value

A gp3bayes_loo_influence_atlas.

create_model_card	<i>Create a gp3bayes Model Card</i>
-------------------	-------------------------------------

Description

Creates a compact, structured record of model identity, computational diagnostics, prediction evidence, provenance, and interpretation boundaries. The card is documentation; it does not issue a model-adequacy certificate.

Usage

```
create_model_card(fit, analysis_bundle = NULL, manifest = NULL, label = NULL)
```

Arguments

fit	A fitted gp3bayes_fit.
analysis_bundle	Optional gp3bayes_analysis_bundle.
manifest	Optional gp3bayes_analysis_manifest.
label	Optional human-readable label.

Value

A gp3bayes_model_card.

create_model_contract	<i>Create an Approved Bayesian Model Contract</i>
-----------------------	---

Description

Creates an inspectable model-contract object for one of the two model families approved for the initial gp3bayes development scope. The function records neutral data-column mappings while preserving the approved likelihood, link, estimands, assumptions, diagnostics, sensitivity requirements, and interpretation boundaries.

Usage

```
create_model_contract(
  family,
  outcome_col,
  participant_col,
  item_col = NULL,
  trial_col = NULL,
  condition_col = NULL,
  time_col = NULL,
```



```

    predictors = character(),
    interaction = NULL,
    random_slope = FALSE,
    outcome_unit = NULL,
    notes = character()
  )

```

Arguments

family	Character scalar identifying the approved model family. Supported values are "binary" and "duration".
outcome_col	Character scalar naming the outcome column.
participant_col	Character scalar naming the participant identifier column.
item_col	Optional character scalar naming an item or stimulus identifier column.
trial_col	Optional character scalar naming a trial identifier column.
condition_col	Optional character scalar naming the focal condition column.
time_col	Optional character scalar naming a linear time or trial order column. This does not define a time-course or autocorrelation model.
predictors	Character vector naming additional predictors.
interaction	Optional character vector of length two naming one prespecified two-way interaction. Higher-order or multiple interactions are not supported by the initial contract.
random_slope	Logical scalar indicating whether one participant-level random slope for the focal condition is requested. Readiness must be assessed separately before fitting.
outcome_unit	Optional character scalar recording the outcome unit. It is required for the duration family and must be NULL for the binary family.
notes	Optional character vector containing user-supplied design or analysis notes. Notes do not override the approved model contract.

Details

The returned object is a specification and audit record. It does not validate a data frame, construct a backend formula, fit a model, or imply that the proposed analysis is appropriate. Those gates are handled by separate workflows.

The binary contract uses a Bernoulli likelihood with a logit link. The duration contract uses a lognormal likelihood for strictly positive, finite, uncensored durations.

Value

An object of class `gp3bayes_model_contract`. It is a named list containing the approved methodological specification, neutral column mappings, requested model structure, assumptions, diagnostics, sensitivity requirements, limitations, and unsupported uses.

Interpretation boundaries

Contract creation does not establish causal identification, model adequacy, convergence, predictive validity, or substantive validity. Behavioural measurements must not be interpreted as direct measures of latent psychological or protected attributes.

Examples

```
binary_contract <- create_model_contract(
  family = "binary",
  outcome_col = "selected",
  participant_col = "participant_id",
  item_col = "stimulus_id",
  trial_col = "trial_id",
  condition_col = "condition"
)

binary_contract

duration_contract <- create_model_contract(
  family = "duration",
  outcome_col = "response_time",
  participant_col = "participant_id",
  trial_col = "trial_id",
  condition_col = "condition",
  outcome_unit = "milliseconds"
)

duration_contract
```

```
create_model_evidence_report
      Create a Model Evidence Report
```

Description

Writes an explicit Markdown inventory of available evidence components.

Usage

```
create_model_evidence_report(evidence, file, overwrite = FALSE)
```

Arguments

evidence	A gp3bayes_model_evidence object.
file	Explicit Markdown output path.
overwrite	Whether an existing file may be replaced.

Value

The normalized output path, invisibly.

```
create_model_specification
```

Create a Complete Model Specification

Description

Combines a model contract, a successful readiness audit, an approved formula, and a validated prior specification into one backend-independent model specification.

Usage

```
create_model_specification(contract, audit, priors)
```

Arguments

contract	A gp3bayes_model_contract.
audit	A gp3bayes_readiness_audit.
priors	A gp3bayes_prior_specification.

Value

An object of class gp3bayes_model_specification.

```
create_prediction_contrast_profile
```

Create a Prediction Contrast Profile

Description

Create a Prediction Contrast Profile

Usage

```
create_prediction_contrast_profile(
  fit,
  variable,
  contrast_variable,
  contrast_levels = NULL,
  values = NULL,
  n = 40L,
  at = list(),
```

```

measure = c("difference", "ratio", "odds_ratio"),
include_group_effects = FALSE,
ndraws = NULL,
probs = c(0.025, 0.5, 0.975)
)

```

Arguments

fit	A fitted gp3bayes_fit.
variable	Numeric profile variable.
contrast_variable	Variable defining two contrasted levels.
contrast_levels	Optional two levels.
values	Optional profile values.
n	Number of values when values is omitted.
at	Named values for other predictors.
measure	"difference", "ratio", or "odds_ratio".
include_group_effects	Whether group effects are included.
ndraws	Optional posterior draws.
probs	Three interval probabilities.

Value

A gp3bayes_prediction_contrast_profile.

```
create_prediction_grid
```

Create a Governed Prediction Grid

Description

Creates a Cartesian prediction grid from declared model predictors. Numeric covariates are held at an observed typical value unless values are supplied explicitly through at.

Usage

```

create_prediction_grid(
  x,
  variables = NULL,
  at = list(),
  numeric_at = c("median", "mean"),
  max_rows = 5000L
)

```

Arguments

x	A gp3bayes fit or approved model specification.
variables	Optional variables to vary. By default the declared condition and non-identifier predictors are considered.
at	Named list of explicit values for selected variables.
numeric_at	One of "median" or "mean" for numeric covariates not supplied in at.
max_rows	Maximum permitted grid size.

Value

A data frame suitable for predict_model().

create_prediction_profile

Create a Numeric Prediction Profile

Description

Create a Numeric Prediction Profile

Usage

```
create_prediction_profile(
  fit,
  variable,
  values = NULL,
  n = 50L,
  at = list(),
  type = c("expected", "predictive", "linear", "median"),
  include_group_effects = FALSE,
  allow_new_levels = FALSE,
  ndraws = NULL,
  probs = c(0.025, 0.5, 0.975),
  seed = 1L
)
```

Arguments

fit	A fitted gp3bayes_fit.
variable	Numeric predictor to vary.
values	Optional explicit predictor values.
n	Number of values when values is omitted.
at	Named values holding other predictors fixed.
type	Prediction quantity.

include_group_effects	Whether group effects are included.
allow_new_levels	Whether new grouping levels are permitted.
ndraws	Optional posterior draws.
probs	Three interval probabilities.
seed	Predictive simulation seed where applicable.

Value

A gp3bayes_prediction_profile.

create_prediction_surface

Create a Two-Dimensional Prediction Surface

Description

Create a Two-Dimensional Prediction Surface

Usage

```
create_prediction_surface(
  fit,
  x,
  y,
  x_values = NULL,
  y_values = NULL,
  n = 30L,
  at = list(),
  type = c("expected", "predictive", "linear", "median"),
  include_group_effects = FALSE,
  allow_new_levels = FALSE,
  ndraws = NULL,
  probs = c(0.025, 0.5, 0.975),
  seed = 1L,
  max_rows = 2500L
)
```

Arguments

fit	A fitted gp3bayes_fit.
x, y	Numeric predictors.
x_values, y_values	Optional explicit predictor values.

n	Values per predictor when explicit values are omitted.
at	Named values holding other predictors fixed.
type	Prediction quantity.
include_group_effects	Whether group effects are included.
allow_new_levels	Whether new grouping levels are permitted.
ndraws	Optional posterior draws.
probs	Three interval probabilities.
seed	Predictive simulation seed.
max_rows	Maximum grid rows.

Value

A gp3bayes_prediction_surface.

create_predictive_distribution_atlas

Create a Posterior-Predictive Distribution Atlas

Description

Create a Posterior-Predictive Distribution Atlas

Usage

```
create_predictive_distribution_atlas(
  fit,
  ndraws = 500L,
  include_group_effects = TRUE,
  seed = 1L
)
```

Arguments

fit	A fitted gp3bayes_fit.
ndraws	Number of posterior predictive draws.
include_group_effects	Whether fitted group effects are included.
seed	Predictive seed.

Value

A gp3bayes_predictive_distribution_atlas.

```
create_predictor_scaling_sensitivity_specification
```

Create a Predictor-Scaling Sensitivity Specification

Description

Changes one already-scaled predictor by a declared scale factor and requires an explicit coefficient-prior scale for the new parameterisation. This avoids pretending that a common coefficient prior is automatically invariant to predictor scaling.

Usage

```
create_predictor_scaling_sensitivity_specification(
  specification,
  predictor,
  scale_factor,
  coefficient_scale,
  interaction_scale = NULL
)
```

Arguments

<code>specification</code>	An approved model specification.
<code>predictor</code>	A declared predictor that was scaled during preparation.
<code>scale_factor</code>	New scale divided by the original recorded scale. Values above one make the transformed predictor numerically smaller.
<code>coefficient_scale</code>	Explicit population-coefficient prior scale under the alternative parameterisation.
<code>interaction_scale</code>	Optional explicit interaction prior scale when the advanced separate-interaction prior is used.

Value

An approved alternative specification.

 create_prior_specification

Create a Backend-Independent Prior Specification

Description

Creates an inspectable prior table for one approved model family. The returned object contains no backend-specific prior objects and performs no sampling.

Usage

```
create_prior_specification(
  contract,
  baseline = NULL,
  intercept_scale = NULL,
  coefficient_scale = NULL,
  group_sd_scale = 1,
  residual_scale = NULL,
  correlation_eta = 2,
  student_df = 3
)
```

Arguments

- | | |
|-------------------|--|
| contract | A gp3bayes_model_contract created by create_model_contract() . |
| baseline | Numeric scalar describing the expected baseline outcome. For binary models this is a probability strictly between zero and one. The default is 0.5. For duration models this is a strictly positive baseline median in the recorded outcome unit and must be supplied. |
| intercept_scale | Optional positive numeric scalar for the normal intercept prior. Defaults to 1.5 for binary models and 1 for duration models. |
| coefficient_scale | Optional positive numeric scalar for normal population-level coefficient priors. Defaults to 1 for binary models and 0.5 for duration models. |
| group_sd_scale | Positive numeric scalar for half-Student-t group-level standard-deviation priors. |
| residual_scale | Optional positive numeric scalar for the half-Student-t residual standard-deviation prior. It applies only to the duration family and defaults to 1. |
| correlation_eta | Numeric scalar greater than or equal to one for the LKJ prior used when a participant-level random slope is requested. |
| student_df | Positive numeric scalar giving the degrees of freedom for half-Student-t scale priors. |

Details

Binary baseline probabilities are transformed with the logit function. Duration baseline medians are transformed with the natural logarithm.

Value

An object of class `gp3bayes_prior_specification`.

Examples

```
binary_contract <- create_model_contract(  
  family = "binary",  
  outcome_col = "selected",  
  participant_col = "participant_id"  
)  
  
create_prior_specification(  
  binary_contract,  
  baseline = 0.35  
)
```

```
create_publication_registry
```

Create a Publication Registry

Description

Create a Publication Registry

Usage

```
create_publication_registry(label = NULL)
```

Arguments

`label` Optional registry label.

Value

A `gp3bayes_publication_registry`.

`create_publication_table_set`*Extract Publication Tables from an Analysis Bundle*

Description

Extract Publication Tables from an Analysis Bundle

Usage

```
create_publication_table_set(x)
```

Arguments

`x` A gp3bayes_analysis_bundle.

Value

A named list of data frames suitable for downstream formatting.

`create_pupil_advanced_sensitivity_suite`*Create a pre-fit advanced pupillometry sensitivity suite*

Description

The suite materializes scientifically interpretable alternative model specifications without fitting, ranking, or choosing among them.

Usage

```
create_pupil_advanced_sensitivity_suite(  
  specification,  
  include = c("likelihood", "residual_scale", "autocorrelation", "temporal", "gp_kernel")  
)
```

Arguments

`specification` Baseline advanced specification.
`include` Character subset of "likelihood", "residual_scale", "autocorrelation", "temporal", and "gp_kernel".

Value

A gp3bayes_pupil_advanced_sensitivity_suite object.

```
create_pupil_arma_spec
```

Create an explicit bounded ARMA configuration

Description

Create an explicit bounded ARMA configuration

Usage

```
create_pupil_arma_spec(p = 1L, q = 0L, covariance = FALSE)
```

Arguments

p	Autoregressive order, constrained to 0–3.
q	Moving-average order, constrained to 0–2.
covariance	Logical; request covariance-form ARMA. In gp3bayes 0.5 this is permitted only for order (1,0), (0,1), or (1,1).

Value

A gp3bayes_pupil_arma_spec object.

```
create_pupil_contract
```

Create a governed pupil-timecourse contract

Description

Records the measurement and analysis declarations required for the restricted Gaussian hierarchical pupil-timecourse family. Contract creation performs no preprocessing, exclusion, correction, model fitting, or psychological interpretation.

Usage

```
create_pupil_contract(
  outcome_col,
  participant_col,
  trial_col,
  time_col,
  pupil_unit,
  sampling_frequency,
  time_unit = c("seconds", "milliseconds"),
  item_col = NULL,
  condition_col = NULL,
  timestamp_col = NULL,
```

```

eye = c("unknown", "left", "right", "combined"),
left_pupil_col = NULL,
right_pupil_col = NULL,
channel_audit_unit = NULL,
validity_col = NULL,
interpolation_col = NULL,
blink_col = NULL,
gaze_x_col = NULL,
gaze_y_col = NULL,
luminance_col = NULL,
contrast_col = NULL,
screen_width = NA_real_,
screen_height = NA_real_,
baseline_window = NULL,
baseline_method = c("unknown", "none", "subtract", "divide", "proportion_change",
  "percent_change"),
baseline_applied = FALSE,
pfe_corrected = FALSE,
pfe_method = NULL,
source_vendor = NA_character_,
device_model = NA_character_,
preprocessing_provenance = NA_character_,
upstream_package = NA_character_,
upstream_version = NA_character_,
notes = character()
)

```

Arguments

outcome_col	Numeric pupil-response column to model.
participant_col	Participant identifier column.
trial_col	Trial identifier column.
time_col	Event-relative time column.
pupil_unit	One of "millimetres", "metres", "pixels", "arbitrary_units", "standardized", "ratio", "proportion_change", or "percent_change".
sampling_frequency	Declared nominal sampling frequency in Hz.
time_unit	Unit of time_col: "seconds" or "milliseconds". Preparation converts the canonical event-time column to seconds and records that deterministic conversion.
item_col	Optional item/stimulus identifier.
condition_col	Optional experimental condition.
timestamp_col	Optional absolute or recording timestamp.
eye	Declared channel: "left", "right", "combined", or "unknown". The function never chooses an eye automatically.

<code>left_pupil_col, right_pupil_col</code>	Optional paired pupil channels retained only for left/right disagreement auditing. Either may equal <code>outcome_col</code> .
<code>channel_audit_unit</code>	Unit for paired audit channels; defaults to <code>pupil_unit</code> when either paired channel is declared.
<code>validity_col, interpolation_col, blink_col</code>	Optional measurement-quality indicator columns.
<code>gaze_x_col, gaze_y_col</code>	Optional gaze-position columns.
<code>luminance_col, contrast_col</code>	Optional visual-stimulus nuisance columns.
<code>screen_width, screen_height</code>	Optional screen dimensions in declared screen units; use <code>NA_real_</code> when unknown.
<code>baseline_window</code>	Optional two-element event-relative baseline window expressed in <code>time_unit</code> .
<code>baseline_method</code>	Declared upstream/current baseline state: "none", "subtract", "divide", "proportion_change", "percent_change", or "unknown".
<code>baseline_applied</code>	Whether baseline correction has already been applied.
<code>pfe_corrected</code>	Whether pupil-foreshortening correction was applied upstream.
<code>pfe_method</code>	Optional description of the upstream PFE method.
<code>source_vendor, device_model</code>	Optional source metadata. Missing metadata remain explicitly unknown.
<code>preprocessing_provenance</code>	Optional free-text provenance.
<code>upstream_package, upstream_version</code>	Optional upstream package metadata.
<code>notes</code>	Optional user notes.

Value

A `gp3bayes_pupil_contract`.

Governance boundary

The contract records decisions but does not detect blinks, interpolate, smooth, correct PFE, correct luminance, choose a baseline, or exclude data. `gp3bayes` does not automatically correct blink/data-loss, PFE, gaze-position, luminance, contrast, or baseline decisions recorded upstream. `gp3bayes` does not infer cognitive load, attention, arousal, stress, emotion, surprise, or effort from a pupil measurement or posterior pupil contrast. Interpretation remains the researcher's responsibility and must be justified by the study design, measurement context, and substantive scientific argument.

Examples

```
contract <- create_pupil_contract(
  outcome_col = "pupil_mm",
  participant_col = "participant_id",
  trial_col = "trial_id",
  time_col = "event_time",
  pupil_unit = "millimetres",
  sampling_frequency = 60,
  condition_col = "condition",
  eye = "combined"
)
contract
```

create_pupil_gp_spec *Create a Gaussian-process configuration for pupil trajectories*

Description

Create a Gaussian-process configuration for pupil trajectories

Usage

```
create_pupil_gp_spec(
  kernel = c("matern32", "matern52", "exp_quad"),
  basis = c("approximate", "exact"),
  k = 30L,
  scale = TRUE
)
```

Arguments

kernel	GP covariance kernel.
basis	"exact" or "approximate".
k	Number of Hilbert-space basis functions when basis = "approximate".
scale	Whether brms should internally scale GP predictors.

Value

A gp3bayes_pupil_gp_spec object.

`create_pupil_lfo_plan` *Create an explicit leave-future-out validation plan*

Description

The plan defines sequential training cut-points within a single ordered series. Execution is deliberately separate because it requires refitting.

Usage

```
create_pupil_lfo_plan(
  fit,
  initial_fraction = 0.6,
  horizon = 5L,
  step = 5L,
  max_refits = 8L
)
```

Arguments

<code>fit</code>	An advanced fitted model.
<code>initial_fraction</code>	Initial fraction of each series available for the earliest training set.
<code>horizon</code>	Number of future samples scored at each refit.
<code>step</code>	Number of samples by which the training cut moves.
<code>max_refits</code>	Maximum refits per model.

Value

A `gp3bayes_pupil_lfo_plan` object.

`create_pupil_measurement_model`
Create a governed measurement-uncertainty specification

Description

Declares known standard-error columns for pupil covariates and/or the pupil response. The object records uncertainty; it does not alter or impute data.

Usage

```
create_pupil_measurement_model(
  baseline_error = NULL,
  luminance_error = NULL,
  gaze_error = NULL,
  response_error = NULL,
  covariate_errors = NULL
)
```

Arguments

`baseline_error`, `luminance_error`, `gaze_error`
Optional standard-error column names for common adjustment variables.

`response_error` Optional known standard-error column for the pupil response.

`covariate_errors`
Optional named character vector mapping arbitrary covariate names to standard-error columns.

Value

A `gp3bayes_pupil_measurement_model` object.

```
create_pupil_missingness_spec
```

Create a governed missingness specification

Description

Create a governed missingness specification

Usage

```
create_pupil_missingness_spec(
  response = c("exclude", "model"),
  predictors = character(),
  assumptions = "MAR",
  auxiliary_predictors = character()
)
```

Arguments

`response` Either "exclude" or "model". "model" uses brms missing-response syntax and is interpreted under an explicitly declared MAR-oriented modelling assumption, not as proof that MAR holds.

`predictors` Character vector of predictor columns whose missing values should be modelled jointly.

assumptions Currently only "MAR" is accepted for modelling.
 auxiliary_predictors Optional fully observed columns to use in the missing-predictor submodels.

Value

A gp3bayes_pupil_missingness_spec object.

create_pupil_model_set

Create a named set of fitted pupil models

Description

Create a named set of fitted pupil models

Usage

```
create_pupil_model_set(
  ...,
  predictive_target = c("new_trial_known_participant", "new_participant",
    "future_segment", "new_sample_known_trial")
)
```

Arguments

... Fitted advanced or compatible brms-backed pupil models, or one named list of models.
 predictive_target Declared target for interpreting comparison.

Value

A gp3bayes_pupil_model_set object.

create_pupil_sensitivity_suite

Create a governed pupil sensitivity suite

Description

Declares scientifically consequential alternatives without choosing the alternative that produces the largest effect. The suite is inert until a scenario is explicitly materialized or results are supplied for comparison.

Usage

```
create_pupil_sensitivity_suite(
  specification,
  baseline_windows = list(),
  baseline_window_operation = NULL,
  baseline_operations = character(),
  interpolation_policy = character(),
  blink_adjacent_margins = numeric(),
  gaze_adjustment = character(),
  luminance_adjustment = character(),
  pfe_prepared = list(),
  smooth_basis_dimensions = integer(),
  autocorrelation = character(),
  analysis_windows = list()
)
```

Arguments

specification Baseline pupil model specification.

baseline_windows List of alternative two-element baseline windows.

baseline_window_operation Optional baseline transformation to pair with `baseline_windows` when the baseline specification used "none". This must be declared explicitly; gp3bayes never chooses one.

baseline_operations Alternative baseline transformations.

interpolation_policy "retain" and/or "exclude_flagged".

blink_adjacent_margins Non-negative margins in seconds; zero means no blink-adjacent deletion.

gaze_adjustment "none" and/or "declared_covariates".

luminance_adjustment "none" and/or "declared_covariate".

pfe_prepared Optional named list of explicitly prepared alternative pupil series (for example upstream corrected and uncorrected versions). gp3bayes does not perform PFE correction.

smooth_basis_dimensions Alternative approved basis dimensions.

autocorrelation Alternative "none"/"ar1" structures.

analysis_windows List of declared estimand windows in canonical event-time seconds.

Value

A gp3bayes_pupil_sensitivity plan.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

```
create_pupil_validation_plan
```

Create an explicit pupil predictive-validation plan

Description

Defines the prediction target before choosing a partition. The plan distinguishes observation-level known-trial prediction, new trials for known participants, new participants, and finite future time segments. Only non-missing model-outcome rows enter validation partitions.

Usage

```
create_pupil_validation_plan(
  x,
  target = c("new_trial_known_participant", "new_participant", "future_segment",
    "new_sample_known_trial"),
  K = 5L,
  future_fraction = 0.2,
  seed = 2026
)
```

Arguments

x	A prepared pupil object, pupil specification, or pupil fit.
target	One of "new_sample_known_trial", "new_trial_known_participant", "new_participant", or "future_segment".
K	Number of folds for K-fold targets.
future_fraction	Fraction at the end of each trial series held out for the future-segment target.
seed	Reproducibility seed.

Value

A gp3bayes_pupil_validation_plan with explicit fold/split membership and leakage checks.

Interpretation

Observation-wise validation is not presented as a universal default for temporally dependent pupil samples. The declared prediction target determines the partition.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

`create_random_slope_sensitivity_plan`*Create a Random-Slope Sensitivity Plan*

Description

Constructs approved random-intercept and random-slope specifications from a common prepared design. No model is fitted and neither structure is selected.

Usage

```
create_random_slope_sensitivity_plan(specification)
```

Arguments

`specification` An approved binary or duration specification.

Value

A `gp3bayes_random_slope_sensitivity_plan`.

`create_reporting_checklist`*Create a Reporting Checklist*

Description

Create a Reporting Checklist

Usage

```
create_reporting_checklist(x)
```

Arguments

`x` A fitted `gp3bayes_fit` or `gp3bayes_model_card`.

Value

A data frame describing evidence that is available for reporting.

```
create_sensitivity_suite_plan
```

Create a Sensitivity Suite Plan

Description

Creates a declarative plan for sensitivity analyses already supported by gp3bayes. Expensive refits are opt-in and are never started merely by creating a plan.

Usage

```
create_sensitivity_suite_plan(
  prior_scale = FALSE,
  powerscale = FALSE,
  psis_loo = FALSE,
  random_slope_plan = NULL,
  group_deletion_plan = NULL,
  alternative_estimands = list(),
  duration_unit = NULL,
  prior_scale_args = list(),
  powerscale_args = list(),
  psis_args = list(),
  random_slope_args = list(),
  group_deletion_args = list()
)
```

Arguments

<code>prior_scale</code>	Whether to run the family-specific prior-scale refit.
<code>powerscale</code>	Whether to run local power-scaling through priorsense.
<code>psis_loo</code>	Whether to compute PSIS-LOO for the reference fit.
<code>random_slope_plan</code>	Optional result of create_random_slope_sensitivity_plan() .
<code>group_deletion_plan</code>	Optional result of create_group_deletion_sensitivity_plan() .
<code>alternative_estimands</code>	Optional named list of already-computed estimands from coding/scaling/unit or other approved sensitivity fits.
<code>duration_unit</code>	Optional list containing estimand, multiplier, and optional tolerance for audit_duration_unit_invariance() .
<code>prior_scale_args</code>	Named argument list passed to the family-specific prior-scale sensitivity function.
<code>powerscale_args</code>	Named argument list passed to assess_powerscaled_sensitivity() .

psis_args Named argument list passed to compute_psis_loo().
 random_slope_args Named argument list passed to run_random_slope_sensitivity().
 group_deletion_args Named argument list passed to run_group_deletion_sensitivity().

Value

A gp3bayes_sensitivity_plan.

create_transformation_recipe
Create a Reusable Transformation Recipe

Description

Extracts the recorded condition coding, outcome mapping or unit conversion, numeric scaling registry, fixed-effects formula, and model-matrix columns from a prepared gp3bayes object.

Usage

```
create_transformation_recipe(prepared)
```

Arguments

prepared A binary or duration prepared object.

Value

A gp3bayes_transformation_recipe.

design_support_table *Design-Support Table*

Description

Design-Support Table

Usage

```
design_support_table(x)
```

Arguments

x A gp3bayes_design_support_audit.

Value

The design-support component table.

detect_binary_separation

Detect Separation in the Binary Fixed-Effects Screen

Description

Runs `detectseparation::detect_separation()` as a pre-fit fixed-effects screen. This screen does not replace the hierarchical model or prove that the Bayesian posterior is adequate.

Usage

```
detect_binary_separation(x, formula = NULL, data = NULL)
```

Arguments

<code>x</code>	An approved binary specification, or a data frame when formula is supplied.
<code>formula</code>	Optional fixed-effects binomial formula.
<code>data</code>	Optional data frame. It overrides data extracted from <code>x</code> .

Value

A `gp3bayes_separation_screen`.

diagnose_advanced_pupil_fit

Diagnose an advanced pupil fit

Description

Diagnose an advanced pupil fit

Usage

```
diagnose_advanced_pupil_fit(fit, rhat_threshold = 1.01, ess_threshold = 400)
```

Arguments

<code>fit</code>	An advanced pupil fit.
<code>rhat_threshold</code>	Maximum preferred R-hat.
<code>ess_threshold</code>	Minimum preferred bulk/tail ESS.

Value

A `gp3bayes_pupil_advanced_diagnostics` object.

diagnose_binary_fit *Diagnose a Fitted Binary Model*

Description

Computes rank-normalized R-hat, bulk and tail effective sample sizes, divergent-transition counts, maximum-tredepth saturation, and chain-level energy diagnostics for an approved binary fit.

Usage

```
diagnose_binary_fit(
  fit,
  rhat_pass = 1.01,
  rhat_fail = 1.05,
  ess_per_chain_pass = 100,
  ess_per_chain_fail = 50,
  maximum_tredepth_fraction = 0.01,
  ebfmi_pass = 0.3,
  ebfmi_fail = 0.2
)
```

Arguments

<code>fit</code>	A fitted <code>gp3bayes_fit</code> .
<code>rhat_pass</code>	R-hat value at or below which the component passes.
<code>rhat_fail</code>	R-hat value above which the component fails.
<code>ess_per_chain_pass</code>	Bulk or tail ESS per chain at or above which the component passes.
<code>ess_per_chain_fail</code>	Bulk or tail ESS per chain below which the component fails.
<code>maximum_tredepth_fraction</code>	Maximum fraction of post-warmup draws that may reach the configured maximum tredepth before the component fails.
<code>ebfmi_pass</code>	E-BFMI value at or above which the energy component passes.
<code>ebfmi_fail</code>	E-BFMI value below which the energy component fails.

Details

The overall status is "fail" when any component fails, "review" when any component requires review or cannot be assessed, and "pass" only when every component passes. A pass does not automatically establish convergence or posterior adequacy.

Value

A `gp3bayes_binary_diagnostics` object containing parameter, component, and chain-level diagnostic tables.

diagnose_duration_fit *Diagnose a Fitted Duration Model*

Description

Applies the package sampling-diagnostic contract to an approved hierarchical lognormal duration fit.

Usage

```
diagnose_duration_fit(
  fit,
  rhat_pass = 1.01,
  rhat_fail = 1.05,
  ess_per_chain_pass = 100,
  ess_per_chain_fail = 50,
  maximum_treedepth_fraction = 0.01,
  ebfmi_pass = 0.3,
  ebfmi_fail = 0.2
)
```

Arguments

fit	A gp3bayes_duration_fit.
rhat_pass	R-hat value at or below which the component passes.
rhat_fail	R-hat value above which the component fails.
ess_per_chain_pass	Bulk or tail ESS per chain at or above which the component passes.
ess_per_chain_fail	Bulk or tail ESS per chain below which the component fails.
maximum_treedepth_fraction	Maximum fraction of post-warmup draws that may reach the configured maximum treedepth before the component fails.
ebfmi_pass	E-BFMI value at or above which the energy component passes.
ebfmi_fail	E-BFMI value below which the energy component fails.

Details

The returned status reports prespecified numerical sampling thresholds. It does not automatically establish convergence or posterior adequacy.

Value

A gp3bayes_duration_diagnostics object.

diagnose_model_fit	<i>Diagnose an Approved gp3bayes Fit</i>
--------------------	--

Description

Family-neutral wrapper around `diagnose_binary_fit()` and `diagnose_duration_fit()`.

Usage

```
diagnose_model_fit(fit, ...)
```

Arguments

<code>fit</code>	A gp3bayes_fit.
<code>...</code>	Family-specific diagnostic arguments.

Value

A family-specific gp3bayes_sampling_diagnostics object.

diagnose_pupil_fit	<i>Diagnose temporal and sampling behaviour of a pupil fit</i>
--------------------	--

Description

Reports posterior R-hat/ESS summaries, NUTS sampler evidence when available, residual temporal drift, and residual autocorrelation. Thresholds are numerical review gates, not adequacy certification.

Usage

```
diagnose_pupil_fit(fit, ndraws = 200L, max_lag = 10L, max_cells = 3000000L)
```

Arguments

<code>fit</code>	Fitted pupil model.
<code>ndraws</code>	Draws used for expected-value residual summaries.
<code>max_lag</code>	Maximum residual ACF lag.
<code>max_cells</code>	Maximum draw-by-observation cells.

Value

A gp3bayes_pupil_diagnostics.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

diagnostic_dashboard_table	<i>Diagnostic Dashboard Table</i>
----------------------------	-----------------------------------

Description

Diagnostic Dashboard Table

Usage

diagnostic_dashboard_table(x)

Arguments

x A diagnostic dashboard.

Value

Component availability and status.

duration_pit_table	<i>Duration Probability-Integral-Transform Table</i>
--------------------	--

Description

Duration Probability-Integral-Transform Table

Usage

duration_pit_table(x)

Arguments

x A duration posterior predictive gp3bayes_prediction.

Value

Observation-level empirical posterior predictive PIT values.

duration_prediction_scores	<i>Duration Prediction Scores</i>
----------------------------	-----------------------------------

Description

Duration Prediction Scores

Usage

duration_prediction_scores(x, observed = NULL)

Arguments

- x A duration prediction object or positive numeric predictions.
- observed Optional positive observed durations.

Value

A one-row table of absolute and squared prediction errors on the response and log scales.

Examples

duration_prediction_scores(c(100, 120, 90), c(110, 115, 100))

duration_qq_table	<i>Duration Predictive Q-Q Table</i>
-------------------	--------------------------------------

Description

Duration Predictive Q-Q Table

Usage

duration_qq_table(x, probs = seq(0.05, 0.95, by = 0.05))

Arguments

- x A duration posterior predictive gp3bayes_prediction.
- probs Quantile probabilities.

Value

A quantile-comparison table.

duration_quantile_calibration
<i>Duration Quantile Calibration</i>

Description

Duration Quantile Calibration

Usage

```
duration_quantile_calibration(x, quantiles = c(0.1, 0.25, 0.5, 0.75, 0.9))
```

Arguments

- x A duration posterior predictive gp3bayes_prediction.
- quantiles Predictive quantiles to assess.

Value

A table comparing nominal predictive quantiles with empirical coverage below those quantiles.

duration_tail_check	<i>Duration Tail Check</i>
---------------------	----------------------------

Description

Duration Tail Check

Usage

```
duration_tail_check(x, threshold)
```

Arguments

- x A duration posterior predictive gp3bayes_prediction.
- threshold Positive duration threshold.

Value

A one-row table comparing observed and posterior predictive tail rates.

estimand_sensitivity_table

Estimand Sensitivity Table

Description

Estimand Sensitivity Table

Usage

```
estimand_sensitivity_table(x)
```

Arguments

x A gp3bayes_estimand_sensitivity.

Value

Alternative-versus-reference estimand summaries.

estimate_binocular_pupil_trajectory

Estimate joint binocular posterior trajectories

Description

Estimate joint binocular posterior trajectories

Usage

```
estimate_binocular_pupil_trajectory(
  fit,
  newdata = NULL,
  ndraws = 500L,
  probability = 0.95
)
```

Arguments

fit A binocular fit.
newdata Optional prediction grid.
ndraws Posterior draws.
probability Central interval probability.

Value

A gp3bayes_binocular_pupil_trajectory object.

`estimate_model_estimands`*Estimate the Approved Primary Estimands*

Description

Dispatches to design-standardised probability contrasts for binary fits and duration estimands for positive lognormal fits.

Usage

```
estimate_model_estimands(fit, ...)
```

Arguments

<code>fit</code>	A gp3bayes_fit.
<code>...</code>	Family-specific diagnostic arguments.

Value

A gp3bayes_estimand.

`estimate_pupil_auc`*Estimate area under a declared pupil-response window*

Description

Estimate area under a declared pupil-response window

Usage

```
estimate_pupil_auc(prediction, window, probability = 0.95)
```

Arguments

<code>prediction</code>	A pupil prediction object.
<code>window</code>	Prespecified event-relative time window.
<code>probability</code>	Credible probability.

Value

A pupil estimand. AUC units are pupil-unit times time-unit.

Examples

```
grid <- data.frame(.event_time = seq(0, 1, length.out = 5))
draws <- matrix(rnorm(500), nrow = 100)
prediction <- as_pupil_prediction_draws(draws, grid, "millimetres")
estimate_pupil_auc(prediction, c(0.2, 0.8))
```

`estimate_pupil_dynamic_contrast`*Estimate a dynamic posterior contrast between two pupil conditions*

Description

The contrast is evaluated pointwise on an explicitly supplied posterior trajectory (or its derivative).
No favorable time window is selected.

Usage

```
estimate_pupil_dynamic_contrast(  
  prediction,  
  contrast,  
  threshold = 0,  
  probability = 0.95  
)
```

Arguments

<code>prediction</code>	An advanced trajectory or derivative object.
<code>contrast</code>	Character vector of exactly two condition levels: first minus second.
<code>threshold</code>	Prespecified scientifically meaningful contrast threshold.
<code>probability</code>	Central posterior interval probability.

Value

A `gp3bayes_pupil_dynamic_contrast` object.

estimate_pupil_peak	<i>Estimate posterior peak pupil response inside a declared window</i>
---------------------	--

Description

Estimate posterior peak pupil response inside a declared window

Usage

```
estimate_pupil_peak(prediction, window, probability = 0.95)
```

Arguments

prediction	A pupil prediction object.
window	Prespecified event-relative time window.
probability	Credible probability.

Value

A pupil estimand with posterior uncertainty in the peak.

Examples

```
grid <- data.frame(.event_time = seq(0, 1, length.out = 5))
draws <- matrix(rnorm(500), nrow = 100)
prediction <- as_pupil_prediction_draws(draws, grid, "millimetres")
estimate_pupil_peak(prediction, c(0.2, 0.8))
```

estimate_pupil_peak_latency	<i>Estimate posterior peak latency inside a declared window</i>
-----------------------------	---

Description

Estimate posterior peak latency inside a declared window

Usage

```
estimate_pupil_peak_latency(prediction, window, probability = 0.95)
```

Arguments

prediction	A pupil prediction object.
window	Prespecified event-relative time window.
probability	Credible probability.

Value

A pupil estimand with posterior uncertainty in peak latency.

Examples

```
grid <- data.frame(.event_time = seq(0, 1, length.out = 5))
draws <- matrix(rnorm(500), nrow = 100)
prediction <- as_pupil_prediction_draws(draws, grid, "millimetres")
estimate_pupil_peak_latency(prediction, c(0.2, 0.8))
```

`estimate_pupil_residual_scale`

Estimate the residual-scale trajectory from a distributional model

Description

Estimate the residual-scale trajectory from a distributional model

Usage

```
estimate_pupil_residual_scale(
  fit,
  newdata = NULL,
  ndraws = 500L,
  probability = 0.95
)
```

Arguments

<code>fit</code>	An advanced fit.
<code>newdata</code>	Optional prediction data.
<code>ndraws</code>	Number of posterior draws.
<code>probability</code>	Central interval probability.

Value

A `gp3bayes_pupil_residual_scale` object.

```
estimate_pupil_response_parameters
```

Estimate nonlinear response-shape parameters

Description

Estimate nonlinear response-shape parameters

Usage

```
estimate_pupil_response_parameters(fit, probability = 0.95)
```

Arguments

fit	A response-shape fit.
probability	Central interval probability.

Value

A gp3bayes_pupil_response_parameters object.

```
estimate_pupil_threshold_duration
```

Estimate posterior duration above a prespecified dynamic threshold

Description

Duration is computed draw-by-draw from the dynamic contrast on its existing time grid. The threshold and direction must be supplied before interpretation; the function performs no threshold or window optimization.

Usage

```
estimate_pupil_threshold_duration(
  contrast,
  direction = c("above", "below", "absolute"),
  threshold = contrast$threshold,
  probability = 0.95
)
```

Arguments

contrast	A dynamic-contrast object.
direction	"above", "below", or "absolute".
threshold	Optional threshold overriding the contrast's stored threshold.
probability	Central interval probability.

Value

A gp3bayes_pupil_threshold_duration object.

estimate_pupil_trajectory

Estimate posterior pupil trajectories

Description

Summarises a finite, declared prediction grid with pointwise or grid-wise simultaneous posterior bands.

Usage

```
estimate_pupil_trajectory(
  prediction,
  probability = 0.95,
  interval = c("pointwise", "simultaneous")
)
```

Arguments

prediction	A gp3bayes_pupil_prediction.
probability	Credible probability.
interval	"pointwise" or "simultaneous".

Value

A gp3bayes_pupil_trajectory.

Uncertainty

"simultaneous" constructs a grid-wise band from the empirical posterior maximum standardized deviation over the supplied finite grid. It is not a universal continuous-time confidence band.

Examples

```
grid <- data.frame(.event_time = seq(0, 1, length.out = 5))
draws <- matrix(rnorm(500), nrow = 100)
prediction <- as_pupil_prediction_draws(draws, grid, "millimetres")
estimate_pupil_trajectory(prediction)
```

estimate_pupil_trajectory_derivative

Estimate posterior temporal derivatives of a pupil trajectory

Description

Computes finite-difference posterior derivatives on the prediction grid. This is a descriptive functional estimand: it does not automatically define physiological onset, changepoints, or cognitively meaningful phases.

Usage

```
estimate_pupil_trajectory_derivative(
  prediction,
  order = 1L,
  probability = 0.95
)
```

Arguments

prediction	A gp3bayes_pupil_advanced_trajectory object.
order	Derivative order, 1 (velocity/slope) or 2 (acceleration/curvature).
probability	Central posterior interval probability.

Value

A gp3bayes_pupil_trajectory_derivative object.

estimate_pupil_window *Estimate a declared-window mean pupil response*

Description

Estimate a declared-window mean pupil response

Usage

```
estimate_pupil_window(prediction, window, probability = 0.95)
```

Arguments

prediction	A pupil prediction object.
window	Prespecified event-relative time window.
probability	Credible probability.

Value

A gp3bayes_pupil_estimand.

Examples

```
grid <- data.frame(.event_time = seq(0, 1, length.out = 5))
draws <- matrix(rnorm(500), nrow = 100)
prediction <- as_pupil_prediction_draws(draws, grid, "millimetres")
estimate_pupil_window(prediction, c(0.2, 0.8))
```

estimate_standardized_duration_estimands

Estimate Design-Standardised Duration Estimands

Description

Produces posterior draws of average conditional medians, their difference and ratio, the average log-duration contrast, and a posterior predictive upper quantile under each focal-condition level.

Usage

```
estimate_standardized_duration_estimands(
  fit,
  target_data = NULL,
  target_scale = c("prepared", "raw"),
  predictive_quantile = 0.9,
  ndraws = NULL,
  include_group_effects = FALSE,
  seed = 1L
)
```

Arguments

fit	An approved gp3bayes duration fit.
target_data	Optional target covariate distribution.
target_scale	Whether supplied target data are raw or prepared.
predictive_quantile	Predictive quantile probability.
ndraws	Optional number of posterior draws.
include_group_effects	Whether group-level effects are included.
seed	Seed used for posterior predictive draws.

Value

A gp3bayes_estimand.

```
estimate_standardized_probability_contrast
```

Estimate a Design-Standardised Binary Probability Contrast

Description

Replaces the focal-condition value across a declared target covariate distribution, obtains population-level expected probabilities using `brms::posterior_epred()`, and averages within each posterior draw.

Usage

```
estimate_standardized_probability_contrast(
  fit,
  target_data = NULL,
  target_scale = c("prepared", "raw"),
  ndraws = NULL,
  include_group_effects = FALSE
)
```

Arguments

<code>fit</code>	An approved <code>gp3bayes</code> binary fit.
<code>target_data</code>	Optional target covariate distribution.
<code>target_scale</code>	Whether supplied target data are raw or already prepared.
<code>ndraws</code>	Optional number of posterior draws.
<code>include_group_effects</code>	Whether recorded group-level effects are included. The default <code>FALSE</code> targets population-level predictions.

Value

A `gp3bayes_estimand` with probability-difference draws.

```
evaluate_pathological_simulation
```

Evaluate a Pathological Simulation Against the Contract Gate

Description

Evaluate a Pathological Simulation Against the Contract Gate

Usage

```
evaluate_pathological_simulation(x)
```


Arguments

x A gp3bayes_pathological_simulation.

Value

A gp3bayes_pathology_evaluation.

evidence_inventory_table
Evidence Inventory Table

Description

Evidence Inventory Table

Usage

```
evidence_inventory_table(x)
```

Arguments

x A gp3bayes_evidence_inventory.

Value

Inventory metadata.

extract_expected_predictions
Extract Expected Posterior Predictions

Description

Extract Expected Posterior Predictions

Usage

```
extract_expected_predictions(  
  fit,  
  newdata = NULL,  
  include_group_effects = FALSE,  
  allow_new_levels = FALSE,  
  ndraws = NULL  
)
```

Arguments

<code>fit</code>	A fitted <code>gp3bayes_fit</code> .
<code>newdata</code>	Optional data frame. NULL uses the fitted prepared data.
<code>include_group_effects</code>	Whether fitted group-level effects are included.
<code>allow_new_levels</code>	Whether new grouping levels are permitted by brms.
<code>ndraws</code>	Optional number of posterior draws.

Value

A numeric matrix of conditional expected-response draws.

`extract_linear_predictions`
Extract Linear-Predictor Draws

Description

Extract Linear-Predictor Draws

Usage

```
extract_linear_predictions(
  fit,
  newdata = NULL,
  include_group_effects = FALSE,
  allow_new_levels = FALSE,
  ndraws = NULL
)
```

Arguments

<code>fit</code>	A fitted <code>gp3bayes_fit</code> .
<code>newdata</code>	Optional data frame. NULL uses the fitted prepared data.
<code>include_group_effects</code>	Whether fitted group-level effects are included.
<code>allow_new_levels</code>	Whether new grouping levels are permitted by brms.
<code>ndraws</code>	Optional number of posterior draws.

Value

A numeric matrix on the model linear-predictor scale.

 extract_log_likelihood

Extract Pointwise Log-Likelihood Draws

Description

Extract Pointwise Log-Likelihood Draws

Usage

```
extract_log_likelihood(
  fit,
  newdata = NULL,
  include_group_effects = TRUE,
  ndraws = NULL
)
```

Arguments

fit	A fitted gp3bayes_fit.
newdata	Optional prediction data. NULL uses the fitted data.
include_group_effects	Whether fitted group-level effects are included.
ndraws	Optional number of posterior draws.

Value

A numeric matrix with posterior draws in rows and observations in columns.

 extract_posterior_draws

Extract Posterior Draws from a gp3bayes Fit

Description

Converts the fitted brms posterior into a standard posterior draws representation without changing the fitted model.

Usage

```
extract_posterior_draws(
  fit,
  variables = NULL,
  regex = NULL,
  format = c("array", "matrix", "df", "rvars")
)
```

Arguments

fit	A fitted gp3bayes_fit.
variables	Optional exact posterior variable names.
regex	Optional regular expression used to retain posterior variables.
format	One of "array", "matrix", "df", or "rvars".

Value

A posterior draws object in the requested format.

extract_posterior_predictions

Extract Posterior Predictive Draws

Description

Extract Posterior Predictive Draws

Usage

```
extract_posterior_predictions(
  fit,
  newdata = NULL,
  include_group_effects = FALSE,
  allow_new_levels = FALSE,
  ndraws = NULL,
  seed = 1L
)
```

Arguments

fit	A fitted gp3bayes_fit.
newdata	Optional data frame. NULL uses the fitted prepared data.
include_group_effects	Whether fitted group-level effects are included.
allow_new_levels	Whether new grouping levels are permitted by brms.
ndraws	Optional number of posterior draws.
seed	Non-negative seed used for posterior predictive simulation.

Value

A numeric matrix of new-outcome posterior predictive draws.

 extract_sampler_diagnostics

Extract NUTS Sampler Diagnostics

Description

Extract NUTS Sampler Diagnostics

Usage

```
extract_sampler_diagnostics(fit)
```

Arguments

`fit` A fitted gp3bayes_fit.

Value

A data frame returned from `brms::nuts_params()`.

fit_advanced_pupil_model

Fit an advanced pupil model using rstan

Description

Fit an advanced pupil model using rstan

Usage

```
fit_advanced_pupil_model(
  specification,
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = min(2L, chains),
  seed = 2026,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)
```

Arguments

`specification` An advanced pupil specification.
`chains`, `iter`, `warmup`, `cores`, `seed`, `adapt_delta`, `max_treedepth`, `refresh`
 Restricted MCMC controls.

```
fit_advanced_pupil_model_backend
```

Fit an advanced pupil model through an approved brms backend

Description

Fit an advanced pupil model through an approved brms backend

Usage

```
fit_advanced_pupil_model_backend(
  specification,
  backend = c("rstan", "cmdstanr"),
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = min(2L, chains),
  seed = 2026,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)
```

Arguments

`specification` An advanced pupil specification.

`backend` "rstan" or "cmdstanr".

`chains`, `iter`, `warmup`, `cores`, `seed`, `adapt_delta`, `max_treedepth`, `refresh`
Restricted MCMC controls.

Value

A `gp3bayes_pupil_advanced_fit` object.

```
fit_advanced_pupil_model_cmdstanr
```

Fit an advanced pupil model using cmdstanr

Description

Fit an advanced pupil model using cmdstanr

Usage

```
fit_advanced_pupil_model_cmdstanr(
  specification,
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = min(2L, chains),
  seed = 2026,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)
```

Arguments

specification An advanced pupil specification.
 chains, iter, warmup, cores, seed, adapt_delta, max_treedepth, refresh
 Restricted MCMC controls.

fit_binary_model	<i>Fit an Approved Hierarchical Binary Model</i>
------------------	--

Description

Fits an approved binary model specification using full MCMC sampling through the fixed brms and rstan route.

Usage

```
fit_binary_model(
  specification,
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = .gp3b_default_cores(chains),
  seed = 1L,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)
```

Arguments

specification A gp3bayes_binary_model_specification.
 chains Number of MCMC chains.
 iter Total iterations per chain, including warmup.

warmup	Warmup iterations per chain.
cores	Number of processor cores. It cannot exceed chains.
seed	Non-negative integer random-number seed.
adapt_delta	Target acceptance probability for the No-U-Turn sampler.
max_treedepth	Maximum tree depth for the No-U-Turn sampler.
refresh	Console progress refresh interval. Use zero to suppress iteration progress output.

Details

The function fixes the likelihood to Bernoulli, the link to logit, the interface to brms, the sampling backend to rstan, and the algorithm to full MCMC sampling. It does not expose arbitrary backend arguments.

A returned fit is not evidence of convergence, posterior adequacy, causal identification, or substantive validity. Those assessments require separate diagnostic and reporting gates.

Value

A gp3bayes_binary_fit containing the fitted backend object, original specification, restricted translation, and recorded sampling settings.

Examples

```
if (
  requireNamespace("brms", quietly = TRUE) &&
  requireNamespace("rstan", quietly = TRUE) &&
  identical(
    validate_backend_environment(
      "rstan",
      compile_test = FALSE,
      strict = FALSE
    )$status,
    "pass"
  )
) {
  simulation <- simulate_hierarchical_binary_data(
    n_participants = 8,
    trials_per_participant = 6,
    n_items = 4,
    random_slope_sd = 0,
    seed = 2026
  )

  contract <- create_model_contract(
    family = "binary",
    outcome_col = "selected",
    participant_col = "participant_id",
    item_col = "item_id",
    trial_col = "trial_id",
    condition_col = "condition"
  )
}
```



```

prepared <- prepare_hierarchical_binary_data(
  simulation$data,
  contract,
  condition_levels = c("control", "treatment")
)

specification <- specify_binary_model(prepared, baseline = 0.35)

fit <- fit_binary_model(
  specification,
  chains = 2,
  iter = 200,
  warmup = 100,
  cores = 2,
  seed = 2026,
  refresh = 0
)
}

```

fit_binary_model_backend

Fit a Binary Model with a Selected Full-MCMC Backend

Description

The formula and family remain contract-restricted. The only selectable implementation detail is whether brms delegates full MCMC sampling to rstan or CmdStanR.

Usage

```

fit_binary_model_backend(
  specification,
  backend = c("rstan", "cmdstanr"),
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = .gp3b_default_cores(chains),
  seed = 1L,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)

fit_binary_model_cmdstanr(
  specification,
  chains = 4L,

```

```

    iter = 2000L,
    warmup = 1000L,
    cores = .gp3b_default_cores(chains),
    seed = 1L,
    adapt_delta = 0.95,
    max_treedepth = 12L,
    refresh = 0L
  )

```

Arguments

specification	An approved binary model specification.
backend	Either "rstan" or "cmdstanr".
chains	Number of chains.
iter	Total iterations per chain.
warmup	Warmup iterations per chain.
cores	Number of cores, not exceeding chains.
seed	Non-negative integer seed.
adapt_delta	NUTS target acceptance probability.
max_treedepth	NUTS maximum tree depth.
refresh	Progress refresh interval.

Value

A gp3bayes_backend_portable_fit.

fit_binocular_pupil_model

Fit a joint binocular pupil model

Description

Fit a joint binocular pupil model

Usage

```

fit_binocular_pupil_model(
  specification,
  backend = c("rstan", "cmdstanr"),
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = min(2L, chains),
  seed = 2026,

```

```

    adapt_delta = 0.95,
    max_treedepth = 12L,
    refresh = 0L
  )

```

Arguments

specification A binocular specification.
backend rstan or cmdstanr.
chains, iter, warmup, cores, seed, adapt_delta, max_treedepth, refresh Restricted sampling controls.

fit_duration_model	<i>Fit an Approved Hierarchical Lognormal Duration Model</i>
--------------------	--

Description

Fits an approved strictly positive uncensored duration model using full MCMC through the fixed brms and rstan route.

Usage

```

fit_duration_model(
  specification,
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = .gp3b_default_cores(chains),
  seed = 1L,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)

```

Arguments

specification A gp3bayes_duration_model_specification.
chains Number of MCMC chains.
iter Total iterations per chain.
warmup Warmup iterations per chain.
cores Processor cores, not exceeding chains.
seed Non-negative integer seed.
adapt_delta Target NUTS acceptance probability.
max_treedepth Maximum NUTS tree depth.
refresh Console progress refresh interval.

Details

The likelihood is fixed to lognormal, the link to identity on the mean-log parameter, the interface to brms, the backend to rstan, and the algorithm to full MCMC sampling. A returned fit does not establish convergence or posterior adequacy.

Value

A gp3bayes_duration_fit.

fit_duration_model_backend

Fit a Duration Model with a Selected Full-MCMC Backend

Description

Fit a Duration Model with a Selected Full-MCMC Backend

Usage

```
fit_duration_model_backend(
  specification,
  backend = c("rstan", "cmdstanr"),
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = .gp3b_default_cores(chains),
  seed = 1L,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)

fit_duration_model_cmdstanr(
  specification,
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = .gp3b_default_cores(chains),
  seed = 1L,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)
```

Arguments

specification	An approved duration model specification.
backend	Either "rstan" or "cmdstanr".
chains	Number of chains.
iter	Total iterations per chain.
warmup	Warmup iterations per chain.
cores	Number of cores, not exceeding chains.
seed	Non-negative integer seed.
adapt_delta	NUTS target acceptance probability.
max_treedepth	NUTS maximum tree depth.
refresh	Progress refresh interval.

Value

A gp3bayes_backend_portable_fit.

fit_pupil_model	<i>Fit a pupil model through the fixed rstan route</i>
-----------------	--

Description

Fit a pupil model through the fixed rstan route

Usage

```
fit_pupil_model(
  specification,
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = min(2L, chains),
  seed = 2026,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)
```

Arguments

specification Approved pupil model specification.

chains, iter, warmup, cores, seed
 Sampling controls. Package-controlled cores are capped at two.

adapt_delta, max_treedepth, refresh
 Fixed safe sampling controls.

Value

A gp3bayes_pupil_fit.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

```
fit_pupil_model_backend
```

Fit the restricted pupil time-course model with an approved backend

Description

Fits only a specify_pupil_timecourse_model() specification through brms using full MCMC sampling and either rstan or cmdstanr.

Usage

```
fit_pupil_model_backend(
  specification,
  backend = c("rstan", "cmdstanr"),
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = min(2L, chains),
  seed = 2026,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)
```

Arguments

```
specification  Approved pupil model specification.
backend        "rstan" or "cmdstanr".
chains, iter, warmup, cores, seed
               Sampling controls. Package-controlled cores are capped at two.
adapt_delta, max_treedepth, refresh
               Fixed safe sampling controls.
```

Value

A gp3bayes_pupil_fit.

Governance boundary

This interface accepts no arbitrary formula, family, Stan program, inference algorithm, or unrestricted backend arguments. A returned fit does not establish convergence, adequacy, predictive validity, or psychological interpretation.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

```
fit_pupil_model_cmdstanr
```

Fit a pupil model through the fixed cmdstanr route

Description

Fit a pupil model through the fixed cmdstanr route

Usage

```
fit_pupil_model_cmdstanr(
  specification,
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = min(2L, chains),
  seed = 2026,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L
)
```

Arguments

`specification` Approved pupil model specification.
`chains, iter, warmup, cores, seed` Sampling controls. Package-controlled cores are capped at two.
`adapt_delta, max_treedepth, refresh` Fixed safe sampling controls.

Value

A `gp3bayes_pupil_fit`.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

```
fit_pupil_response_shape_model
```

Fit the experimental nonlinear response-shape model

Description

Fit the experimental nonlinear response-shape model

Usage

```
fit_pupil_response_shape_model(
  specification,
  backend = c("rstan", "cmdstanr"),
  chains = 4L,
  iter = 2500L,
  warmup = 1250L,
  cores = min(2L, chains),
  seed = 2026,
  adapt_delta = 0.97,
  max_treedepth = 13L,
  refresh = 0L
)
```

Arguments

`specification` A response-shape specification.

`backend` rstan or cmdstanr.

`chains`, `iter`, `warmup`, `cores`, `seed`, `adapt_delta`, `max_treedepth`, `refresh`
Restricted sampling controls.

```
freeze_analysis_manifest
```

Freeze an Analysis Manifest

Description

Computes a deterministic hash over analysis-defining fields. When `file` is supplied the frozen manifest is written explicitly to that path. No file is written when `file = NULL`.

Usage

```
freeze_analysis_manifest(manifest, file = NULL, overwrite = FALSE)
```


Arguments

manifest	A valid analysis manifest.
file	Optional explicit .rds output path.
overwrite	Whether an existing explicit output file may be replaced.

Value

A frozen gp3bayes_analysis_manifest.

freeze_gp3bayes_schema

Freeze a gp3bayes Object Schema

Description

Marks a captured schema as frozen and optionally writes it to an explicit RDS path. When file = NULL, no file is written.

Usage

```
freeze_gp3bayes_schema(schema, file = NULL, overwrite = FALSE)
```

Arguments

schema	A captured schema or gp3bayes object.
file	Optional explicit .rds path.
overwrite	Whether an existing file may be replaced.

Value

The frozen schema, invisibly when written.

gazeppoint_pupil_mapping_table

Return the Gazeppoint pupil mapping audit table

Description

Return the Gazeppoint pupil mapping audit table

Usage

```
gazeppoint_pupil_mapping_table(x)
```

Arguments

x A result from `inspect_gazepoint_pupil_schema()` or a compatible data frame to inspect.

Value

A data frame with documented fields, roles, units, eye, and presence.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

```
gp3bayes_specification_traceability
```

Specification-Closure Traceability Matrix

Description

Returns an auditable mapping between the Phase-0 closure requirements and their first-class implementation points.

Usage

```
gp3bayes_specification_traceability()
```

Value

A data frame.

```
grouped_prediction_check
```

Grouped Posterior Predictive Check

Description

Grouped Posterior Predictive Check

Usage

```
grouped_prediction_check(
  fit,
  group,
  ndraws = 1000L,
  probs = c(0.025, 0.5, 0.975),
  seed = 1L
)
```

Arguments

fit	A fitted gp3bayes_fit.
group	Name of a grouping column in the prepared model data.
ndraws	Number of posterior predictive draws.
probs	Posterior interval probabilities.
seed	Non-negative seed.

Value

A gp3bayes_group_prediction_check.

group_deletion_sensitivity_table
<i>Group-Deletion Sensitivity Table</i>

Description

Group-Deletion Sensitivity Table

Usage

group_deletion_sensitivity_table(x)

Arguments

x	A gp3bayes_group_deletion_sensitivity.
---	--

Value

Omitted-unit estimand summaries.

group_effect_draws_table
<i>Group-Level Posterior Draw Table</i>

Description

Group-Level Posterior Draw Table

Usage

```
group_effect_draws_table(
  fit,
  groups = NULL,
  coefficients = NULL,
  ndraws = NULL,
  seed = 1L,
  max_rows = 1000000L
)
```

Arguments

fit	A fitted gp3bayes_fit.
groups	Optional grouping factors.
coefficients	Optional group-level coefficients.
ndraws	Optional number of draws retained.
seed	Seed used only for draw subsampling.
max_rows	Maximum permitted long-format rows.

Value

A long posterior draw table on the model linear-predictor scale.

group_effect_rank_probability_table

Group-Level Rank-Probability Table

Description

Group-Level Rank-Probability Table

Usage

```
group_effect_rank_probability_table(
  fit,
  group,
  coefficient = "Intercept",
  ndraws = 1000L,
  seed = 1L
)
```

Arguments

fit	A fitted gp3bayes_fit.
group	One grouping-factor name.
coefficient	One group-level coefficient.
ndraws	Number of posterior draws.
seed	Draw-subsampling seed.

Value

Descriptive posterior rank probabilities. Rank 1 is the largest group-level deviation.

group_effect_table	<i>Group-Level Effect Table</i>
--------------------	---------------------------------

Description

Group-Level Effect Table

Usage

```
group_effect_table(fit, groups = NULL, probs = c(0.025, 0.975))
```

Arguments

fit	A fitted gp3bayes_fit.
groups	Optional grouping factors to retain.
probs	Lower and upper credible interval probabilities.

Value

A tidy data frame of estimated group-level deviations.

```
group_prediction_summary
```

Group Prediction Summary

Description

Aggregates posterior prediction draws over one or more columns already present in the prediction data.

Usage

```
group_prediction_summary(x, by, probs = c(0.025, 0.5, 0.975))
```

Arguments

x	A gp3bayes_prediction.
by	Character vector naming grouping columns in x\$newdata.
probs	Three probabilities for group-level posterior intervals.

Value

A group-level posterior prediction table.

```
identify_identifier_like_predictors
```

Identify Identifier-Like Numeric Predictors

Description

Applies conservative heuristics to declared numeric predictors. A flagged predictor is a review signal only; explicit declaration in a contract is never silently overridden.

Usage

```
identify_identifier_like_predictors(
  data,
  contract,
  unique_fraction = 0.9,
  integer_fraction = 0.98,
  monotone_correlation = 0.98
)
```

Arguments

data	A data frame.
contract	An approved model contract.
unique_fraction	Fraction of rows that must be unique before a predictor can be considered identifier-like.
integer_fraction	Fraction of finite values that must be integer-like.
monotone_correlation	Absolute correlation with row order used as a heuristic for sequence-like identifiers.

Value

A gp3bayes_identifier_predictor_audit object.

identify_loo_influential_observations
Identify Influential PSIS-LOO Observations

Description

Identify Influential PSIS-LOO Observations

Usage

```
identify_loo_influential_observations(x, threshold = NULL, data = NULL)
```

Arguments

x	A gp3bayes_psis_loo or raw loo object.
threshold	Optional explicit Pareto-k threshold.
data	Optional observation-level data to append.

Value

A data frame of flagged observations.

identify_mcmc_issues *Identify MCMC Diagnostic Flags*

Description

Identify MCMC Diagnostic Flags

Usage

```
identify_mcmc_issues(  
  x,  
  rhat_threshold = 1.01,  
  min_bulk_ess = 400,  
  min_tail_ess = 400,  
  max_mcse_fraction = 0.1  
)
```

Arguments

x A fitted gp3bayes object or an MCMC diagnostic table.

rhat_threshold R-hat value above which a parameter is flagged.

min_bulk_ess Minimum bulk effective sample size.

min_tail_ess Minimum tail effective sample size.

max_mcse_fraction Maximum MCSE-to-posterior-SD fraction.

Value

A parameter-level flag table. Flags request review; they do not establish or negate model adequacy.

Examples

```
d <- data.frame(  
  variable = c("a", "b"),  
  sd = c(1, 1),  
  rhat = c(1.00, 1.03),  
  ess_bulk = c(1000, 150),  
  ess_tail = c(900, 120),  
  mcse_mean = c(0.02, 0.15)  
)  
identify_mcmc_issues(d)
```

`inspect_gazepoint_pupil_schema`*Inspect verified Gazepoint pupil and gaze fields*

Description

Compares column names with the documented Open Gaze API field identifiers. The inspector reports candidates and ambiguity but never chooses a pupil channel automatically. Export variants that use other names remain unrecognized rather than being guessed.

Usage

```
inspect_gazepoint_pupil_schema(data)
```

Arguments

<code>data</code>	A data frame containing a Gazepoint export or API record table.
-------------------	---

Details

LPD and RPD are documented pixel diameters. LPUPILD and RPUPIILD are documented in metres. These scales are intentionally kept distinct.

Value

A `gp3bayes_gazepoint_pupil_schema` with detected fields, pupil-channel candidates, and an audit table.

Governance boundary

Recognition is schema evidence only. The function does not establish device validity, select an eye, convert units, correct PFE, or infer preprocessing history.

Examples

```
x <- data.frame(TIME = 0:2 / 60, LPD = c(32, 33, 31), LPV = 1)
inspect_gazepoint_pupil_schema(x)
```

interaction_prior_summary
<i>Summarise Separate Interaction Prior Metadata</i>

Description

Summarise Separate Interaction Prior Metadata

Usage

interaction_prior_summary(specification)

Arguments

specification An advanced binary or duration specification.

Value

A one-row data frame.

invert_transformation_recipe
<i>Invert a Recorded Transformation Recipe</i>

Description

Reconstructs a raw-scale representation from prepared data when the recorded transformations are invertible. This is intended for replay tests and controlled sensitivity construction, not recovery of discarded rows.

Usage

invert_transformation_recipe(data, recipe)

Arguments

data Prepared-scale data.
recipe A transformation recipe or prepared object.

Value

A raw-scale data frame.

loo_diagnostic_table	<i>LOO Diagnostic Table</i>
----------------------	-----------------------------

Description

LOO Diagnostic Table

Usage

```
loo_diagnostic_table(x)
```

Arguments

x A gp3bayes_psis_loo or loo object.

Value

Observation-level Pareto-k diagnostics.

loo_flagged_data	<i>Flagged LOO Observation Data</i>
------------------	-------------------------------------

Description

Flagged LOO Observation Data

Usage

```
loo_flagged_data(x, threshold = 0.7)
```

Arguments

x A LOO object or pointwise table.
threshold Explicit Pareto-k threshold.

Value

Rows meeting the requested threshold.

loo_group_influence_table
<i>Aggregate LOO Influence by a Declared Group</i>

Description

Aggregate LOO Influence by a Declared Group

Usage

loo_group_influence_table(x, group, data = NULL)

Arguments

- | | |
|-------|---|
| x | A LOO influence atlas, pointwise LOO table, gp3bayes PSIS-LOO, or raw loo object. |
| group | Grouping-column name. |
| data | Optional observation-level data when needed. |

Value

A descriptive group-level influence table.

loo_influence_atlas_table
<i>LOO Influence Atlas Table</i>

Description

LOO Influence Atlas Table

Usage

loo_influence_atlas_table(x)

Arguments

- | | |
|---|---------------------------------|
| x | A gp3bayes_loo_influence_atlas. |
|---|---------------------------------|

Value

The complete pointwise atlas table.

`loo_influence_summary` *LOO Influence Summary*

Description

LOO Influence Summary

Usage

```
loo_influence_summary(x)
```

Arguments

`x` A LOO object or pointwise LOO table.

Value

A one-row influence summary.

`loo_pointwise_table` *Pointwise LOO Table*

Description

Pointwise LOO Table

Usage

```
loo_pointwise_table(x, data = NULL)
```

Arguments

`x` A gp3bayes PSIS-LOO or raw loo object.
`data` Optional observation-level data with matching rows.

Value

Pointwise LOO estimates and Pareto-k diagnostics.

loo_summary_table	<i>LOO Summary Table</i>
-------------------	--------------------------

Description

LOO Summary Table

Usage

loo_summary_table(x)

Arguments

x A gp3bayes_psis_loo or loo object.

Value

A tidy table of LOO estimates and standard errors.

manifest_comparison_table	<i>Manifest-Comparison Table</i>
---------------------------	----------------------------------

Description

Manifest-Comparison Table

Usage

manifest_comparison_table(x)

Arguments

x A gp3bayes_manifest_comparison.

Value

Manifest component comparisons.

```
materialize_pupil_advanced_sensitivity_scenario
```

Materialize one advanced sensitivity scenario

Description

Materialize one advanced sensitivity scenario

Usage

```
materialize_pupil_advanced_sensitivity_scenario(suite, scenario)
```

Arguments

suite	A sensitivity suite.
scenario	Scenario name from suite\$scenarios.

Value

An advanced pupil specification.

```
materialize_pupil_sensitivity_scenario
```

Materialize one declared pupil sensitivity scenario

Description

Creates the alternate prepared/specification state for a declared scenario. This does not fit the model. Analysis-window scenarios are returned as estimand instructions. PFE scenarios select only a user-supplied upstream prepared alternative; no PFE correction is performed by gp3bayes.

Usage

```
materialize_pupil_sensitivity_scenario(suite, scenario_id)
```

Arguments

suite	Pupil sensitivity suite.
scenario_id	Scenario identifier from pupil_sensitivity_table().

Value

A list containing the materialized specification and/or declared estimand window.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

mcmc_diagnostic_table *Modern MCMC Diagnostic Table*

Description

Computes rank-normalised R-hat, bulk ESS, tail ESS, and Monte Carlo standard error summaries through the posterior package.

Usage

```
mcmc_diagnostic_table(fit, variables = NULL, regex = NULL)
```

Arguments

fit	A fitted gp3bayes_fit.
variables	Optional exact posterior variable names.
regex	Optional posterior-variable regular expression.

Value

A data frame with posterior diagnostics by variable.

missingness_audit_table
Missingness-Audit Table

Description

Missingness-Audit Table

Usage

```
missingness_audit_table(x)
```

Arguments

x	A gp3bayes_missingness_audit.
---	-------------------------------

Value

Column-level missingness diagnostics.

model_card_table	<i>Model-Card Evidence Table</i>
------------------	----------------------------------

Description

Model-Card Evidence Table

Usage

```
model_card_table(x)
```

Arguments

x A gp3bayes_model_card.

Value

The model-card evidence inventory.

model_comparison_table	<i>LOO Model-Comparison Table</i>
------------------------	-----------------------------------

Description

LOO Model-Comparison Table

Usage

```
model_comparison_table(x)
```

Arguments

x A gp3bayes_loo_comparison or matrix returned by loo::loo_compare().

Value

A data frame retaining ELPD differences and their standard errors.

<code>model_evidence_table</code>	<i>Model-Evidence Table</i>
-----------------------------------	-----------------------------

Description

Model-Evidence Table

Usage

`model_evidence_table(x)`

Arguments

`x` A `gp3bayes_model_evidence`.

Value

The model-evidence inventory table.

<code>model_weights_table</code>	<i>LOO Model-Weight Table</i>
----------------------------------	-------------------------------

Description

LOO Model-Weight Table

Usage

`model_weights_table(x)`

Arguments

`x` A `gp3bayes_loo_weights` object or named numeric weight vector.

Value

A data frame of model weights. The table does not select a model.

model_workflow_status *Summarise Workflow Stage Completion*

Description

Creates an inspectable stage map for a gp3bayes object. Stage completion is descriptive only and is not an adequacy or validity declaration.

Usage

```
model_workflow_status(x)
```

Arguments

x A gp3bayes contract, prepared object, specification, fit, or evidence object.

Value

A gp3bayes_workflow_status data frame.

plot.gp3bayes_kfold_cv
Plot Governed Exact K-Fold Results

Description

Delegates plotting to the LOO-compatible object returned by brms::kfold().

Usage

```
## S3 method for class 'gp3bayes_kfold_cv'  
plot(x, ...)
```

Arguments

x A gp3bayes_kfold_cv object.
... Arguments passed to the underlying plot method.

Value

x, invisibly.

```
plot.gp3bayes_strict_readiness_audit
```

Plot a Strict Readiness Audit

Description

Plot a Strict Readiness Audit

Usage

```
## S3 method for class 'gp3bayes_strict_readiness_audit'
plot(x, type = c("status", "condition", "duration_extremes"), ...)
```

Arguments

x	A gp3bayes_strict_readiness_audit.
type	One of "status", "condition", or "duration_extremes".
...	Additional base-graphics arguments.

Value

x, invisibly.

```
plot_advanced_pupil_simulation
```

Plot simulated advanced pupil trajectories

Description

Plot simulated advanced pupil trajectories

Usage

```
plot_advanced_pupil_simulation(x, observed = TRUE, ...)
```

Arguments

x	An advanced pupil simulation.
observed	If TRUE, plot observed condition means; otherwise plot stored latent means.
...	Additional graphical arguments.

plot_advanced_pupil_trajectory
Plot an advanced posterior pupil trajectory

Description

Plot an advanced posterior pupil trajectory

Usage

```
plot_advanced_pupil_trajectory(x, probability = 0.95, ...)
```

Arguments

x	An advanced trajectory prediction.
probability	Central interval probability.
...	Additional graphical arguments.

plot_autocorrelation *Autocorrelation Diagnostic Plot*

Description

Autocorrelation Diagnostic Plot

Usage

```
plot_autocorrelation(fit, variables = NULL, regex = "^b_", lags = 20L)
```

Arguments

fit	A fitted gp3bayes_fit.
variables	Optional posterior variables.
regex	Optional posterior-variable regular expression.
lags	Maximum autocorrelation lag.

Value

A ggplot object.

`plot_backend_environment_gg`*Plot Backend Environment Checks with ggplot2*

Description

Plot Backend Environment Checks with ggplot2

Usage

```
plot_backend_environment_gg(x)
```

Arguments

x A gp3bayes_backend_environment.

Value

A ggplot.

`plot_backend_parity_gg`*Plot Backend Parity with ggplot2*

Description

Plot Backend Parity with ggplot2

Usage

```
plot_backend_parity_gg(x)
```

Arguments

x A gp3bayes_backend_parity_audit.

Value

A ggplot.

`plot_binary_calibration`*Binary Calibration Plot*

Description

Binary Calibration Plot

Usage

```
plot_binary_calibration(x, bins = 10L)
```

Arguments

x	A binary calibration table or binary expected prediction.
bins	Number of bins if x is a prediction object.

Value

A ggplot object.

`plot_binary_calibration_uncertainty`*Plot Binary Calibration Uncertainty*

Description

Plot Binary Calibration Uncertainty

Usage

```
plot_binary_calibration_uncertainty(x)
```

Arguments

x	A binary calibration-uncertainty object or table.
---	---

Value

A ggplot.

plot_binary_group_calibration
Grouped Binary Calibration Plot

Description

Grouped Binary Calibration Plot

Usage

```
plot_binary_group_calibration(x, group = NULL)
```

Arguments

x	Grouped calibration table or binary expected prediction.
group	Grouping-column name when x is a prediction object.

Value

A ggplot.

plot_binary_precision_recall
Binary Precision-Recall Plot

Description

Binary Precision-Recall Plot

Usage

```
plot_binary_precision_recall(x, observed = NULL)
```

Arguments

x	A precision-recall table, binary expected prediction, or numeric probabilities.
observed	Optional observed outcomes for numeric probabilities.

Value

A ggplot.

plot_binary_roc	<i>Binary ROC Plot</i>
-----------------	------------------------

Description

Binary ROC Plot

Usage

```
plot_binary_roc(x, observed = NULL)
```

Arguments

x	A ROC table, binary expected prediction, or numeric probabilities.
observed	Optional observed outcomes for numeric probabilities.

Value

A ggplot.

plot_binary_threshold_metrics	<i>Binary Threshold-Metric Plot</i>
-------------------------------	-------------------------------------

Description

Binary Threshold-Metric Plot

Usage

```
plot_binary_threshold_metrics(  
  x,  
  observed = NULL,  
  thresholds = seq(0.1, 0.9, by = 0.05)  
)
```

Arguments

x	Threshold-metric table, binary expected prediction, or numeric probabilities.
observed	Optional observed outcomes for numeric predictions.
thresholds	Thresholds to evaluate when x is not already a table.

Value

A ggplot object.

`plot_binocular_pupil_trajectory`*Plot binocular posterior trajectories*

Description

Plot binocular posterior trajectories

Usage

```
plot_binocular_pupil_trajectory(x, probability = x$probability, ...)
```

Arguments

<code>x</code>	A binocular trajectory.
<code>probability</code>	Central interval probability.
<code>...</code>	Additional graphical arguments.

`plot_design_support_gg`*Plot Design-Support Components with ggplot2*

Description

Plot Design-Support Components with ggplot2

Usage

```
plot_design_support_gg(x)
```

Arguments

<code>x</code>	A <code>gp3bayes_design_support_audit</code> .
----------------	--

Value

A `ggplot`.

`plot_diagnostic_dashboard`*Plot Diagnostic Dashboard Availability*

Description

Plot Diagnostic Dashboard Availability

Usage

```
plot_diagnostic_dashboard(x)
```

Arguments

`x` A diagnostic dashboard.

Value

A ggplot.

`plot_duration_pit`*Duration PIT Plot*

Description

Duration PIT Plot

Usage

```
plot_duration_pit(x, bins = 10L)
```

Arguments

`x` A duration PIT table or duration posterior predictive object.

`bins` Number of histogram bins.

Value

A ggplot object.

<code>plot_duration_qq</code>	<i>Duration Predictive Q-Q Plot</i>
-------------------------------	-------------------------------------

Description

Duration Predictive Q-Q Plot

Usage

`plot_duration_qq(x)`

Arguments

`x` A duration Q-Q table or duration posterior predictive object.

Value

A ggplot.

<code>plot_duration_quantile_calibration</code>	<i>Duration Quantile-Calibration Plot</i>
---	---

Description

Duration Quantile-Calibration Plot

Usage

`plot_duration_quantile_calibration(x)`

Arguments

`x` A duration quantile-calibration table or duration predictive object.

Value

A ggplot object.

plot_duration_tail	<i>Duration Tail-Check Plot</i>
--------------------	---------------------------------

Description

Duration Tail-Check Plot

Usage

```
plot_duration_tail(x)
```

Arguments

x	A duration tail-check table.
---	------------------------------

Value

A ggplot.

plot_estimand_intervals	<i>Estimand Interval Plot</i>
-------------------------	-------------------------------

Description

Estimand Interval Plot

Usage

```
plot_estimand_intervals(x, quantities = NULL, probs = c(0.025, 0.5, 0.975))
```

Arguments

x	A gp3bayes_estimand.
quantities	Optional estimand quantities.
probs	Posterior interval probabilities.

Value

A ggplot object.

`plot_estimand_sensitivity_gg`*Plot Estimand Sensitivity with ggplot2*

Description

Plot Estimand Sensitivity with ggplot2

Usage

```
plot_estimand_sensitivity_gg(x)
```

Arguments

`x` An estimand-sensitivity object or its table.

Value

A ggplot.

`plot_exceedance_probability`*Prediction Exceedance-Probability Plot*

Description

Prediction Exceedance-Probability Plot

Usage

```
plot_exceedance_probability(x)
```

Arguments

`x` A table returned by `prediction_exceedance_probability()`.

Value

A ggplot object.

`plot_grouped_prediction_check`*Grouped Posterior Predictive Plot*

Description

Grouped Posterior Predictive Plot

Usage

```
plot_grouped_prediction_check(x)
```

Arguments

`x` A gp3bayes_group_prediction_check.

Value

A ggplot object.

`plot_group_deletion_sensitivity`*Plot Group-Deletion Sensitivity*

Description

Plot Group-Deletion Sensitivity

Usage

```
plot_group_deletion_sensitivity(x)
```

Arguments

`x` A group-deletion object or summary table.

Value

A ggplot.

plot_group_effects	<i>Group-Effect Plot</i>
--------------------	--------------------------

Description

Group-Effect Plot

Usage

plot_group_effects(x, groups = NULL)

Arguments

- x A group-effect table or fitted gp3bayes object.
- groups Optional grouping factors when x is a fit.

Value

A faceted ggplot object.

plot_group_effect_distribution	<i>Plot Group-Effect Posterior Distributions</i>
--------------------------------	--

Description

Plot Group-Effect Posterior Distributions

Usage

plot_group_effect_distribution(x, max_levels = 20L)

Arguments

- x A table returned by group_effect_draws_table().
- max_levels Maximum displayed grouping levels.

Value

A faceted ggplot.

plot_group_effect_rank_probability
<i>Plot Group-Effect Rank Probabilities</i>

Description

Plot Group-Effect Rank Probabilities

Usage

plot_group_effect_rank_probability(x)

Arguments

x A rank-probability table.

Value

A ggplot.

plot_group_predictions
<i>Group Prediction Plot</i>

Description

Group Prediction Plot

Usage

plot_group_predictions(x, group_column)

Arguments

x A group prediction summary.
group_column Name of the grouping column to use on the axis.

Value

A ggplot.

plot_loo_group_elpd	<i>Plot Grouped Pointwise ELPD Contribution</i>
---------------------	---

Description

Plot Grouped Pointwise ELPD Contribution

Usage

plot_loo_group_elpd(x)

Arguments

x A grouped LOO influence table.

Value

A ggplot.

plot_loo_group_influence	<i>Plot Grouped LOO Influence</i>
--------------------------	-----------------------------------

Description

Plot Grouped LOO Influence

Usage

plot_loo_group_influence(x)

Arguments

x A grouped LOO influence table.

Value

A ggplot.

plot_loo_influence	<i>LOO Influence Plot</i>
--------------------	---------------------------

Description

LOO Influence Plot

Usage

```
plot_loo_influence(x)
```

Arguments

x A gp3bayes PSIS-LOO object, loo object, or LOO diagnostic table.

Value

A ggplot object.

plot_loo_influence_rank	<i>Plot Ranked LOO Influence</i>
-------------------------	----------------------------------

Description

Plot Ranked LOO Influence

Usage

```
plot_loo_influence_rank(x)
```

Arguments

x A LOO object or pointwise table.

Value

A ggplot.

`plot_loo_pareto_vs_elpd`*Plot Pareto-k versus Pointwise ELPD*

Description

Plot Pareto-k versus Pointwise ELPD

Usage

```
plot_loo_pareto_vs_elpd(x)
```

Arguments

`x` A LOO object or pointwise table.

Value

A ggplot.

`plot_loo_pointwise_elpd`*Plot Pointwise LOO ELPD*

Description

Plot Pointwise LOO ELPD

Usage

```
plot_loo_pointwise_elpd(x)
```

Arguments

`x` A LOO object or pointwise table.

Value

A ggplot.

`plot_manifest_comparison_gg`*Plot Manifest Differences with ggplot2*

Description

Plot Manifest Differences with ggplot2

Usage

```
plot_manifest_comparison_gg(x)
```

Arguments

`x` A gp3bayes_manifest_comparison.

Value

A ggplot.

`plot_mcmc_quality`*MCMC Review-Flag Plot*

Description

MCMC Review-Flag Plot

Usage

```
plot_mcmc_quality(x)
```

Arguments

`x` A fitted gp3bayes object or gp3bayes_mcmc_quality.

Value

A ggplot showing parameter-level review flags.

plot_missingness_gg	<i>Plot Missingness Fractions with ggplot2</i>
---------------------	--

Description

Plot Missingness Fractions with ggplot2

Usage

```
plot_missingness_gg(x)
```

Arguments

x	A gp3bayes_missingness_audit.
---	-------------------------------

Value

A ggplot.

plot_model_comparison	<i>LOO Model-Comparison Plot</i>
-----------------------	----------------------------------

Description

LOO Model-Comparison Plot

Usage

```
plot_model_comparison(x)
```

Arguments

x	A gp3bayes LOO comparison or comparison table.
---	--

Value

A ggplot object.

`plot_model_evidence_gg`*Plot Model-Evidence Availability with ggplot2*

Description

Plot Model-Evidence Availability with ggplot2

Usage

```
plot_model_evidence_gg(x)
```

Arguments

`x` A gp3bayes_model_evidence.

Value

A ggplot.

`plot_model_weights` *LOO Model-Weight Plot*

Description

LOO Model-Weight Plot

Usage

```
plot_model_weights(x)
```

Arguments

`x` A gp3bayes LOO weight object or weight table.

Value

A ggplot object.

plot_posterior_areas *Posterior Area Plot*

Description

Posterior Area Plot

Usage

```
plot_posterior_areas(
  x,
  variables = NULL,
  regex = NULL,
  prob = 0.5,
  prob_outer = 0.95
)
```

Arguments

x	A gp3bayes fit or posterior draws accepted by posterior_interval_table().
variables, regex	Posterior variable selectors.
prob	Inner interval probability.
prob_outer	Outer interval probability.

Value

A ggplot object.

plot_posterior_correlations
 Posterior Correlation Plot

Description

Posterior Correlation Plot

Usage

```
plot_posterior_correlations(
  x,
  variables = NULL,
  regex = NULL,
  method = c("pearson", "spearman")
)
```


Arguments

x	A gp3bayes fit or posterior draws.
variables, regex	Posterior variable selectors.
method	Correlation method.

Value

A ggplot heatmap of posterior-draw correlations.

`plot_posterior_density`*Posterior Density Plot*

Description

Posterior Density Plot

Usage

```
plot_posterior_density(x, variables = NULL, regex = NULL)
```

Arguments

x	A gp3bayes fit or posterior draws.
variables, regex	Posterior variable selectors.

Value

A ggplot object.

`plot_posterior_intervals`*Posterior Interval Plot*

Description

Posterior Interval Plot

Usage

```
plot_posterior_intervals(
  x,
  variables = NULL,
  regex = NULL,
  prob = 0.8,
  prob_outer = 0.95
)
```

Arguments

x	A gp3bayes fit or posterior draws accepted by posterior_interval_table().
variables, regex	Posterior variable selectors.
prob	Inner interval probability.
prob_outer	Outer interval probability.

Value

A ggplot object.

plot_posterior_pairs *Posterior Pairs Plot*

Description

Posterior Pairs Plot

Usage

```
plot_posterior_pairs(fit, variables = NULL, regex = "^b_", max_variables = 8L)
```

Arguments

fit	A fitted gp3bayes_fit.
variables	Optional variables.
regex	Optional variable regular expression.
max_variables	Maximum number of variables displayed.

Value

A bayesplot pairs object.

`plot_powerscale_sensitivity_gg`*Plot Power-Scale Sensitivity with ggplot2*

Description

Plot Power-Scale Sensitivity with ggplot2

Usage

```
plot_powerscale_sensitivity_gg(x)
```

Arguments

`x` A power-scale sensitivity object or its table.

Value

A ggplot.

`plot_ppc_statistic`*Posterior Predictive Statistic Plot*

Description

Posterior Predictive Statistic Plot

Usage

```
plot_ppc_statistic(x, bins = 30L)
```

Arguments

`x` A gp3bayes_ppc_statistic.

`bins` Histogram bins.

Value

A ggplot.

`plot_prediction_contrast_profile`*Plot a Prediction Contrast Profile*

Description

Plot a Prediction Contrast Profile

Usage

```
plot_prediction_contrast_profile(x)
```

Arguments

`x` A prediction contrast profile.

Value

A ggplot.

`plot_prediction_draws` *Prediction-Draw Distribution Plot*

Description

Prediction-Draw Distribution Plot

Usage

```
plot_prediction_draws(x, observations = NULL, max_draws = 500L)
```

Arguments

`x` A gp3bayes_prediction.
`observations` Optional prediction-row indices.
`max_draws` Maximum posterior draws displayed.

Value

A ggplot.

`plot_prediction_gradient`*Plot Prediction-Profile Gradients*

Description

Plot Prediction-Profile Gradients

Usage

```
plot_prediction_gradient(x)
```

Arguments

`x` A prediction profile or gradient table.

Value

A ggplot.

`plot_prediction_intervals`*Prediction-Interval Plot*

Description

Prediction-Interval Plot

Usage

```
plot_prediction_intervals(x, max_rows = 100L)
```

Arguments

`x` A gp3bayes_prediction.

`max_rows` Maximum rows displayed.

Value

A ggplot object.

plot_prediction_interval_width
Prediction Interval-Width Plot

Description

Prediction Interval-Width Plot

Usage

```
plot_prediction_interval_width(x)
```

Arguments

x A prediction object or interval-width table.

Value

A ggplot.

plot_prediction_profile
Plot a Prediction Profile

Description

Plot a Prediction Profile

Usage

```
plot_prediction_profile(x)
```

Arguments

x A prediction profile.

Value

A ggplot.

plot_prediction_rank_probabilities
Prediction Rank-Probability Plot

Description

Prediction Rank-Probability Plot

Usage

```
plot_prediction_rank_probabilities(x)
```

Arguments

x A ranking-probability table.

Value

A ggplot.

plot_prediction_score_uncertainty
Plot Prediction Score Uncertainty

Description

Plot Prediction Score Uncertainty

Usage

```
plot_prediction_score_uncertainty(x)
```

Arguments

x A score-uncertainty object.

Value

A faceted ggplot.

`plot_prediction_support`*Prediction-Support Plot*

Description

Prediction-Support Plot

Usage

```
plot_prediction_support(x)
```

Arguments

`x` A prediction-support audit or its table.

Value

A ggplot object.

`plot_prediction_surface`*Plot a Prediction Surface*

Description

Plot a Prediction Surface

Usage

```
plot_prediction_surface(x)
```

Arguments

`x` A prediction surface.

Value

A ggplot.

`plot_prediction_surface_uncertainty`*Plot Prediction-Surface Uncertainty*

Description

Plot Prediction-Surface Uncertainty

Usage

```
plot_prediction_surface_uncertainty(x)
```

Arguments

`x` A prediction surface.

Value

A ggplot.

`plot_predictive_atlas_statistics`*Plot Predictive Atlas Statistics*

Description

Plot Predictive Atlas Statistics

Usage

```
plot_predictive_atlas_statistics(x)
```

Arguments

`x` A predictive distribution atlas.

Value

A faceted ggplot.

`plot_predictive_coverage`*Predictive-Coverage Plot*

Description

Predictive-Coverage Plot

Usage

```
plot_predictive_coverage(x)
```

Arguments

`x` Predictive coverage table or posterior predictive object.

Value

A ggplot object.

`plot_predictive_quantile_envelope`*Plot Posterior-Predictive Quantile Envelope*

Description

Plot Posterior-Predictive Quantile Envelope

Usage

```
plot_predictive_quantile_envelope(x)
```

Arguments

`x` A quantile-envelope table.

Value

A ggplot.

`plot_predictive_residuals`*Predictive-Residual Plot*

Description

Predictive-Residual Plot

Usage

```
plot_predictive_residuals(x)
```

Arguments

`x` Residual table returned by `predictive_residuals()`.

Value

A ggplot object.

`plot_prior_posterior_contraction`*Plot Prior-to-Posterior Contraction*

Description

Plot Prior-to-Posterior Contraction

Usage

```
plot_prior_posterior_contraction(x)
```

Arguments

`x` A prior-posterior bridge.

Value

A ggplot.

`plot_prior_posterior_density`*Plot Declared Prior and Posterior Densities*

Description

Plot Declared Prior and Posterior Densities

Usage

```
plot_prior_posterior_density(x, max_draws = 1000L)
```

Arguments

<code>x</code>	A prior-posterior bridge.
<code>max_draws</code>	Maximum draws displayed per distribution.

Value

A faceted ggplot.

`plot_prior_posterior_intervals`*Plot Prior and Posterior Intervals*

Description

Plot Prior and Posterior Intervals

Usage

```
plot_prior_posterior_intervals(x)
```

Arguments

<code>x</code>	A prior-posterior bridge.
----------------	---------------------------

Value

A faceted ggplot.

`plot_prior_posterior_shift`*Plot Prior-to-Posterior Location Shift*

Description

Plot Prior-to-Posterior Location Shift

Usage

```
plot_prior_posterior_shift(x)
```

Arguments

`x` A prior-posterior bridge.

Value

A ggplot.

`plot_prior_sensitivity`*Plot Prior-Scale Sensitivity*

Description

Plot Prior-Scale Sensitivity

Usage

```
plot_prior_sensitivity(x)
```

Arguments

`x` A prior-sensitivity object or comparison table.

Value

A faceted ggplot.

`plot_prior_sensitivity_scenarios`*Plot Prior-Sensitivity Scenario Maxima*

Description

Plot Prior-Sensitivity Scenario Maxima

Usage

```
plot_prior_sensitivity_scenarios(x)
```

Arguments

`x` A prior-sensitivity object or scenario table.

Value

A ggplot.

`plot_pupil_autocorrelation_comparison`*Plot residual autocorrelation comparison*

Description

Plot residual autocorrelation comparison

Usage

```
plot_pupil_autocorrelation_comparison(x, absolute = TRUE, ...)
```

Arguments

`x` An autocorrelation comparison object.
`absolute` Plot median absolute ACF if TRUE.
`...` Additional graphical arguments.

plot_pupil_dynamic_contrast
Plot a dynamic pupil condition contrast

Description

Plot a dynamic pupil condition contrast

Usage

```
plot_pupil_dynamic_contrast(x, ...)
```

Arguments

x	A dynamic contrast.
...	Additional base graphics arguments.

plot_pupil_estimand *Plot declared-window pupil estimands*

Description

Plot declared-window pupil estimands

Usage

```
plot_pupil_estimand(x)
```

Arguments

x	Pupil estimand object.
---	------------------------

Value

A ggplot object.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

<code>plot_pupil_gp_hyperparameters</code>
<i>Plot Gaussian-process hyperparameters</i>

Description

Plot Gaussian-process hyperparameters

Usage

`plot_pupil_gp_hyperparameters(x, ...)`

Arguments

- | | |
|------------------|---------------------------------|
| <code>x</code> | A GP-hyperparameter object. |
| <code>...</code> | Additional graphical arguments. |

<code>plot_pupil_identifiability_audit</code>
<i>Plot advanced identifiability/design-support audit</i>

Description

Plot advanced identifiability/design-support audit

Usage

`plot_pupil_identifiability_audit(x, ...)`

Arguments

- | | |
|------------------|---------------------------------|
| <code>x</code> | An identifiability audit. |
| <code>...</code> | Additional graphical arguments. |

plot_pupil_lfo	<i>Plot leave-future-out scores</i>
----------------	-------------------------------------

Description

Plot leave-future-out scores

Usage

```
plot_pupil_lfo(x, ...)
```

Arguments

x	An executed LFO validation or LFO comparison.
...	Additional graphical arguments.

plot_pupil_measurement_audit	<i>Plot gaze/PFE and luminance measurement-context evidence</i>
------------------------------	---

Description

Plot gaze/PFE and luminance measurement-context evidence

Usage

```
plot_pupil_measurement_audit(x)
```

Arguments

x	A pupil measurement audit.
---	----------------------------

Value

A ggplot object.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

<code>plot_pupil_measurement_uncertainty</code>
<i>Plot measurement-uncertainty audit</i>

Description

Plot measurement-uncertainty audit

Usage

`plot_pupil_measurement_uncertainty(x, ...)`

Arguments

- | | |
|------------------|---------------------------------|
| <code>x</code> | A measurement audit. |
| <code>...</code> | Additional graphical arguments. |

<code>plot_pupil_missingness</code>
<i>Plot missing-response fraction over time</i>

Description

Plot missing-response fraction over time

Usage

`plot_pupil_missingness(x, ...)`

Arguments

- | | |
|------------------|---------------------------------|
| <code>x</code> | A missingness audit. |
| <code>...</code> | Additional graphical arguments. |

plot_pupil_model_comparison
Plot predictive model comparison

Description

Plot predictive model comparison

Usage

```
plot_pupil_model_comparison(x, ...)
```

Arguments

x	A model-comparison object.
...	Additional graphical arguments.

plot_pupil_model_complexity
Plot advanced model computational complexity

Description

Plot advanced model computational complexity

Usage

```
plot_pupil_model_complexity(x, ...)
```

Arguments

x	A complexity audit or advanced specification.
...	Additional graphical arguments.

plot_pupil_observed_trajectory
Plot observed pupil trajectories

Description

Plot observed pupil trajectories

Usage

```
plot_pupil_observed_trajectory(x, summary = TRUE)
```

Arguments

x	Prepared pupil object.
summary	Whether to plot condition means rather than individual participant-trial traces.

Value

A ggplot object.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

plot_pupil_posterior_trajectory
Plot posterior pupil trajectories or condition differences

Description

Plot posterior pupil trajectories or condition differences

Usage

```
plot_pupil_posterior_trajectory(x)
```

Arguments

x	Pupil trajectory or condition-contrast object.
---	--

Value

A ggplot object.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

plot_pupil_ppc	<i>Plot pupil posterior-predictive evidence</i>
----------------	---

Description

Uses one consolidated plotting interface for trajectory, residual, feature, autocorrelation, heterogeneity, or measurement-context PPC evidence.

Usage

```
plot_pupil_ppc(  
  x,  
  component = c("trajectory", "residuals", "features", "autocorrelation",  
                "heterogeneity", "measurement_context")  
)
```

Arguments

x	Pupil PPC object.
component	Evidence component to plot.

Value

A ggplot object.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

plot_pupil_predictive_calibration	<i>Plot posterior predictive calibration</i>
-----------------------------------	--

Description

Plot posterior predictive calibration

Usage

```
plot_pupil_predictive_calibration(x, ...)
```

Arguments

x	A predictive score/calibration object.
...	Additional graphical arguments.

plot_pupil_readiness *Plot pupil readiness evidence*

Description

Plot pupil readiness evidence

Usage

```
plot_pupil_readiness(x)
```

Arguments

x Pupil readiness audit.

Value

A ggplot object.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

plot_pupil_residual_acf
 Plot pupil residual autocorrelation

Description

Plot pupil residual autocorrelation

Usage

```
plot_pupil_residual_acf(x)
```

Arguments

x Pupil diagnostics object or residual ACF table.

Value

A ggplot object.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

`plot_pupil_residual_scale`*Plot posterior residual scale over time*

Description

Plot posterior residual scale over time

Usage

```
plot_pupil_residual_scale(x, ...)
```

Arguments

<code>x</code>	A residual-scale estimand.
<code>...</code>	Additional graphical arguments.

`plot_pupil_residual_spectrum`*Plot residual spectrum*

Description

Plot residual spectrum

Usage

```
plot_pupil_residual_spectrum(x, ...)
```

Arguments

<code>x</code>	A residual-spectrum object.
<code>...</code>	Additional graphical arguments.

`plot_pupil_response_parameters`*Plot experimental nonlinear response parameters*

Description

Plot experimental nonlinear response parameters

Usage

```
plot_pupil_response_parameters(x, ...)
```

Arguments

<code>x</code>	A response-parameter object.
<code>...</code>	Additional graphical arguments.

`plot_pupil_sensitivity`*Plot pupil sensitivity scenarios or result comparisons*

Description

Plot pupil sensitivity scenarios or result comparisons

Usage

```
plot_pupil_sensitivity(x)
```

Arguments

<code>x</code>	Pupil sensitivity suite or comparison.
----------------	--

Value

A ggplot object.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

plot_pupil_temporal_dependence
Plot empirical temporal-dependence audit

Description

Plot empirical temporal-dependence audit

Usage

```
plot_pupil_temporal_dependence(x, ...)
```

Arguments

x	A temporal-dependence audit.
...	Additional graphical arguments.

plot_pupil_trajectory_derivative
Plot posterior pupil trajectory derivative

Description

Plot posterior pupil trajectory derivative

Usage

```
plot_pupil_trajectory_derivative(x, probability = x$probability, ...)
```

Arguments

x	A trajectory-derivative object.
probability	Central posterior interval probability.
...	Additional graphical arguments.

<code>plot_pupil_validation</code>	<i>Plot pupil validation evidence</i>
------------------------------------	---------------------------------------

Description

Plot pupil validation evidence

Usage

`plot_pupil_validation(x)`

Arguments

`x` Pupil validation object.

Value

A ggplot object.

Examples

`# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.`

<code>plot_random_intercept_variance_partition</code>	<i>Plot Random-Intercept Variance Partition</i>
---	---

Description

Plot Random-Intercept Variance Partition

Usage

`plot_random_intercept_variance_partition(x)`

Arguments

`x` A variance-partition object or table.

Value

A ggplot.

plot_random_slope_sensitivity
Plot Random-Slope Sensitivity

Description

Plot Random-Slope Sensitivity

Usage

```
plot_random_slope_sensitivity(x)
```

Arguments

x A random-slope sensitivity object or estimand-sensitivity table.

Value

A ggplot.

plot_rank_diagnostics *Rank-Diagnostic Plot*

Description

Rank-Diagnostic Plot

Usage

```
plot_rank_diagnostics(fit, variables = NULL, regex = "^b_")
```

Arguments

fit A fitted gp3bayes_fit.
variables Optional posterior variables.
regex Optional posterior-variable regular expression.

Value

A ggplot rank-overlay diagnostic.

<code>plot_recovery_bias</code>	<i>Plot Recovery Standardized Bias</i>
---------------------------------	--

Description

Plot Recovery Standardized Bias

Usage

`plot_recovery_bias(x)`

Arguments

`x` A recovery object or recovery parameter table.

Value

A ggplot.

<code>plot_recovery_coverage</code>	<i>Plot Recovery Coverage</i>
-------------------------------------	-------------------------------

Description

Plot Recovery Coverage

Usage

`plot_recovery_coverage(x)`

Arguments

`x` A recovery object or recovery parameter table.

Value

A ggplot.

`plot_recovery_estimates`*Plot Repetition-Level Recovery Estimates*

Description

Plot Repetition-Level Recovery Estimates

Usage

```
plot_recovery_estimates(x, variables = NULL)
```

Arguments

<code>x</code>	A recovery object or repetition-level estimate table.
<code>variables</code>	Optional variables to retain.

Value

A faceted ggplot.

`plot_recovery_fit_status`*Plot Recovery Fit Status*

Description

Plot Recovery Fit Status

Usage

```
plot_recovery_fit_status(x)
```

Arguments

<code>x</code>	A recovery object or fit-status table.
----------------	--

Value

A ggplot.

<code>plot_recovery_rmse</code>	<i>Plot Recovery RMSE</i>
---------------------------------	---------------------------

Description

Plot Recovery RMSE

Usage

`plot_recovery_rmse(x)`

Arguments

`x` A recovery object or recovery parameter table.

Value

A ggplot.

<code>plot_reporting_checklist</code>	<i>Plot a Reporting Checklist</i>
---------------------------------------	-----------------------------------

Description

Plot a Reporting Checklist

Usage

`plot_reporting_checklist(x)`

Arguments

`x` A reporting checklist, fit, or model card.

Value

A ggplot.

`plot_sampler_diagnostics`*Sampler-Diagnostic Plot*

Description

Sampler-Diagnostic Plot

Usage

```
plot_sampler_diagnostics(fit)
```

Arguments

<code>fit</code>	A fitted <code>gp3bayes_fit</code> .
------------------	--------------------------------------

Value

A ggplot of sampler metrics relative to review thresholds.

`plot_sampling_diagnostics`*Plot Sampling Diagnostics*

Description

Produces trace, energy, treedepth, or divergence plots for an approved fitted `gp3bayes` model.

Usage

```
plot_sampling_diagnostics(  
  fit,  
  type = c("trace", "energy", "treedepth", "divergence"),  
  variables = NULL  
)
```

Arguments

<code>fit</code>	A fitted <code>gp3bayes_fit</code> .
<code>type</code>	One of "trace", "energy", "treedepth", or "divergence".
<code>variables</code>	Optional posterior parameter names used for trace plots.

Details

Diagnostic plots support interpretation of sampling behaviour. They do not establish convergence or substantive model adequacy by themselves.

Value

A plot object created by bayesplot.

<code>plot_sbc_coverage_gg</code>	<i>SBC Coverage Plot</i>
-----------------------------------	--------------------------

Description

SBC Coverage Plot

Usage

```
plot_sbc_coverage_gg(x, variables = NULL, ...)
```

Arguments

- `x` A `gp3bayes_sbc_result`.
- `variables` Optional variables.
- `...` Arguments passed to SBC.

Value

A plot object returned by SBC.

<code>plot_sbc_ecdf_gg</code>	<i>SBC ECDF-Difference Plot</i>
-------------------------------	---------------------------------

Description

SBC ECDF-Difference Plot

Usage

```
plot_sbc_ecdf_gg(x, variables = NULL, ...)
```

Arguments

- `x` A `gp3bayes_sbc_result`.
- `variables` Optional variables.
- `...` Arguments passed to SBC.

Value

A plot object returned by SBC.

plot_sbc_rank_gg	<i>SBC Rank-Histogram Plot</i>
------------------	--------------------------------

Description

SBC Rank-Histogram Plot

Usage

```
plot_sbc_rank_gg(x, variables = NULL, ...)
```

Arguments

x	A gp3bayes_sbc_result.
variables	Optional variables.
...	Arguments passed to SBC.

Value

A plot object returned by SBC.

plot_sbc_simulated_vs_estimated_gg	<i>SBC Simulated-versus-Estimated Plot</i>
------------------------------------	--

Description

SBC Simulated-versus-Estimated Plot

Usage

```
plot_sbc_simulated_vs_estimated_gg(x, variables = NULL, ...)
```

Arguments

x	A gp3bayes_sbc_result.
variables	Optional variables.
...	Arguments passed to SBC.

Value

A plot object returned by SBC.

`plot_schema_comparison_gg`*Plot Schema Differences with ggplot2*

Description

Plot Schema Differences with ggplot2

Usage

```
plot_schema_comparison_gg(x)
```

Arguments

x A gp3bayes_schema_comparison.

Value

A ggplot.

`plot_sensitivity_suite_gg`*Plot a Sensitivity Suite with ggplot2*

Description

Plot a Sensitivity Suite with ggplot2

Usage

```
plot_sensitivity_suite_gg(x)
```

Arguments

x A gp3bayes_sensitivity_suite.

Value

A ggplot.

plot_uncertainty_decomposition
Prediction-Uncertainty Plot

Description

Prediction-Uncertainty Plot

Usage

```
plot_uncertainty_decomposition(x, max_rows = 100L)
```

Arguments

x	A gp3bayes_prediction_uncertainty.
max_rows	Maximum observations displayed.

Value

A ggplot object.

plot_variance_components
Variance-Component Plot

Description

Variance-Component Plot

Usage

```
plot_variance_components(x)
```

Arguments

x	A variance-component table or fitted gp3bayes object.
---	---

Value

A ggplot object.

posterior_correlation_table
Posterior Correlation Table

Description

Posterior Correlation Table

Usage

```
posterior_correlation_table(
  x,
  variables = NULL,
  regex = NULL,
  method = c("pearson", "spearman")
)
```

Arguments

x	A gp3bayes fit or posterior draws accepted by posterior_interval_table().
variables, regex	Posterior variable selectors.
method	Correlation method.

Value

A long data frame of unique posterior-draw correlations.

Examples

```
x <- cbind(a = rnorm(100), b = rnorm(100), c = rnorm(100))
posterior_correlation_table(x)
```

posterior_interval_table
Posterior Interval Table

Description

Posterior Interval Table

Usage

```
posterior_interval_table(
  x,
  variables = NULL,
  regex = NULL,
  probs = c(0.025, 0.5, 0.975)
)
```

Arguments

x	A gp3bayes fit, posterior draws object, numeric matrix, or numeric data frame.
variables	Optional exact posterior variable names.
regex	Optional regular expression for posterior variable names.
probs	Three probabilities defining lower, median, and upper summaries.

Value

A data frame containing posterior location, spread, and intervals.

Examples

```
draws <- cbind(alpha = rnorm(200), beta = rnorm(200, 0.5))
posterior_interval_table(draws)
```

posterior_predictive_statistic

Posterior Predictive Statistic Check

Description

Computes one scalar discrepancy statistic for every posterior predictive draw and compares that distribution with the same statistic in the observed data. The returned probability is descriptive and is not an automatic model verdict.

Usage

```
posterior_predictive_statistic(
  x,
  statistic = c("mean", "sd", "median", "q90", "q95", "max", "tail_rate"),
  threshold = NULL
)
```

Arguments

x	A posterior predictive gp3bayes_prediction.
statistic	Built-in statistic: "mean", "sd", "median", "q90", "q95", "max", or "tail_rate".
threshold	Required for "tail_rate".

Value

A gp3bayes_ppc_statistic object.

posterior_predictive_summary_table
<i>Posterior Predictive Summary Table</i>

Description

Posterior Predictive Summary Table

Usage

```
posterior_predictive_summary_table(x, probs = c(0.025, 0.5, 0.975))
```

Arguments

x	A posterior predictive gp3bayes_prediction.
probs	Three summary probabilities.

Value

Observation-level predictive summaries.

posterior_probability_table
<i>Posterior Direction and ROPE Probability Table</i>

Description

Posterior Direction and ROPE Probability Table

Usage

```
posterior_probability_table(x, variables = NULL, regex = NULL, rope = NULL)
```

Arguments

x	A gp3bayes fit or posterior draws accepted by posterior_interval_table().
variables, regex	Posterior variable selectors.
rope	Optional two-element interval defining a region of practical equivalence. It is descriptive only.

Value

A data frame with posterior direction probabilities and, when requested, the posterior probability inside the supplied interval.

Examples

```
draws <- cbind(alpha = rnorm(500), beta = rnorm(500, 0.4))
posterior_probability_table(draws, rope = c(-0.1, 0.1))
```

```
powerscale_sensitivity_table
```

Power-Scale Sensitivity Table

Description

Power-Scale Sensitivity Table

Usage

```
powerscale_sensitivity_table(x)
```

Arguments

x A gp3bayes_powerscale_sensitivity.

Value

The tabular representation provided by priorsense.

```
powerscale_sequence_for_fit
```

Create a Power-Scaling Sequence

Description

Create a Power-Scaling Sequence

Usage

```
powerscale_sequence_for_fit(
  fit,
  variable = NULL,
  prior_selection = NULL,
  likelihood_selection = NULL,
  component = "both"
)
```

Arguments

- `fit` A gp3bayes fit or brmsfit.
- `variable` Optional posterior variables to inspect.
- `prior_selection` Optional tagged priors to perturb.
- `likelihood_selection` Optional likelihood subset.
- `component` Either prior, likelihood, or both, as supported by priorsense.

Value

A priorsense powerscaled sequence.

<code>ppc_statistic_table</code>	<i>Posterior Predictive Statistic Table</i>
----------------------------------	---

Description

Posterior Predictive Statistic Table

Usage

`ppc_statistic_table(x)`

Arguments

- `x` A gp3bayes_ppc_statistic.

Value

A one-row data frame.

<code>prediction_contrast</code>	<i>Posterior Prediction Contrast</i>
----------------------------------	--------------------------------------

Description

Posterior Prediction Contrast

Usage

```
prediction_contrast(  
  x,  
  row1,  
  row2,  
  measure = c("difference", "ratio", "odds_ratio"),  
  probs = c(0.025, 0.5, 0.975)  
)
```

Arguments

x	A gp3bayes_prediction.
row1, row2	Two prediction rows to compare.
measure	"difference", "ratio", or "odds_ratio".
probs	Posterior interval probabilities.

Value

A one-row posterior contrast summary.

prediction_contrast_profile_table
<i>Prediction Contrast Profile Table</i>

Description

Prediction Contrast Profile Table

Usage

```
prediction_contrast_profile_table(x)
```

Arguments

x	A prediction contrast profile.
---	--------------------------------

Value

Contrast summaries.

prediction_draws_long *Convert Prediction Draws to Long Form*

Description

Convert Prediction Draws to Long Form

Usage

```
prediction_draws_long(x, max_draws = NULL, seed = 1L)
```

Arguments

x	A gp3bayes_prediction.
max_draws	Optional maximum number of posterior draws retained.
seed	Non-negative integer used only if draws are subsampled.

Value

A long data frame with draw, observation, and predicted value.

prediction_exceedance_probability
Posterior Exceedance Probabilities

Description

Posterior Exceedance Probabilities

Usage

```
prediction_exceedance_probability(
  x,
  threshold,
  direction = c("above", "below")
)
```

Arguments

x	A gp3bayes_prediction.
threshold	Finite response-scale threshold.
direction	Whether to evaluate values above or below the threshold.

Value

Observation-level posterior exceedance probabilities.

prediction_gradient_table
<i>Prediction-Profile Gradient Table</i>

Description

Prediction-Profile Gradient Table

Usage

```
prediction_gradient_table(x, probs = c(0.025, 0.5, 0.975))
```

Arguments

- x A prediction profile.
- probs Three interval probabilities.

Value

Finite-difference posterior predictive gradients.

prediction_interval_width
<i>Prediction Interval Width Table</i>

Description

Prediction Interval Width Table

Usage

```
prediction_interval_width(x)
```

Arguments

- x A gp3bayes_prediction.

Value

A table of posterior interval width by prediction row.

```
prediction_pairwise_contrasts
```

Pairwise Prediction Contrasts

Description

Pairwise Prediction Contrasts

Usage

```
prediction_pairwise_contrasts(
  x,
  rows = NULL,
  measure = c("difference", "ratio"),
  max_rows = 20L,
  probs = c(0.025, 0.5, 0.975)
)
```

Arguments

<code>x</code>	A gp3bayes_prediction.
<code>rows</code>	Optional prediction-row indices. At most <code>max_rows</code> rows may be compared.
<code>measure</code>	Difference or ratio.
<code>max_rows</code>	Maximum number of prediction rows allowed.
<code>probs</code>	Three posterior interval probabilities.

Value

A data frame containing every unique pairwise contrast.

```
prediction_profile_table
```

Prediction Profile Table

Description

Prediction Profile Table

Usage

```
prediction_profile_table(x)
```

Arguments

<code>x</code>	A prediction profile.
----------------	-----------------------

Value

Profile summaries.

prediction_rank_probabilities
<i>Posterior Ranking Probabilities for Prediction Rows</i>

Description

Summarises relative ordering among a small, explicitly supplied set of prediction rows. No row is automatically selected or declared superior.

Usage

```
prediction_rank_probabilities(  
  x,  
  rows = NULL,  
  direction = c("higher", "lower"),  
  max_rows = 20L  
)
```

Arguments

- | | |
|-----------|--|
| x | A gp3bayes_prediction. |
| rows | Optional prediction rows. |
| direction | Whether larger or smaller values receive rank 1. |
| max_rows | Maximum rows that may be ranked. |

Value

A descriptive ranking-probability table.

prediction_score_uncertainty
<i>Posterior Uncertainty in Prediction Scores</i>

Description

Binary fits use Brier and logarithmic scores; duration fits use RMSE and MAE.

Usage

```
prediction_score_uncertainty(
  fit,
  newdata = NULL,
  include_group_effects = FALSE,
  ndraws = 1000L,
  probs = c(0.025, 0.5, 0.975)
)
```

Arguments

<code>fit</code>	A fitted <code>gp3bayes_fit</code> .
<code>newdata</code>	Optional data containing observed outcomes.
<code>include_group_effects</code>	Whether fitted group effects are included.
<code>ndraws</code>	Expected-response posterior draws.
<code>probs</code>	Three interval probabilities.

Value

A `gp3bayes_prediction_score_uncertainty`.

`prediction_score_uncertainty_table`

Prediction Score-Uncertainty Table

Description

Prediction Score-Uncertainty Table

Usage

```
prediction_score_uncertainty_table(x)
```

Arguments

<code>x</code>	A prediction-score uncertainty object.
----------------	--

Value

Metric summaries.

prediction_support_table
Prediction-Support Table

Description

Prediction-Support Table

Usage

prediction_support_table(x)

Arguments

x A gp3bayes_prediction_support object.

Value

The underlying support audit table.

prediction_surface_table
Prediction Surface Table

Description

Prediction Surface Table

Usage

prediction_surface_table(x)

Arguments

x A prediction surface.

Value

Surface summaries.

prediction_table	<i>Prediction Summary Table</i>
------------------	---------------------------------

Description

Prediction Summary Table

Usage

```
prediction_table(x)
```

Arguments

x A gp3bayes_prediction.

Value

The observation-level posterior prediction summary.

prediction_uncertainty_decomposition	<i>Decompose Prediction Uncertainty Descriptively</i>
--------------------------------------	---

Description

Separates variability in conditional expected-response draws from total posterior predictive variability. The difference is a descriptive Monte Carlo decomposition and is not a causal variance decomposition.

Usage

```
prediction_uncertainty_decomposition(
  fit,
  newdata = NULL,
  include_group_effects = FALSE,
  allow_new_levels = FALSE,
  ndraws = 1000L,
  seed = 1L
)
```


Arguments

fit	A fitted gp3bayes_fit.
newdata	Optional data frame. NULL uses the fitted prepared data.
include_group_effects	Whether fitted group-level effects are included.
allow_new_levels	Whether new grouping levels are permitted by brms.
ndraws	Optional number of posterior draws.
seed	Non-negative seed used for posterior predictive simulation.

Value

A gp3bayes_prediction_uncertainty object.

predictive_coverage_table

Posterior Predictive Coverage Table

Description

Posterior Predictive Coverage Table

Usage

```
predictive_coverage_table(x, levels = c(0.5, 0.8, 0.9, 0.95))
```

Arguments

x	A posterior predictive gp3bayes_prediction.
levels	Central predictive interval levels.

Value

A table of empirical coverage and mean interval width.

`predictive_distribution_atlas_table`*Predictive Distribution Atlas Table*

Description

Predictive Distribution Atlas Table

Usage

```
predictive_distribution_atlas_table(x)
```

Arguments

`x` A predictive distribution atlas.

Value

Draw-level distribution summaries.

`predictive_quantile_envelope`*Posterior-Predictive Quantile Envelope*

Description

Posterior-Predictive Quantile Envelope

Usage

```
predictive_quantile_envelope(  
  x,  
  probabilities = seq(0.05, 0.95, by = 0.05),  
  probs = c(0.025, 0.5, 0.975),  
  ndraws = 500L,  
  include_group_effects = TRUE,  
  seed = 1L  
)
```

Arguments

x	A fitted model or predictive atlas.
probabilities	Outcome quantile probabilities.
probs	Posterior interval probabilities for each replicated quantile.
ndraws	Predictive draws when x is a fit.
include_group_effects	Whether fitted group effects are included.
seed	Predictive seed.

Value

A quantile-envelope table.

predictive_residuals	<i>Posterior Predictive Residuals</i>
----------------------	---------------------------------------

Description

Posterior Predictive Residuals

Usage

```
predictive_residuals(fit, type = NULL, ndraws = 1000L)
```

Arguments

fit	A fitted gp3bayes_fit.
type	Residual type. Binary models support "raw" and "pearson"; duration models support "raw", "log", and "relative".
ndraws	Optional number of expected-response draws.

Value

A data frame of observed values, posterior expected values, and descriptive residuals.

```
predict_advanced_pupil_trajectory
```

Predict an advanced pupil trajectory

Description

Predict an advanced pupil trajectory

Usage

```
predict_advanced_pupil_trajectory(
  fit,
  newdata = NULL,
  type = c("expected", "posterior_predictive", "linear"),
  ndraws = 500L,
  population_only = TRUE,
  allow_new_levels = FALSE,
  max_grid = 5000L
)
```

Arguments

<code>fit</code>	A fitted advanced pupil model.
<code>newdata</code>	Optional prediction data. If omitted, a population-level time-by-condition grid is generated with covariates held at reference values.
<code>type</code>	"expected", "posterior_predictive", or "linear".
<code>ndraws</code>	Number of posterior draws.
<code>population_only</code>	Exclude group-level effects when TRUE.
<code>allow_new_levels</code>	Passed to brms prediction methods.
<code>max_grid</code>	Maximum generated grid size.

Value

A gp3bayes_pupil_advanced_trajectory object.

predict_binary_probability
Binary Event-Probability Predictions

Description

Binary Event-Probability Predictions

Usage

```
predict_binary_probability(  
  fit,  
  newdata = NULL,  
  include_group_effects = FALSE,  
  allow_new_levels = FALSE,  
  ndraws = NULL,  
  probs = c(0.025, 0.5, 0.975)  
)
```

Arguments

fit	A fitted gp3bayes_fit.
newdata	Optional data frame. NULL uses the fitted prepared data.
include_group_effects	Whether fitted group-level effects are included.
allow_new_levels	Whether new grouping levels are permitted by brms.
ndraws	Optional number of posterior draws.
probs	Three probabilities used to summarise predictions.

Value

A gp3bayes_prediction of event probabilities.

predict_duration *Duration Predictions*

Description

Duration Predictions

Usage

```
predict_duration(  
  fit,  
  newdata = NULL,  
  type = c("median", "expected", "predictive"),  
  include_group_effects = FALSE,  
  allow_new_levels = FALSE,  
  ndraws = NULL,  
  probs = c(0.025, 0.5, 0.975),  
  seed = 1L  
)
```

Arguments

- fit A fitted gp3bayes_fit.
- newdata Optional data frame. NULL uses the fitted prepared data.
- type One of "median", "expected", or "predictive".
- include_group_effects Whether fitted group-level effects are included.
- allow_new_levels Whether new grouping levels are permitted by brms.
- ndraws Optional number of posterior draws.
- probs Three probabilities used to summarise predictions.
- seed Non-negative seed used for posterior predictive simulation.

Value

A gp3bayes_prediction on the recorded duration scale.

predict_model	<i>Posterior Prediction for Approved gp3bayes Models</i>
---------------	--

Description

Distinguishes conditional expectations, new-outcome posterior predictions, linear-predictor draws, and the conditional median for lognormal duration models.

Usage

```
predict_model(  
  fit,  
  newdata = NULL,  
  type = c("expected", "predictive", "linear", "median"),  
  include_group_effects = FALSE,  
  allow_new_levels = FALSE,
```

```

    ndraws = NULL,
    probs = c(0.025, 0.5, 0.975),
    seed = 1L
  )

```

Arguments

<code>fit</code>	A fitted <code>gp3bayes_fit</code> .
<code>newdata</code>	Optional data frame. <code>NULL</code> uses the fitted prepared data.
<code>type</code>	Prediction quantity: "expected", "predictive", "linear", or "median".
<code>include_group_effects</code>	Whether fitted group-level effects are included.
<code>allow_new_levels</code>	Whether new grouping levels are permitted by brms.
<code>ndraws</code>	Optional number of posterior draws.
<code>probs</code>	Three probabilities used to summarise predictions.
<code>seed</code>	Non-negative seed used for posterior predictive simulation.

Value

A `gp3bayes_prediction` containing draws, summaries, prediction data, and interpretation meta-data.

`predict_pupil_trajectory`

Predict governed pupil trajectories

Description

Obtains expected, posterior-predictive, or linear-predictor draws from an approved pupil fit with explicit draw and grid-size guards.

Usage

```

predict_pupil_trajectory(
  fit,
  newdata = NULL,
  type = c("expected", "posterior_predictive", "linear"),
  ndraws = 500L,
  population_only = TRUE,
  allow_new_levels = FALSE,
  max_grid = 5000L,
  max_cells = 5000000L
)

```

Arguments

<code>fit</code>	A <code>gp3bayes_pupil_fit</code> .
<code>newdata</code>	Optional prepared prediction grid. When omitted, a compact population grid is built from observed event times and conditions. Participant-conditioned prediction requires explicit <code>newdata</code> .
<code>type</code>	"expected", "posterior_predictive", or "linear".
<code>ndraws</code>	Maximum posterior draws to retain.
<code>population_only</code>	If TRUE, group-level effects are excluded from the prediction via <code>re_formula = NA</code> .
<code>allow_new_levels</code>	Passed conservatively to brms prediction methods.
<code>max_grid</code>	Maximum grid rows.
<code>max_cells</code>	Maximum draw-by-grid cells.

Value

A `gp3bayes_pupil_prediction`.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

```
preflight_model_specification
```

Preflight an Approved Model Specification

Description

Convenience wrapper around `audit_design_support()` for an already-created specification.

Usage

```
preflight_model_specification(specification, ...)
```

Arguments

<code>specification</code>	A <code>gp3bayes</code> model specification.
<code>...</code>	Arguments passed to <code>audit_design_support()</code> .

Value

A `gp3bayes_design_support_audit`.

`prepare_binocular_pupil_timecourse`*Prepare binocular pupil data without averaging eyes*

Description

Prepare binocular pupil data without averaging eyes

Usage

```
prepare_binocular_pupil_timecourse(  
  data,  
  left_col = "pupil_left",  
  right_col = "pupil_right",  
  participant_col = "participant_id",  
  time_col = "time_ms",  
  condition_col = "condition",  
  trial_col = "trial_id",  
  item_col = NULL,  
  covariates = character()  
)
```

Arguments

<code>data</code>	A data frame containing left and right pupil responses.
<code>left_col, right_col</code>	Left/right pupil response columns.
<code>participant_col, time_col, condition_col</code>	Structural columns.
<code>trial_col, item_col</code>	Optional structural columns.
<code>covariates</code>	Optional additional covariates.

Value

A `gp3bayes_binocular_pupil_prepared` object.

```
prepare_hierarchical_binary_data
```

Prepare Hierarchical Binary Data

Description

Applies explicit binary outcome mapping, explicit two-level condition coding, optional recorded numeric scaling, and a model-readiness gate. No variable is silently scaled or recoded.

Usage

```
prepare_hierarchical_binary_data(
  data,
  contract,
  outcome_mapping = NULL,
  condition_levels = NULL,
  condition_coding = c(-0.5, 0.5),
  scale_predictors = character(),
  scale_time = FALSE,
  missing = c("error", "drop")
)
```

Arguments

<code>data</code>	A data frame containing the columns declared in <code>contract</code> .
<code>contract</code>	A binary <code>gp3bayes_model_contract</code> .
<code>outcome_mapping</code>	Optional named vector mapping two labelled outcome values to 0 and 1. It is required for non-logical, non-0/1 outcomes.
<code>condition_levels</code>	Optional two-value vector listing the condition levels in reference-to-focal order.
<code>condition_coding</code>	Two distinct finite numeric values used to encode the declared condition. The default is <code>c(-0.5, 0.5)</code> .
<code>scale_predictors</code>	Character vector naming declared numeric predictors to centre and divide by their sample standard deviation.
<code>scale_time</code>	Whether to centre and scale the declared linear time variable.
<code>missing</code>	Either "error" or "drop". Dropping is performed only after this explicit argument is selected, and removed row positions are recorded.

Details

This function performs deterministic preparation only. It does not fit a model, create posterior draws, or establish causal or substantive validity.

Value

A gp3bayes_binary_prepared object containing the analysis data, contract, readiness audit, transformation registry, fixed-effects formula, design-matrix columns, and row accounting.

Examples

```
simulation <- simulate_hierarchical_binary_data(
  n_participants = 12,
  trials_per_participant = 8,
  seed = 2026
)

contract <- create_model_contract(
  family = "binary",
  outcome_col = "selected",
  participant_col = "participant_id",
  item_col = "item_id",
  trial_col = "trial_id",
  condition_col = "condition",
  predictors = c(
    "participant_covariate",
    "trial_covariate"
  ),
  interaction = c(
    "condition",
    "participant_covariate"
  ),
  random_slope = TRUE
)

prepared <- prepare_hierarchical_binary_data(
  simulation$data,
  contract,
  condition_levels = c("control", "treatment")
)

prepared
```

`prepare_hierarchical_duration_data`*Prepare Hierarchical Duration Data*

Description

Validates strictly positive finite uncensored durations, applies explicit unit conversion and recorded scaling, and runs the duration readiness gate.

Usage

```
prepare_hierarchical_duration_data(
  data,
  contract,
  condition_levels = NULL,
  condition_coding = c(-0.5, 0.5),
  scale_predictors = character(),
  scale_time = FALSE,
  outcome_multiplier = 1,
  converted_unit = NULL,
  missing = c("error", "drop")
)
```

Arguments

<code>data</code>	A data frame containing columns declared in <code>contract</code> .
<code>contract</code>	A duration <code>gp3bayes_model_contract</code> .
<code>condition_levels</code>	Optional two-value condition order.
<code>condition_coding</code>	Two distinct finite numeric condition codes.
<code>scale_predictors</code>	Declared numeric predictors to centre and scale.
<code>scale_time</code>	Whether to centre and scale the declared linear time variable.
<code>outcome_multiplier</code>	Positive unit-conversion multiplier.
<code>converted_unit</code>	Required destination unit when <code>outcome_multiplier != 1</code> .
<code>missing</code>	Either "error" or "drop".

Details

Zero, negative, non-finite, censored, truncated, or shifted durations are not supported. Unit conversion is never inferred.

Value

A `gp3bayes_duration_prepared` object.

```
prepare_pupil_timecourse
```

Prepare a pupil time course under explicit transformations

Description

Validates ordering, identifiers, event time, pupil values, measurement flags, sampling intervals, and baseline support. Only explicitly requested deterministic transformations are applied and recorded.

Usage

```
prepare_pupil_timecourse(
  data,
  contract,
  baseline_operation = c("none", "subtract", "divide", "proportion_change",
    "percent_change"),
  baseline_window = NULL,
  output_unit = NULL,
  scale_covariates = character(),
  max_rows = 2000000L,
  irregularity_review_cv = 0.1
)
```

Arguments

<code>data</code>	Source data frame.
<code>contract</code>	A <code>create_pupil_contract()</code> result.
<code>baseline_operation</code>	One of "none", "subtract", "divide", "proportion_change", or "percent_change".
<code>baseline_window</code>	Required when <code>baseline_operation != "none"</code> unless already recorded in the contract. Values use the contract's source <code>time_unit</code> and are converted to canonical seconds during preparation.
<code>output_unit</code>	Optional physical output unit. Only metre/millimetre conversions are defined.
<code>scale_covariates</code>	Declared numeric covariates to standardize.
<code>max_rows</code>	Maximum accepted input rows.
<code>irregularity_review_cv</code>	Coefficient-of-variation threshold recorded as a sampling-irregularity review signal.

Value

A `gp3bayes_pupil_prepared` object. The source pupil values are retained in `.pupil_source`; the modelled values are `.pupil_model`.

Governance boundary

This function does not detect or interpolate blinks, smooth traces, choose a baseline, correct PFE, correct luminance, or automatically exclude samples.

Examples

```
sim <- simulate_pupil_timecourse(
  n_participants = 3, trials_per_participant = 3,
  sampling_frequency = 20, seed = 11
)
```

```

contract <- create_pupil_contract(
  "pupil_mm", "participant_id", "trial_id", "event_time",
  "millimetres", 20, condition_col = "condition",
  blink_col = "blink", interpolation_col = "interpolated",
  validity_col = "valid", gaze_x_col = "gaze_x", gaze_y_col = "gaze_y",
  luminance_col = "luminance", baseline_window = c(-0.5, 0)
)
prepared <- prepare_pupil_timecourse(
  sim$data, contract, baseline_operation = "subtract"
)
prepared

```

prior_posterior_bridge

Create a Declared-Prior versus Posterior Bridge

Description

Compares marginal draws from the recorded gp3bayes prior specification with fitted posterior draws on the same parameter scale.

Usage

```

prior_posterior_bridge(
  fit,
  variables = NULL,
  regex = "^(b_|sd_|cor_|sigma$)",
  ndraws = 4000L,
  probs = c(0.025, 0.5, 0.975),
  seed = 1L
)

```

Arguments

<code>fit</code>	A fitted gp3bayes_fit.
<code>variables</code>	Optional exact posterior variables.
<code>regex</code>	Optional posterior-variable regular expression.
<code>ndraws</code>	Number of prior draws and maximum posterior draws used.
<code>probs</code>	Three interval probabilities.
<code>seed</code>	Non-negative integer seed.

Value

A gp3bayes_prior_posterior_bridge.

`prior_posterior_distance_table`*Prior-versus-Posterior Distance Table*

Description

Prior-versus-Posterior Distance Table

Usage

```
prior_posterior_distance_table(x)
```

Arguments

`x` A gp3bayes_prior_posterior_bridge.

Value

Marginal empirical distribution-distance summaries.

`prior_posterior_draws_long`*Long Prior and Posterior Draw Table*

Description

Long Prior and Posterior Draw Table

Usage

```
prior_posterior_draws_long(x, max_draws = 1000L, seed = 1L)
```

Arguments

`x` A gp3bayes_prior_posterior_bridge.
`max_draws` Maximum draws per distribution and variable.
`seed` Non-negative integer seed.

Value

A long draw table.

prior_posterior_summary_table
<i>Prior-versus-Posterior Summary Table</i>

Description

Prior-versus-Posterior Summary Table

Usage

prior_posterior_summary_table(x)

Arguments

x A gp3bayes_prior_posterior_bridge.

Value

Location, spread, interval, shift, and contraction summaries.

prior_sensitivity_scenario_table
<i>Prior Sensitivity Scenario Table</i>

Description

Prior Sensitivity Scenario Table

Usage

prior_sensitivity_scenario_table(x)

Arguments

x A gp3bayes_prior_sensitivity.

Value

Scenario-level maximum shifts and diagnostic statuses.

`prior_sensitivity_table`*Prior Sensitivity Table*

Description

Prior Sensitivity Table

Usage

```
prior_sensitivity_table(x)
```

Arguments

`x` A gp3bayes_prior_sensitivity.

Value

Parameter-by-scenario posterior shifts.

`prior_specification_table`*Prior-Specification Table*

Description

Prior-Specification Table

Usage

```
prior_specification_table(x)
```

Arguments

`x` A gp3bayes fit, specification, or prior specification.

Value

The backend-independent declared prior table.

publication_registry_table
<i>Publication Registry Table</i>

Description

Publication Registry Table

Usage

publication_registry_table(x)

Arguments

x A publication registry.

Value

Entry metadata.

pupil_advanced_capabilities
<i>Report the gp3bayes 0.5 advanced-pupillometry capability boundary</i>

Description

Report the gp3bayes 0.5 advanced-pupillometry capability boundary

Usage

pupil_advanced_capabilities()

Value

A data frame listing supported, experimental, and deliberately excluded capabilities.

pupil_advanced_compatibility_table

Report governed compatibility rules for 0.5 advanced models

Description

Report governed compatibility rules for 0.5 advanced models

Usage

pupil_advanced_compatibility_table()

Value

A data frame documenting combinations that are supported, reviewed, or deliberately blocked in the 0.5 governed interface.

pupil_advanced_mapping_table

Inspect the resolved 0.5 pupil column mapping

Description

Resolves the response, time, participant, condition, trial, and item columns that the advanced 0.5 layer will use. Resolution is read-only and does not modify the prepared object.

Usage

pupil_advanced_mapping_table(prepared)

Arguments

prepared A 0.4 prepared pupil object or compatible data frame.

Value

A data frame describing the resolved mapping.

`pupil_advanced_specification_table`*Tabulate an advanced pupil model specification*

Description

Tabulate an advanced pupil model specification

Usage

```
pupil_advanced_specification_table(x)
```

Arguments

`x` An advanced specification.

Value

A one-row data frame.

`pupil_autocorrelation_table`*Tabulate a temporal-dependence audit*

Description

Tabulate a temporal-dependence audit

Usage

```
pupil_autocorrelation_table(x, level = c("summary", "series"))
```

Arguments

`x` A temporal-dependence audit.
`level` "summary" or "series".

pupil_binocular_agreement_table
Summarise binocular posterior agreement

Description

Summarise binocular posterior agreement

Usage

```
pupil_binocular_agreement_table(trajecotory, tolerance = 0.1)
```

Arguments

trajecotory A binocular trajectory object.
tolerance A scientifically declared absolute right-minus-left tolerance.

Value

A data frame with posterior agreement probabilities.

pupil_binocular_correlation
Extract posterior residual binocular correlation

Description

Extract posterior residual binocular correlation

Usage

```
pupil_binocular_correlation(fit, probability = 0.95)
```

Arguments

fit A binocular fit with residual correlation enabled.
probability Central interval probability.

Value

A data frame.

pupil_binocular_difference

Estimate posterior right-minus-left binocular differences

Description

Estimate posterior right-minus-left binocular differences

Usage

```
pupil_binocular_difference(x)
```

Arguments

x A binocular trajectory object.

Value

A data frame.

pupil_condition_contrast

Estimate a posterior condition-difference trajectory

Description

Estimate a posterior condition-difference trajectory

Usage

```
pupil_condition_contrast(
  prediction,
  contrast,
  threshold = 0,
  probability = 0.95
)
```

Arguments

prediction Pupil prediction with a .condition column.
 contrast Character vector c(level_a, level_b) defining a - b.
 threshold Scientifically declared threshold on the pupil scale.
 probability Credible probability.

Value

A pupil trajectory/contrast object with pointwise probabilities that the declared contrast exceeds threshold.

Examples

```
grid <- expand.grid(
  .event_time = seq(0, 1, length.out = 5),
  .condition = factor(c("control", "treatment"))
)
draws <- matrix(rnorm(1000), nrow = 100, ncol = nrow(grid))
prediction <- as_pupil_prediction_draws(draws, grid, "millimetres")
pupil_condition_contrast(
  prediction, contrast = c("treatment", "control"), threshold = 0.1
)
```

pupil_distribution_table

Tabulate a pupil distribution declaration or advanced specification

Description

Tabulate a pupil distribution declaration or advanced specification

Usage

```
pupil_distribution_table(x)
```

Arguments

x A distribution declaration, advanced specification, or advanced fit.

Value

A one-row data frame.

pupil_dynamic_contrast_table

Tabulate a dynamic pupil contrast

Description

Tabulate a dynamic pupil contrast

Usage

```
pupil_dynamic_contrast_table(x)
```

Arguments

x A dynamic-contrast object.

Value

A data frame.

pupil_gp_hyperparameters	<i>Extract Gaussian-process hyperparameters</i>
--------------------------	---

Description

Extract Gaussian-process hyperparameters

Usage

pupil_gp_hyperparameters(fit, probability = 0.95)

Arguments

fit A fitted GP pupil model.
probability Central interval probability.

Value

A gp3bayes_pupil_gp_hyperparameters object.

pupil_gp_table	<i>Tabulate GP hyperparameters</i>
----------------	------------------------------------

Description

Tabulate GP hyperparameters

Usage

pupil_gp_table(x)

Arguments

x A GP-hyperparameter object.

Value

A data frame.

`pupil_identifiability_table`*Tabulate advanced pupil identifiability/design-support audit*

Description

Tabulate advanced pupil identifiability/design-support audit

Usage

```
pupil_identifiability_table(x)
```

Arguments

`x` An identifiability audit.

Value

A data frame.

`pupil_measurement_audit_table`*Extract a pupil measurement-context table*

Description

Extract a pupil measurement-context table

Usage

```
pupil_measurement_audit_table(x)
```

Arguments

`x` A gp3bayes_pupil_measurement_audit.

Value

A data frame.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

pupil_measurement_uncertainty_table

Summarise measurement uncertainty declared in a model

Description

Summarise measurement uncertainty declared in a model

Usage

```
pupil_measurement_uncertainty_table(x)
```

Arguments

x A measurement model, advanced specification, or advanced fit.

Value

A data frame.

pupil_missingness_table

Tabulate missingness audit results

Description

Tabulate missingness audit results

Usage

```
pupil_missingness_table(x)
```

Arguments

x A missingness audit.

Value

A data frame.

pupil_model_card	<i>Build an auditable advanced pupil model card</i>
------------------	---

Description

Build an auditable advanced pupil model card

Usage

```
pupil_model_card(x)
```

Arguments

x	An advanced specification or fit.
---	-----------------------------------

Value

A gp3bayes_pupil_model_card object containing structured metadata and governance statements suitable for methods supplements.

pupil_model_card_table	<i>Tabulate a pupil model card</i>
------------------------	------------------------------------

Description

Tabulate a pupil model card

Usage

```
pupil_model_card_table(x)
```

Arguments

x	A model-card object, specification, or fit.
---	---

pupil_model_comparison_table
Tabulate model comparison

Description

Tabulate model comparison

Usage

```
pupil_model_comparison_table(x)
```

Arguments

x A model-comparison object.

Value

A data frame.

pupil_model_weights *Compute explicit predictive model weights*

Description

Compute explicit predictive model weights

Usage

```
pupil_model_weights(x, method = c("stacking", "pseudobma"), BB = TRUE)
```

Arguments

x A LOO-based model comparison or a model set.
method "stacking" or "pseudobma".
BB Bayesian bootstrap for pseudo-BMA where applicable.

Value

A data frame of weights. Weights are not used automatically for prediction or model selection.

pupil_ppc_table *Extract pupil PPC evidence tables*

Description

Extract pupil PPC evidence tables

Usage

```
pupil_ppc_table(
  x,
  component = c("trajectory", "distribution", "features", "residuals",
    "residual_trajectory", "autocorrelation", "heterogeneity", "measurement_context")
)
```

Arguments

x	A pupil PPC object.
component	One of "trajectory", "distribution", "features", "residuals", "residual_trajectory", "autocorrelation", "heterogeneity", or "measurement_context".

Value

A data frame.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

pupil_readiness_table *Extract pupil readiness tables*

Description

Extract pupil readiness tables

Usage

```
pupil_readiness_table(
  x,
  component = c("summary", "participant", "condition", "trial")
)
```

Arguments

x	A gp3bayes_pupil_readiness.
component	One of "summary", "participant", "condition", or "trial".

Value

A data frame.

Examples

See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.

pupil_residual_acf	<i>Summarise residual autocorrelation for a pupil fit</i>
--------------------	---

Description

Summarise residual autocorrelation for a pupil fit

Usage

pupil_residual_acf(x, max_lag = 10L, ndraws = 200L)

Arguments

- x A pupil fit or pupil diagnostics object.
- max_lag Maximum lag when x is a fit.
- ndraws Draws for fit-based expected residuals.

Value

A data frame of mean within-series residual autocorrelations.

Examples

See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.

pupil_residual_scale_table	<i>Tabulate posterior residual scale</i>
----------------------------	--

Description

Tabulate posterior residual scale

Usage

pupil_residual_scale_table(x)

Arguments

x A residual-scale estimand.

Value

A data frame.

pupil_residual_spectrum
Compute a descriptive residual spectrum

Description

Provides a frequency-domain diagnostic for residual periodicity after posterior mean subtraction. It does not infer physiological oscillations.

Usage

```
pupil_residual_spectrum(fit, ndraws = 300L)
```

Arguments

fit An advanced fit.
ndraws Posterior expected-mean draws.

Value

A gp3bayes_pupil_residual_spectrum object.

pupil_response_parameter_table
Tabulate nonlinear response parameters

Description

Tabulate nonlinear response parameters

Usage

```
pupil_response_parameter_table(x)
```

Arguments

x A response-parameter object.

`pupil_sensitivity_table`*Extract pupil sensitivity scenarios or comparison results*

Description

Extract pupil sensitivity scenarios or comparison results

Usage

```
pupil_sensitivity_table(x)
```

Arguments

`x` A pupil sensitivity suite or sensitivity comparison.

Value

A data frame.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

`pupil_specification_table`*Convert a pupil specification to a publication-ready table*

Description

Convert a pupil specification to a publication-ready table

Usage

```
pupil_specification_table(x)
```

Arguments

`x` A pupil model specification.

Value

A data frame.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

`pupil_trajectory_derivative_table`*Tabulate posterior pupil trajectory derivatives*

Description

Tabulate posterior pupil trajectory derivatives

Usage

```
pupil_trajectory_derivative_table(x, probability = x$probability)
```

Arguments

<code>x</code>	A trajectory-derivative object.
<code>probability</code>	Optional interval probability overriding the stored value.

Value

A data frame.

`pupil_trajectory_table`*Convert a pupil trajectory to a table*

Description

Convert a pupil trajectory to a table

Usage

```
pupil_trajectory_table(x)
```

Arguments

<code>x</code>	A pupil trajectory object.
----------------	----------------------------

Value

A data frame.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

pupil_validation_table

Extract a pupil validation table

Description

Extract a pupil validation table

Usage

```
pupil_validation_table(x)
```

Arguments

x A pupil validation object.

Value

A data frame.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

random_intercept_variance_partition

Random-Intercept Latent Variance Partition

Description

For binary logit models the residual latent variance is $\pi^2 / 3$. For lognormal duration models residual log-scale variance is σ^2 . Random-slope variance is deliberately excluded.

Usage

```
random_intercept_variance_partition(fit, probs = c(0.025, 0.5, 0.975))
```

Arguments

fit A fitted gp3bayes_fit.
probs Three interval probabilities.

Value

A gp3bayes_random_intercept_variance_partition.

random_intercept_variance_partition_table
<i>Random-Intercept Variance-Partition Table</i>

Description

Random-Intercept Variance-Partition Table

Usage

random_intercept_variance_partition_table(x)

Arguments

x A variance-partition object.

Value

Component-level variance and fraction summaries.

random_slope_sensitivity_table
<i>Random-Slope Sensitivity Table</i>

Description

Random-Slope Sensitivity Table

Usage

random_slope_sensitivity_table(x)

Arguments

x A gp3bayes_random_slope_sensitivity.

Value

The retained estimand-sensitivity comparison.

`read_analysis_manifest`*Read a Frozen Analysis Manifest*

Description

Read a Frozen Analysis Manifest

Usage

```
read_analysis_manifest(file)
```

Arguments

file Explicit .rds manifest path.

Value

A validated gp3bayes_analysis_manifest.

`read_gp3bayes_schema` *Read a Frozen gp3bayes Object Schema*

Description

Read a Frozen gp3bayes Object Schema

Usage

```
read_gp3bayes_schema(file)
```

Arguments

file Explicit .rds schema path.

Value

A gp3bayes_object_schema.

recovery_estimate_table
<i>Recovery Estimate Table</i>

Description

Recovery Estimate Table

Usage

recovery_estimate_table(x)

Arguments

x A gp3bayes_recovery.

Value

Repetition-level truth, posterior summaries, and coverage records.

recovery_fit_status_table
<i>Recovery Fit-Status Table</i>

Description

Recovery Fit-Status Table

Usage

recovery_fit_status_table(x)

Arguments

x A gp3bayes_recovery.

Value

Repetition-level completion and diagnostic-status records.

recovery_parameter_table
<i>Recovery Parameter Summary Table</i>

Description

Recovery Parameter Summary Table

Usage

recovery_parameter_table(x)

Arguments

x A gp3bayes_recovery.

Value

Parameter-level recovery summaries.

register_publication_figure
<i>Register a Publication Figure</i>

Description

Register a Publication Figure

Usage

```
register_publication_figure(  
  registry,  
  name,  
  figure,  
  caption = NULL,  
  source = NULL  
)
```

Arguments

registry	A publication registry.
name	Unique entry name.
figure	A ggplot, bayesplot grid, or gtable.
caption	Optional caption.
source	Optional source label.

Value

An updated registry.

register_publication_table
<i>Register a Publication Table</i>

Description

Register a Publication Table

Usage

```
register_publication_table(  
  registry,  
  name,  
  table,  
  caption = NULL,  
  source = NULL  
)
```

Arguments

registry	A publication registry.
name	Unique entry name.
table	A data frame.
caption	Optional caption.
source	Optional source label.

Value

An updated registry.

review_duration_extremes
<i>Review Extreme Positive Durations Without Deleting Them</i>

Description

Flags observations that are extreme on the log-duration scale using both a robust MAD rule and an outer-IQR rule. The function never deletes values and never changes the model family automatically.

Usage

```
review_duration_extremes(data, contract, mad_cutoff = 4, iqr_multiplier = 3)
```

Arguments

`data` A data frame.

`contract` An approved duration contract.

`mad_cutoff` Robust absolute z-score cutoff on log durations.

`iqr_multiplier` Multiplier for the outer-IQR rule on log durations.

Value

A gp3bayes_duration_extreme_review object.

run_binary_recovery	<i>Run Binary Parameter Recovery</i>
---------------------	--------------------------------------

Description

Repeatedly simulates from the approved hierarchical Bernoulli-logit generator, fits the restricted model, and compares posterior intervals with known generating values.

Usage

```
run_binary_recovery(
  repetitions = 20L,
  n_participants = 30L,
  trials_per_participant = 16L,
  n_items = 12L,
  include_items = TRUE,
  random_slope = TRUE,
  seed = 1001L,
  chains = 4L,
  iter = 1500L,
  warmup = 750L,
  cores = min(chains, .gp3b_default_cores(chains)),
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L,
  interval_probability = 0.95,
  minimum_repetitions = 20L,
  maximum_standardized_bias = 0.25,
  minimum_coverage = 0.8,
  minimum_diagnostic_pass_fraction = 0.8,
  continue_on_error = TRUE
)
```


Arguments

repetitions	Number of simulation-fit repetitions.
n_participants, trials_per_participant, n_items	Synthetic design sizes.
include_items	Whether crossed item effects are included.
random_slope	Whether a participant condition slope is generated and fitted.
seed	First simulation seed.
chains, iter, warmup, cores, adapt_delta, max_treedepth, refresh	Restricted sampling controls.
interval_probability	Central posterior interval probability.
minimum_repetitions	Repetitions required before an overall pass is possible.
maximum_standardized_bias	Maximum absolute bias divided by the empirical standard deviation of estimates for a pass.
minimum_coverage	Minimum empirical interval coverage for a pass.
minimum_diagnostic_pass_fraction	Minimum fraction of fits with a diagnostic pass.
continue_on_error	Whether failed repetitions are recorded instead of stopping.

Details

A small recovery run is a smoke test, not validation. Even when all declared thresholds pass, the object records no automatic validation claim.

Value

A gp3bayes_binary_recovery object.

run_duration_recovery *Run Duration Parameter Recovery*

Description

Repeatedly simulates and fits the approved hierarchical lognormal duration model and compares posterior intervals with known generating values.

Usage

```
run_duration_recovery(
  repetitions = 20L,
  n_participants = 30L,
  trials_per_participant = 16L,
  n_items = 12L,
  include_items = TRUE,
  random_slope = TRUE,
  baseline_median = 500,
  outcome_unit = "milliseconds",
  seed = 2001L,
  chains = 4L,
  iter = 1500L,
  warmup = 750L,
  cores = min(chains, .gp3b_default_cores(chains)),
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L,
  interval_probability = 0.95,
  minimum_repetitions = 20L,
  maximum_standardized_bias = 0.25,
  minimum_coverage = 0.8,
  minimum_diagnostic_pass_fraction = 0.8,
  continue_on_error = TRUE
)
```

Arguments

repetitions	Number of simulation-fit repetitions.
n_participants, trials_per_participant, n_items	Synthetic design sizes.
include_items	Whether crossed item effects are included.
random_slope	Whether a participant condition slope is generated and fitted.
baseline_median	Baseline synthetic duration median.
outcome_unit	Synthetic duration unit.
seed	First simulation seed.
chains, iter, warmup, cores, adapt_delta, max_treedepth, refresh	Restricted sampling controls.
interval_probability	Central posterior interval probability.
minimum_repetitions	Repetitions required before an overall pass is possible.
maximum_standardized_bias	Maximum absolute bias divided by the empirical standard deviation of estimates for a pass.

minimum_coverage
 Minimum empirical interval coverage for a pass.
 minimum_diagnostic_pass_fraction
 Minimum fraction of fits with a diagnostic pass.
 continue_on_error
 Whether failed repetitions are recorded instead of stopping.

Details

A small run is a smoke test. No result creates an automatic validation claim.

Value

A gp3bayes_duration_recovery.

run_group_deletion_sensitivity
Run a Group-Deletion Sensitivity Plan

Description

This is an intentionally explicit refitting workflow. It can be expensive. The result reports how the primary estimand changes when declared units are omitted, without automatically labelling any unit invalid.

Usage

```
run_group_deletion_sensitivity(
  plan,
  backend = c("rstan", "cmdstanr"),
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = 1L,
  seed = 1L,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L,
  ndraws = NULL,
  retain_fits = FALSE
)
```

Arguments

plan A group-deletion sensitivity plan.
 backend "rstan" or "cmdstanr".

chains, iter, warmup, cores, seed, adapt_delta, max_treedepth, refresh
 Restricted sampling controls.
 ndraws Optional number of draws used for each estimand.
 retain_fits Whether fitted objects are retained.

Value

A gp3bayes_group_deletion_sensitivity.

run_random_slope_sensitivity

Run Random-Intercept versus Random-Slope Sensitivity

Description

Run Random-Intercept versus Random-Slope Sensitivity

Usage

```
run_random_slope_sensitivity(
  plan,
  backend = c("rstan", "cmdstanr"),
  chains = 4L,
  iter = 2000L,
  warmup = 1000L,
  cores = 1L,
  seed = 1L,
  adapt_delta = 0.95,
  max_treedepth = 12L,
  refresh = 0L,
  ndraws = NULL,
  retain_fits = FALSE
)
```

Arguments

plan A random-slope sensitivity plan.
 backend "rstan" or "cmdstanr".
 chains, iter, warmup, cores, seed, adapt_delta, max_treedepth, refresh
 Restricted sampling controls.
 ndraws Optional number of draws used for each estimand.
 retain_fits Whether fitted objects are retained.

Value

A gp3bayes_random_slope_sensitivity.

run_sbc_plan

Run a Simulation-Based Calibration Plan

Description

Run a Simulation-Based Calibration Plan

Usage

```
run_sbc_plan(
  plan,
  cores_per_fit = 1L,
  keep_fits = FALSE,
  thin_ranks = NULL,
  cache_mode = "none",
  cache_location = NULL
)
```

Arguments

plan	A gp3bayes_sbc_plan.
cores_per_fit	Cores used by each fit.
keep_fits	Whether to retain fitted objects.
thin_ranks	Optional rank thinning.
cache_mode	SBC cache mode.
cache_location	Optional cache location.

Value

A gp3bayes_sbc_result.

run_sensitivity_suite *Run a Unified Sensitivity Suite*

Description

Orchestrates approved sensitivity components. Refitting components only run when explicitly requested in plan. Failures are retained as inspectable component results unless stop_on_error = TRUE.

Usage

```
run_sensitivity_suite(
  fit,
  plan = create_sensitivity_suite_plan(),
  reference_estimand = NULL,
  stop_on_error = FALSE
)
```

Arguments

fit An approved gp3bayes fit.

plan A `create_sensitivity_suite_plan()` result.

reference_estimand Optional precomputed primary estimand. When alternatives are supplied and this is omitted it is computed with `estimate_model_estimands()`.

stop_on_error Whether the first component error should stop the suite.

Value

A gp3bayes_sensitivity_suite.

sampler_diagnostic_table

Summarise NUTS Sampler Diagnostics

Description

Summarise NUTS Sampler Diagnostics

Usage

```
sampler_diagnostic_table(fit)
```

Arguments

fit A fitted gp3bayes_fit.

Value

A data frame containing sampler-level summaries and review flags.

save_figure_set	<i>Save a Figure Set</i>
-----------------	--------------------------

Description

File output is explicit: no current-directory default is provided.

Usage

```
save_figure_set(  
    x,  
    directory,  
    width = 7,  
    height = 5,  
    dpi = 300,  
    device = "png",  
    overwrite = FALSE  
)
```

Arguments

- x A gp3bayes_figure_set.
- directory Existing or creatable output directory.
- width, height Figure dimensions in inches.
- dpi Raster resolution.
- device File extension/device, such as "png" or "pdf".
- overwrite Whether existing files may be replaced.

Value

Invisibly, the written file paths.

save_publication_registry_figures	<i>Save Publication Registry Figures</i>
-----------------------------------	--

Description

Save Publication Registry Figures

Usage

```
save_publication_registry_figures(  
  x,  
  directory,  
  width = 7,  
  height = 5,  
  dpi = 300,  
  device = "png",  
  overwrite = FALSE  
)
```

Arguments

x	A publication registry.
directory	Explicit output directory.
width, height	Figure dimensions in inches.
dpi	Raster resolution.
device	Graphics device/extension.
overwrite	Whether existing files may be replaced.

Value

Invisibly, written file paths.

sbc_overview_table	<i>SBC Overview Table</i>
--------------------	---------------------------

Description

SBC Overview Table

Usage

```
sbc_overview_table(x)
```

Arguments

x	A gp3bayes_sbc_result.
---	------------------------

Value

A conservative one-row metadata summary.

sbc_stats_table	<i>SBC Statistics Table</i>
-----------------	-----------------------------

Description

SBC Statistics Table

Usage

sbc_stats_table(x)

Arguments

x A gp3bayes_sbc_result.

Value

The statistics table retained by the SBC result.

schema_comparison_table	<i>Schema-Comparison Table</i>
-------------------------	--------------------------------

Description

Schema-Comparison Table

Usage

schema_comparison_table(x)

Arguments

x A gp3bayes_schema_comparison.

Value

Structural schema comparisons.

score_pupil_predictions
<i>Score posterior predictive draws against observed pupil values</i>

Description

Metrics are descriptive out-of-sample or held-out scores only when the caller supplies predictions generated without using the scored observations.

Usage

```
score_pupil_predictions(observed, draws, probability = 0.9)
```

Arguments

- observed Numeric observed values.
- draws Matrix of posterior predictive draws (draws x observations).
- probability Central interval used for empirical coverage and width.

Value

A gp3bayes_pupil_predictive_score object.

sensitivity_suite_table
<i>Sensitivity-Suite Table</i>

Description

Sensitivity-Suite Table

Usage

```
sensitivity_suite_table(x)
```

Arguments

- x A gp3bayes_sensitivity_suite.

Value

The component-level sensitivity table.

`simulate_advanced_pupil_timecourse`*Simulate advanced dynamic pupil time courses*

Description

Generates deterministic hierarchical traces with optional Student-t contamination, heteroskedasticity, ARMA dependence, measurement error, and missingness. The simulator is intended for examples, recovery studies, and failure-path validation; it does not claim physiological realism.

Usage

```
simulate_advanced_pupil_timecourse(  
  n_participants = 24L,  
  trials_per_participant = 6L,  
  time_points = 41L,  
  time_range = c(-500, 2500),  
  conditions = c("control", "treatment"),  
  family = c("gaussian", "student"),  
  residual_scale = 0.08,  
  heteroskedastic_strength = 0.35,  
  ar = 0.45,  
  ma = numeric(),  
  participant_sd = 0.12,  
  amplitude_condition = 0.22,  
  latency_condition = 120,  
  outlier_fraction = 0.01,  
  missing_fraction = 0.03,  
  measurement_error_sd = 0.015,  
  student_df = 5,  
  seed = 2026  
)
```

Arguments

<code>n_participants</code>	Number of participants.
<code>trials_per_participant</code>	Trials per participant.
<code>time_points</code>	Number of samples per trial.
<code>time_range</code>	Numeric length-two time range in milliseconds.
<code>conditions</code>	Character condition labels.
<code>family</code>	"gaussian" or "student".
<code>residual_scale</code>	Baseline residual SD.
<code>heteroskedastic_strength</code>	Multiplicative time-varying noise strength.

ar	Numeric AR coefficients, length at most 3.
ma	Numeric MA coefficients, length at most 2.
participant_sd	Participant random-intercept SD.
amplitude_condition	Difference in response amplitude for condition 2.
latency_condition	Difference in peak latency for condition 2.
outlier_fraction	Fraction of observations receiving extra contamination.
missing_fraction	Fraction of pupil observations set missing.
measurement_error_sd	Known response-measurement SD; zero disables.
student_df	Degrees of freedom when family = "student".
seed	Random seed.

Value

A gp3bayes_pupil_advanced_simulation object with data and stored truth.

simulate_binary_pathology
<i>Simulate a Dedicated Binary Pathology Scenario</i>

Description

Simulate a Dedicated Binary Pathology Scenario

Usage

```
simulate_binary_pathology(  
  scenario = c("null_contrast", "weak_information", "severe_imbalance",  
    "near_separation", "omitted_random_slope", "sparse_item_structure",  
    "all_zero_participants", "rank_deficiency", "missing_outcomes"),  
  seed = 1L  
)
```

Arguments

scenario	Pathological or stress-test scenario.
seed	Non-negative integer seed.

Value

A gp3bayes_pathological_simulation.

```
simulate_binocular_pupil_timecourse
```

Simulate joint binocular pupil traces

Description

Simulate joint binocular pupil traces

Usage

```
simulate_binocular_pupil_timecourse(
  ...,
  residual_correlation = 0.65,
  eye_bias = 0.015,
  eye_specific_sd = 0.035
)
```

Arguments

```
...           Arguments passed to simulate\_advanced\_pupil\_timecourse\(\).
residual_correlation
               Approximate left/right innovation correlation.
eye_bias       Mean right-minus-left difference.
eye_specific_sd
               Eye-specific noise SD.
```

Value

A gp3bayes_binocular_pupil_simulation object.

```
simulate_declared_prior_draws
```

Simulate Marginal Draws from Declared gp3bayes Priors

Description

Simulate Marginal Draws from Declared gp3bayes Priors

Usage

```
simulate_declared_prior_draws(
  x,
  variables = NULL,
  regex = NULL,
  ndraws = 4000L,
  seed = 1L
)
```

Arguments

x	A gp3bayes fit, specification, or prior specification.
variables	Posterior-style variable names. When x is a fitted model, supported variables can be inferred.
regex	Optional regular expression applied after inference.
ndraws	Number of marginal prior draws.
seed	Non-negative integer seed.

Value

A numeric draw matrix.

simulate_duration_pathology

Simulate a Dedicated Duration Pathology Scenario

Description

Simulate a Dedicated Duration Pathology Scenario

Usage

```
simulate_duration_pathology(
  scenario = c("null_ratio", "high_group_heterogeneity", "weak_information",
    "severe_imbalance", "heavy_tailed_contamination", "mixture", "censoring",
    "incorrect_unit", "zero_duration", "negative_duration"),
  seed = 1L
)
```

Arguments

scenario	Pathological or stress-test duration scenario.
seed	Non-negative integer seed.

Value

A gp3bayes_pathological_simulation.

`simulate_hierarchical_binary_data`*Simulate Hierarchical Binary Data*

Description

Generates deterministic synthetic repeated-measures data from the approved Bernoulli-logit contract. The simulator records all generating parameters, participant effects, optional crossed item effects, and the random-number seed. It performs no model fitting.

Usage

```
simulate_hierarchical_binary_data(  
  n_participants = 40,  
  trials_per_participant = 20,  
  n_items = 20,  
  intercept = stats::qlogis(0.35),  
  condition_effect = 0.8,  
  participant_covariate_effect = 0.3,  
  trial_covariate_effect = 0.15,  
  interaction_effect = 0.25,  
  participant_sd = 0.7,  
  item_sd = 0.35,  
  random_slope_sd = 0.3,  
  random_slope_cor = 0,  
  condition_probability = 0.5,  
  balanced_condition = TRUE,  
  include_items = TRUE,  
  seed = 1  
)
```

Arguments

<code>n_participants</code>	Number of participants.
<code>trials_per_participant</code>	Number of observations per participant.
<code>n_items</code>	Number of crossed items when <code>include_items = TRUE</code> .
<code>intercept</code>	Population intercept on the log-odds scale.
<code>condition_effect</code>	Population condition contrast on the log-odds scale.
<code>participant_covariate_effect</code>	Participant-covariate coefficient.
<code>trial_covariate_effect</code>	Trial-covariate coefficient.
<code>interaction_effect</code>	Condition-by-participant-covariate coefficient.

participant_sd Participant random-intercept standard deviation.
 item_sd Crossed item random-intercept standard deviation.
 random_slope_sd Participant condition-slope standard deviation.
 random_slope_cor Correlation between participant intercepts and condition slopes. It must lie strictly between -1 and 1.
 condition_probability Treatment probability when balanced_condition = FALSE.
 balanced_condition Whether each participant receives an approximately balanced condition sequence.
 include_items Whether to generate a crossed item identifier.
 seed Non-negative integer random-number seed.

Details

The condition is generated using -0.5 and 0.5 internally and returned as a factor with levels control and treatment. The data-generating model includes one participant random intercept, one optional correlated participant condition slope, and one optional crossed item intercept.

Value

A gp3bayes_binary_simulation containing synthetic data, stored truth, generated random effects, and design metadata.

Examples

```

simulation <- simulate_hierarchical_binary_data(
  n_participants = 12,
  trials_per_participant = 8,
  n_items = 6,
  seed = 2026
)

simulation
head(simulation$data)

```

simulate_hierarchical_duration_data

Simulate Hierarchical Lognormal Duration Data

Description

Generates deterministic strictly positive uncensored durations from the approved hierarchical log-normal contract.

Usage

```
simulate_hierarchical_duration_data(
  n_participants = 40L,
  trials_per_participant = 20L,
  n_items = 20L,
  baseline_median = 500,
  condition_effect = log(1.15),
  participant_covariate_effect = log(1.08),
  trial_covariate_effect = log(1.04),
  interaction_effect = log(1.05),
  participant_sd = 0.35,
  item_sd = 0.2,
  random_slope_sd = 0.15,
  random_slope_cor = 0,
  residual_sd = 0.4,
  condition_probability = 0.5,
  balanced_condition = TRUE,
  include_items = TRUE,
  outcome_unit = "milliseconds",
  seed = 1L
)
```

Arguments

n_participants Number of participants.

trials_per_participant Observations per participant.

n_items Number of crossed items when `include_items = TRUE`.

baseline_median Baseline duration median in `outcome_unit`.

condition_effect Population condition contrast on the log-duration scale.

participant_covariate_effect Participant-covariate coefficient on the log-duration scale.

trial_covariate_effect Trial-covariate coefficient on the log-duration scale.

interaction_effect Condition-by-participant-covariate coefficient on the log-duration scale.

participant_sd Participant random-intercept standard deviation on the log scale.

item_sd Crossed item random-intercept standard deviation.

random_slope_sd Participant condition-slope standard deviation.

random_slope_cor Correlation between participant intercepts and condition slopes.

residual_sd Lognormal residual standard deviation.

condition_probability	Focal-condition probability for an unbalanced design.
balanced_condition	Whether each participant receives an approximately balanced condition sequence.
include_items	Whether crossed items are generated.
outcome_unit	Recorded duration unit.
seed	Non-negative integer random-number seed.

Details

The generated outcome is strictly positive, finite, and uncensored. The function does not generate zero, censored, truncated, shifted, or survival outcomes.

Value

A gp3bayes_duration_simulation containing synthetic data, stored truth, random effects, and design metadata.

simulate_pupil_response_shape	<i>Simulate data from the experimental nonlinear response-shape family</i>
-------------------------------	--

Description

Simulate data from the experimental nonlinear response-shape family

Usage

```
simulate_pupil_response_shape(
  n_participants = 20L,
  trials_per_participant = 6L,
  time_points = 41L,
  conditions = c("control", "treatment"),
  baseline = 3.2,
  amplitude = 0.7,
  onset = 250,
  rise = 180,
  duration = 1200,
  decay = 260,
  condition_amplitude_ratio = 1.2,
  condition_onset_shift = 80,
  residual_sd = 0.08,
  seed = 2026
)
```

Arguments

n_participants,	trials_per_participant,	time_points
	Simulation sizes.	
conditions	Condition labels.	
baseline,	amplitude,	onset,
rise,	duration,	decay
	Shape parameters.	
condition_amplitude_ratio		
	Multiplicative amplitude ratio for condition 2.	
condition_onset_shift		
	Onset shift for condition 2.	
residual_sd	Residual SD.	
seed	Seed.	

simulate_pupil_timecourse

Simulate deterministic hierarchical pupil time courses

Description

Generates event-aligned synthetic pupil data with participant/item heterogeneity, a smooth non-universal response waveform, AR(1) residual dependence, optional blink/data-loss segments, gaze drift, and luminance nuisance variation.

Usage

```
simulate_pupil_timecourse(
  n_participants = 20L,
  trials_per_participant = 12L,
  n_items = 12L,
  sampling_frequency = 60,
  time_window = c(-0.5, 2.5),
  baseline_window = c(-0.5, 0),
  conditions = c("control", "treatment"),
  baseline_pupil = 4,
  response_amplitude = 0.45,
  condition_difference = 0.18,
  peak_latency = 0.9,
  participant_sd = 0.25,
  item_sd = 0.08,
  residual_sd = 0.08,
  ar1 = 0.55,
  blink_trial_probability = 0.15,
  blink_duration = 0.12,
  include_gaze = TRUE,
  include_luminance = TRUE,
```

```

    gaze_drift_sd = 0.002,
    luminance_amplitude = 0.12,
    seed = 2026,
    max_rows = 500000L
)

```

Arguments

n_participants Number of participants.

trials_per_participant Number of trials per participant.

n_items Number of crossed items; use NULL for no item effect.

sampling_frequency Sampling frequency in Hz.

time_window Two-element event-relative time window in seconds.

baseline_window Two-element pre-event baseline window.

conditions Character vector of condition labels.

baseline_pupil Baseline pupil level in millimetres.

response_amplitude Peak amplitude of the common simulated response.

condition_difference Additional peak amplitude in the second condition. For more than two conditions it is multiplied by the zero-based condition index.

peak_latency Time of the simulated waveform peak in seconds.

participant_sd, item_sd Standard deviations for simulated hierarchy.

residual_sd Innovation standard deviation.

ar1 AR(1) residual coefficient with absolute value below one.

blink_trial_probability Probability that a trial contains one synthetic blink/data-loss interval.

blink_duration Blink interval duration in seconds.

include_gaze, include_luminance Whether to add nuisance signals.

gaze_drift_sd Standard deviation of gaze drift increments.

luminance_amplitude Amplitude of the synthetic luminance nuisance.

seed Reproducibility seed.

max_rows Maximum allowed output rows.

Value

A gp3bayes_pupil_simulation containing data and separate truth.

Interpretation

The waveform is a convenient synthetic data-generating shape, not a claim about a universal biological pupil response.

Examples

```
sim <- simulate_pupil_timecourse(
  n_participants = 4, trials_per_participant = 4,
  sampling_frequency = 20, seed = 2026
)
head(sim$data)
```

```
specify_advanced_pupil_timecourse_model
```

Specify an advanced governed pupil time-course model

Description

Builds the 0.5 advanced model contract without compiling or fitting Stan. The function is additive to the frozen 0.4 API: it consumes the same prepared pupil data but returns a distinct advanced specification.

Usage

```
specify_advanced_pupil_timecourse_model(
  prepared,
  temporal_structure = c("smooth", "linear", "gaussian_process"),
  family = c("gaussian", "student"),
  residual_scale = c("constant", "condition", "time", "condition_time"),
  distribution = NULL,
  smooth_basis_dimension = 10L,
  gp_spec = create_pupil_gp_spec(),
  condition_trajectory = NULL,
  autocorrelation = c("none", "ar1", "ar2", "arma11"),
  participant_trajectory = c("none", "factor_smooth"),
  item_effects = NULL,
  covariates = character(),
  measurement_model = NULL,
  missingness_model = NULL,
  prior_scales = NULL,
  predictive_target = c("new_trial_known_participant", "new_participant",
    "future_segment", "new_sample_known_trial"),
  allow_high_complexity = FALSE
)
```

Arguments

<code>prepared</code>	A prepared 0.4 pupil object or compatible data frame.
<code>temporal_structure</code>	"smooth", "linear", or "gaussian_process".
<code>family</code>	"gaussian" or robust "student".
<code>residual_scale</code>	Residual-scale model: constant, condition, time, or condition-by-time.
<code>distribution</code>	Optional object from specify_pupil_distribution() . When supplied, its family and residual-scale declarations override the corresponding scalar arguments.
<code>smooth_basis_dimension</code>	Basis dimension for smooth mean trajectories.
<code>gp_spec</code>	A GP configuration from create_pupil_gp_spec() .
<code>condition_trajectory</code>	Whether condition-specific trajectories are included. Defaults to TRUE when a condition column exists.
<code>autocorrelation</code>	One of "none", "ar1", "ar2", "arma11", or a bounded object from create_pupil_arma_spec() .
<code>participant_trajectory</code>	"none" or "factor_smooth".
<code>item_effects</code>	Include a random item intercept when an item column exists.
<code>covariates</code>	Additional declared covariates.
<code>measurement_model</code>	Optional known-uncertainty declaration.
<code>missingness_model</code>	Optional MAR-oriented missingness declaration.
<code>prior_scales</code>	Optional named numeric prior-scale overrides.
<code>predictive_target</code>	Declared prediction target inherited from the 0.4 validation vocabulary.
<code>allow_high_complexity</code>	Permit specifications flagged by the complexity audit. This is an explicit opt-in, not automatic model approval.

Value

A `gp3bayes_pupil_advanced_specification` object.

specify_binary_model *Specify a Backend-Independent Binary Model*

Description

Combines prepared binary data, a successful readiness audit, the restricted hierarchical formula, and validated family-specific priors. The returned object is not executable and performs no model fitting.

Usage

```
specify_binary_model(
  prepared,
  baseline = 0.5,
  intercept_scale = 1.5,
  coefficient_scale = 0.75,
  group_sd_scale = 1,
  correlation_eta = 2,
  student_df = 3
)
```

Arguments

<code>prepared</code>	A <code>gp3bayes_binary_prepared</code> object.
<code>baseline</code>	Plausible baseline event probability.
<code>intercept_scale</code>	Optional scale for the normal intercept prior.
<code>coefficient_scale</code>	Optional common scale for normal population-level coefficient priors, including the approved interaction.
<code>group_sd_scale</code>	Scale for half-Student-t group standard deviations.
<code>correlation_eta</code>	LKJ shape used when a random slope is requested.
<code>student_df</code>	Degrees of freedom for half-Student-t scale priors.

Details

The specification retains the prepared data because backend-independent prior predictive simulation must reproduce the declared design. It contains no backend object, posterior draws, or fitted model.

Value

A `gp3bayes_binary_model_specification` that also inherits from `gp3bayes_model_specification`.

Examples

```
simulation <- simulate_hierarchical_binary_data(
  n_participants = 12,
  trials_per_participant = 8,
  seed = 2026
)

contract <- create_model_contract(
  family = "binary",
  outcome_col = "selected",
  participant_col = "participant_id",
  item_col = "item_id",
  trial_col = "trial_id",
  condition_col = "condition"
)

prepared <- prepare_hierarchical_binary_data(
  simulation$data,
  contract,
  condition_levels = c("control", "treatment")
)

specification <- specify_binary_model(
  prepared,
  baseline = 0.35
)

specification
```

`specify_binary_model_with_interaction_prior`*Specify a Binary Model with a Separate Interaction Prior*

Description

Retains the approved binary model contract while recording a common main-effect prior scale and a narrower coefficient-specific interaction prior scale.

Usage

```
specify_binary_model_with_interaction_prior(
  prepared,
  baseline,
  intercept_scale = 1.5,
  main_effect_scale = 0.75,
  interaction_scale = 0.5,
  group_sd_scale = 1,
  correlation_eta = 2,
```



```

    student_df = 3
  )

```

Arguments

prepared	A gp3bayes_binary_prepared object containing one approved two-way interaction.
baseline	Plausible baseline event probability.
intercept_scale	Intercept-prior scale.
main_effect_scale	Common population main-effect prior scale.
interaction_scale	Interaction prior scale.
group_sd_scale	Group-level standard-deviation prior scale.
correlation_eta	LKJ shape.
student_df	Student-t degrees of freedom.

Value

A binary model specification with advanced prior metadata.

```
specify_binocular_pupil_model
```

Specify a joint binocular pupil model

Description

Specify a joint binocular pupil model

Usage

```

specify_binocular_pupil_model(
  prepared,
  temporal_structure = c("smooth", "linear", "gaussian_process"),
  family = c("gaussian", "student"),
  smooth_basis_dimension = 10L,
  gp_spec = create_pupil_gp_spec(),
  residual_correlation = TRUE,
  item_effects = NULL,
  prior_scales = NULL,
  allow_high_complexity = FALSE
)

```

Arguments

prepared	A binocular prepared object.
temporal_structure	Mean trajectory type.
family	Gaussian or Student-t for both eyes.
smooth_basis_dimension	Requested smooth basis dimension; when omitted, an unsupported default is conservatively reduced to observed temporal support, while explicitly unsupported values are rejected.
gp_spec	GP configuration when requested.
residual_correlation	Whether to estimate left/right residual correlation.
item_effects	Include item random intercepts when available.
prior_scales	Optional prior-scale overrides.
allow_high_complexity	Explicit computational opt-in for exact GP on large time grids.

Value

A gp3bayes_binocular_pupil_specification object.

specify_duration_model

Specify a Backend-Independent Duration Model

Description

Combines prepared positive-duration data with the approved hierarchical lognormal formula and explicit prior specification.

Usage

```
specify_duration_model(
  prepared,
  baseline,
  intercept_scale = 1,
  coefficient_scale = 0.5,
  group_sd_scale = 1,
  residual_scale = 1,
  correlation_eta = 2,
  student_df = 3
)
```

Arguments

prepared	A gp3bayes_duration_prepared.
baseline	Plausible baseline median in the prepared outcome unit.
intercept_scale	Positive normal-intercept prior scale.
coefficient_scale	Positive population-coefficient prior scale.
group_sd_scale	Positive group standard-deviation prior scale.
residual_scale	Positive lognormal residual-scale prior scale.
correlation_eta	LKJ shape for an approved random slope.
student_df	Degrees of freedom for half-Student-t scale priors.

Value

A gp3bayes_duration_model_specification.

specify_duration_model_with_interaction_prior

Specify a Duration Model with a Separate Interaction Prior

Description

Uses the candidate duration defaults of 0.35 for population main effects and 0.25 for the approved two-way interaction.

Usage

```
specify_duration_model_with_interaction_prior(
  prepared,
  baseline,
  intercept_scale = 1,
  main_effect_scale = 0.35,
  interaction_scale = 0.25,
  group_sd_scale = 0.5,
  residual_scale = 0.5,
  correlation_eta = 2,
  student_df = 3
)
```

Arguments

prepared	A gp3bayes_binary_prepared object containing one approved two-way interaction.
baseline	Plausible baseline median in the recorded outcome unit.
intercept_scale	Intercept-prior scale.
main_effect_scale	Common population main-effect prior scale.
interaction_scale	Interaction prior scale.
group_sd_scale	Group-level standard-deviation prior scale.
residual_scale	Lognormal residual-scale prior scale.
correlation_eta	LKJ shape.
student_df	Student-t degrees of freedom.

Value

A duration model specification with advanced prior metadata.

specify_pupil_distribution

Declare an advanced pupil observation distribution

Description

Declare an advanced pupil observation distribution

Usage

```
specify_pupil_distribution(
  family = c("gaussian", "student"),
  residual_scale = c("constant", "condition", "time", "condition_time")
)
```

Arguments

family	Gaussian or Student-t.
residual_scale	Constant, condition-, time-, or condition-by-time scale.

Value

A gp3bayes_pupil_distribution_spec object.

specify_pupil_response_shape_model

Specify the experimental nonlinear pupil response-shape model

Description

Uses a smooth asymmetric gated response: $\text{baseline} + \exp(\log_amplitude) * \text{logistic}((\text{time-onset})/\exp(\log_rise)) * \text{logistic}((\text{onset}+\exp(\log_duration)-\text{time})/\exp(\log_decay))$.

Usage

```
specify_pupil_response_shape_model(
  prepared,
  family = c("gaussian", "student"),
  condition_effects = c("amplitude", "onset", "duration"),
  participant_effects = c("baseline", "amplitude"),
  covariates = character(),
  prior_scales = NULL
)
```

Arguments

prepared	A prepared pupil object or compatible data frame.
family	Gaussian or Student-t.
condition_effects	Character subset of "amplitude", "onset", and "duration".
participant_effects	Character subset of "baseline" and "amplitude".
covariates	Additional covariates for baseline only.
prior_scales	Optional named positive numeric prior-scale overrides.

Details

Condition effects can enter log-amplitude, onset, and log-duration. This is a deliberately single, inspectable response family rather than an arbitrary nonlinear-formula interface.

Value

A gp3bayes_pupil_response_shape_specification object.

specify_pupil_timecourse_model

Specify the restricted hierarchical pupil time-course model

Description

Constructs an inspectable, closed-set Gaussian model specification. Users cannot supply a raw formula or arbitrary family.

Usage

```
specify_pupil_timecourse_model(
  prepared,
  temporal_structure = c("smooth", "linear"),
  smooth_basis_dimension = 10L,
  condition_trajectory = NULL,
  autocorrelation = c("ar1", "none"),
  participant_trajectory = c("none", "factor_smooth"),
  item_effects = NULL,
  covariates = character(),
  prior_scales = NULL
)
```

Arguments

prepared	A prepare_pupil_timecourse() result.
temporal_structure	"smooth" or "linear".
smooth_basis_dimension	Basis dimension for approved smooth terms.
condition_trajectory	NULL (default) uses a separate trajectory when condition is declared; otherwise supply TRUE or FALSE explicitly.
autocorrelation	"ar1" or "none". AR(1) is blocked when the observed sampling-interval coefficient of variation exceeds the recorded readiness threshold because sample-order AR(1) is not a continuous-time irregular-sampling model.
participant_trajectory	"none" or the restricted factor-smooth option "factor_smooth".
item_effects	NULL (default) includes an item random intercept only when at least two item levels are declared; otherwise supply TRUE or FALSE explicitly.
covariates	Character vector of already-declared numeric nuisance covariates in the prepared data.
prior_scales	Optional named positive scale values. Required for pixels and arbitrary units.

Value

A gp3bayes_pupil_model_specification.

Priors

Defaults are unit-aware weak regularizers for physical millimetres/metres and declared transformed scales. Pixel and arbitrary-unit outcomes require user-declared prior scales because tracker-specific units are not interchangeable.

Governance boundary

No unrestricted formula, likelihood family, smooth, autocorrelation order, or backend argument is accepted.

Examples

```
sim <- simulate_pupil_timecourse(
  n_participants = 3, trials_per_participant = 3,
  sampling_frequency = 20, seed = 2
)
contract <- create_pupil_contract(
  "pupil_mm", "participant_id", "trial_id", "event_time",
  "millimetres", 20, condition_col = "condition"
)
prepared <- prepare_pupil_timecourse(sim$data, contract)
specify_pupil_timecourse_model(prepared, autocorrelation = "none")
```

summarise_binary_group_variation

Summarise Binary Outcome Variation Within Groups

Description

Identifies participants or items whose observed binary outcomes are all zero or all one. Such groups are retained and reported; they are not deleted.

Usage

```
summarise_binary_group_variation(
  data,
  contract,
  group = c("participant", "item")
)
```

Arguments

data	A data frame.
contract	An approved binary model contract.
group	Either "participant" or "item".

Value

A gp3bayes_binary_group_variation object.

summarise_binary_posterior
<i>Summarise a Binary Posterior</i>

Description

Reports posterior location, uncertainty intervals, R-hat, effective sample sizes, probability of a positive coefficient, and odds-ratio transforms for population-level coefficients.

Usage

```
summarise_binary_posterior(fit, probability = 0.95, variables = NULL)
```

Arguments

- | | |
|-------------|--|
| fit | A gp3bayes_binary_fit. |
| probability | Central posterior interval probability. |
| variables | Optional supported posterior variable names. |

Details

Probability-positive values and intervals are descriptive posterior summaries. They are not frequentist significance tests and do not establish causal or substantive validity.

Value

A gp3bayes_binary_posterior_summary.

summarise_condition_balance
<i>Summarise Overall Condition Balance</i>

Description

Computes the observed proportion of each focal-condition level and applies explicit review and failure thresholds. The thresholds are workflow thresholds, not universal statistical laws.

Usage

```
summarise_condition_balance(
  data,
  contract,
  warning_fraction = 0.1,
  failure_fraction = 0.02
)
```

Arguments

data	A data frame.
contract	An approved model contract.
warning_fraction	Minimum condition fraction below which review is requested.
failure_fraction	Minimum condition fraction below which the strict readiness gate fails.

Value

A gp3bayes_condition_balance object.

```
summarise_duration_posterior
```

Summarise a Duration Posterior

Description

Reports posterior location, uncertainty, diagnostics, and multiplicative duration-ratio transforms for population-level coefficients.

Usage

```
summarise_duration_posterior(fit, probability = 0.95, variables = NULL)
```

Arguments

fit	A gp3bayes_duration_fit.
probability	Central posterior interval probability.
variables	Optional supported posterior variable names.

Details

Exponentiating a population-level coefficient gives its conditional multiplicative effect on the median duration under the approved lognormal model. This is not automatically a causal effect.

Value

A gp3bayes_duration_posterior_summary.

summarise_estimand_draws

Summarise Posterior Estimand Draws

Description

Summarise Posterior Estimand Draws

Usage

```
summarise_estimand_draws(x, quantities = NULL, probs = c(0.025, 0.5, 0.975))
```

Arguments

x	A gp3bayes_estimand or finite numeric vector.
quantities	Optional estimand-draw columns to summarise.
probs	Three probabilities defining lower, middle, and upper summaries.

Value

A data frame.

summarise_mcmc_quality

Summarise MCMC Quality Evidence

Description

Summarise MCMC Quality Evidence

Usage

```
summarise_mcmc_quality(fit, ...)
```

Arguments

fit	A fitted gp3bayes_fit.
...	Threshold arguments passed to identify_mcmc_issues().

Value

A gp3bayes_mcmc_quality object containing parameter and sampler diagnostic evidence. It is a review object, not an adequacy certificate.

`summarise_model_posterior`*Summarise an Approved gp3bayes Posterior*

Description

Family-neutral wrapper around `summarise_binary_posterior()` and `summarise_duration_posterior()`.

Usage

```
summarise_model_posterior(fit, ...)
```

Arguments

<code>fit</code>	A gp3bayes_fit.
<code>...</code>	Family-specific diagnostic arguments.

Value

A family-specific gp3bayes posterior summary.

`summarise_pupil_posterior`*Summarise a fitted pupil posterior*

Description

Returns posterior location, uncertainty, R-hat, and ESS evidence for model parameters without converting them into psychological constructs.

Usage

```
summarise_pupil_posterior(fit, probability = 0.95)
```

Arguments

<code>fit</code>	A fitted pupil model.
<code>probability</code>	Credible interval probability.

Value

A gp3bayes_pupil_posterior_summary.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

summarise_sbc_result	<i>Summarise an SBC Result</i>
----------------------	--------------------------------

Description

Summarise an SBC Result

Usage

```
summarise_sbc_result(x)
```

Arguments

x A gp3bayes_sbc_result.

Value

A conservative summary list.

summarise_sensitivity_suite	<i>Summarise a Sensitivity Suite</i>
-----------------------------	--------------------------------------

Description

Summarise a Sensitivity Suite

Usage

```
summarise_sensitivity_suite(x)
```

Arguments

x A gp3bayes_sensitivity_suite.

Value

A component-level data frame.

theme_gp3bayes	<i>gp3bayes Publication Theme</i>
----------------	-----------------------------------

Description

gp3bayes Publication Theme

Usage

```
theme_gp3bayes(base_size = 11, base_family = "")
```

Arguments

base_size	Base font size.
base_family	Base font family.

Value

A ggplot2 theme.

Examples

```
if (requireNamespace("ggplot2", quietly = TRUE)) theme_gp3bayes()
```

translate_advanced_pupil_model_to_brms	<i>Translate an advanced pupil model to brms without fitting</i>
--	--

Description

Translate an advanced pupil model to brms without fitting

Usage

```
translate_advanced_pupil_model_to_brms(specification)
```

Arguments

specification	An advanced pupil specification.
---------------	----------------------------------

Value

A gp3bayes_pupil_advanced_brms_specification object containing the brms formula, family, priors, and translated data.

```
translate_binary_model_to_brms
```

Translate a Binary Model Specification to brms

Description

Converts an approved backend-independent binary specification into a restricted brms representation. The formula, Bernoulli-logit family, and priors are derived entirely from the existing gp3bayes specification.

Usage

```
translate_binary_model_to_brms(specification)
```

Arguments

`specification` A gp3bayes_binary_model_specification.

Details

This function performs translation and prior validation only. It does not compile Stan code, run MCMC, create posterior draws, or assess convergence. Users cannot supply an alternative formula, family, backend, algorithm, or arbitrary backend arguments.

Value

A gp3bayes_binary_backend_specification containing the restricted formula, family, translated priors, validated prior table, and backend metadata.

Examples

```
if (requireNamespace("brms", quietly = TRUE)) {
  simulation <- simulate_hierarchical_binary_data(
    n_participants = 12,
    trials_per_participant = 8,
    n_items = 6,
    random_slope_sd = 0,
    seed = 2026
  )

  contract <- create_model_contract(
    family = "binary",
    outcome_col = "selected",
    participant_col = "participant_id",
    item_col = "item_id",
    trial_col = "trial_id",
    condition_col = "condition"
  )
}
```

```
prepared <- prepare_hierarchical_binary_data(  
  simulation$data,  
  contract,  
  condition_levels = c("control", "treatment")  
)  
  
specification <- specify_binary_model(  
  prepared,  
  baseline = 0.35  
)  
  
translate_binary_model_to_brms(specification)  
}
```

translate_binary_model_with_interaction_prior

Translate an Advanced Binary Specification to brms

Description

Translate an Advanced Binary Specification to brms

Usage

```
translate_binary_model_with_interaction_prior(specification)
```

Arguments

specification An advanced binary specification.

Value

A validated restricted brms translation.

translate_binocular_pupil_model_to_brms

Translate a binocular specification to a brms multivariate formula

Description

Translate a binocular specification to a brms multivariate formula

Usage

```
translate_binocular_pupil_model_to_brms(specification)
```

Arguments

specification A binocular specification.

Value

A gp3bayes_binocular_brms_specification object.

translate_duration_model_to_brms

Translate a Duration Model Specification to brms

Description

Converts an approved backend-independent duration specification into a fixed hierarchical lognormal brms representation.

Usage

```
translate_duration_model_to_brms(specification)
```

Arguments

specification A gp3bayes_duration_model_specification.

Details

Translation validates the formula and priors but does not compile Stan code or fit a model. Users cannot supply an alternative family, formula, backend, algorithm, Stan extension, or arbitrary backend arguments.

Value

A gp3bayes_duration_backend_specification.

translate_duration_model_with_interaction_prior

Translate an Advanced Duration Specification to brms

Description

Translate an Advanced Duration Specification to brms

Usage

```
translate_duration_model_with_interaction_prior(specification)
```


Arguments

specification An advanced duration specification.

Value

A validated restricted brms translation.

```
translate_pupil_model_to_brms
```

Translate an approved pupil model to brms

Description

Creates a fixed Gaussian brms representation without compiling or fitting.

Usage

```
translate_pupil_model_to_brms(specification)
```

Arguments

specification A pupil model specification.

Value

A gp3bayes_pupil_brms_translation.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

```
translate_pupil_response_shape_to_brms
```

Translate the experimental response-shape model to brms

Description

Translate the experimental response-shape model to brms

Usage

```
translate_pupil_response_shape_to_brms(specification)
```

Arguments

specification A response-shape specification.

Value

A backend-independent brms translation object.

```
validate_analysis_manifest
```

Validate an Analysis Manifest

Description

Validate an Analysis Manifest

Usage

```
validate_analysis_manifest(manifest, strict = FALSE)
```

Arguments

manifest	A gp3bayes_analysis_manifest.
strict	Whether failures should raise an error.

Value

A gp3bayes_manifest_validation.

```
validate_backend_environment
```

Validate a Bayesian Backend Environment

Description

Checks whether the packages and external runtime required by one approved gp3bayes backend are available. With `compile_test = TRUE`, an optional minimal compiler smoke test is performed. The smoke test does not fit a statistical model and writes no persistent files.

Usage

```
validate_backend_environment(
  backend = c("rstan", "cmdstanr"),
  compile_test = FALSE,
  strict = FALSE
)
```

Arguments

backend	Either "rstan" or "cmdstanr".
compile_test	Whether to run an optional compiler smoke test.
strict	Whether an unavailable backend should raise an error.

Value

A gp3bayes_backend_environment object.

Examples

```
validate_backend_environment("rstan")
validate_backend_environment("cmdstanr")
```

```
validate_gp3bayes_object
```

Validate a gp3bayes Object

Description

Performs lightweight structural validation for gp3bayes contracts, prepared data, specifications, fits, summaries, diagnostics, manifests, design audits, sensitivity suites, evidence collections, and backend reliability objects. The function checks object structure only; it does not establish statistical adequacy or substantive validity.

Usage

```
validate_gp3bayes_object(x, recursive = TRUE, strict = FALSE)
```

Arguments

x	A gp3bayes object.
recursive	Whether nested contract/specification/prepared objects should also be checked when present.
strict	Whether a failed structural check should raise an error.

Value

A gp3bayes_object_validation object.

`validate_gp3bayes_schema`*Validate an Object Against a Frozen gp3bayes Schema*

Description

Validate an Object Against a Frozen gp3bayes Schema

Usage

```
validate_gp3bayes_schema(x, schema, strict = FALSE, compare_lengths = FALSE)
```

Arguments

<code>x</code>	A gp3bayes object.
<code>schema</code>	A gp3bayes_object_schema.
<code>strict</code>	Whether structural drift should raise an error.
<code>compare_lengths</code>	Whether analysis-specific object lengths should be required to match the frozen schema.

Value

A gp3bayes_schema_validation object.

`validate_prior_specification`*Validate a Prior Specification*

Description

Validates the completeness and internal consistency of a gp3bayes_prior_specification.

Usage

```
validate_prior_specification(priors, contract = NULL)
```

Arguments

<code>priors</code>	A gp3bayes_prior_specification created by create_prior_specification() .
<code>contract</code>	Optional gp3bayes_model_contract.

Value

priors, invisibly.

validate_publication_registry
<i>Validate a Publication Registry</i>

Description

Validate a Publication Registry

Usage

```
validate_publication_registry(x)
```

Arguments

x A publication registry.

Value

A validation object.

validate_pupil_leave_future_out
<i>Execute or materialize leave-future-out validation</i>

Description

Execute or materialize leave-future-out validation

Usage

```
validate_pupil_leave_future_out(  
  fit,  
  plan,  
  execute = FALSE,  
  cores = 1L,  
  seed = 2026  
)
```

Arguments

fit	An advanced fitted model.
plan	An LFO plan.
execute	If FALSE, returns the plan without refitting. TRUE performs sequential model refits and future-block log scoring.
cores	Maximum cores passed to brms update; restricted to 2.
seed	Base seed for refits.

Value

A gp3bayes_pupil_lfo_validation object.

validate_pupil_model	<i>Validate a pupil model for an explicit prediction target</i>
----------------------	---

Description

Executes exact target-specific K-fold through `brms::kfold()` for K-fold targets, or a finite leave-future-segment refit for the future target. Execution is opt-in because it can be computationally expensive.

Usage

```
validate_pupil_model(
  fit,
  plan,
  execute = FALSE,
  ndraws = 200L,
  max_cells = 3000000L
)
```

Arguments

<code>fit</code>	A fitted pupil model.
<code>plan</code>	A pupil validation plan.
<code>execute</code>	Whether to execute refitting. FALSE returns validated partition evidence without refitting.
<code>ndraws</code>	Draws retained for finite future-segment prediction scoring.
<code>max_cells</code>	Memory guard for future-segment predictions.

Value

A gp3bayes_pupil_validation.

Examples

```
# See vignette("bayesian-dynamic-pupillometry", package = "gp3bayes") for a complete workflow.
```

validate_transformation_replay
<i>Validate Exact Transformation Replay</i>

Description

Round-trips the prepared data through the recorded inverse and forward transformation and compares transformed columns and fixed-effect matrices.

Usage

```
validate_transformation_replay(prepared, tolerance = 1e-10)
```

Arguments

prepared	A binary or duration prepared object.
tolerance	Numeric comparison tolerance.

Value

A gp3bayes_transformation_replay_audit.

variance_component_table
<i>Variance-Component Posterior Table</i>

Description

Variance-Component Posterior Table

Usage

```
variance_component_table(fit, probs = c(0.025, 0.5, 0.975))
```

Arguments

fit	A fitted gp3bayes_fit.
probs	Three posterior interval probabilities.

Value

A posterior summary table for group SDs, correlations, and residual scale where applicable.

write_analysis_bundle_report
<i>Write an Analysis-Bundle Markdown Report</i>

Description

Write an Analysis-Bundle Markdown Report

Usage

```
write_analysis_bundle_report(x, file)
```

Arguments

- | | |
|------|-----------------------------|
| x | A gp3bayes_analysis_bundle. |
| file | Explicit output file path. |

Value

Invisibly, the normalized output path.

write_diagnostic_dashboard_report
<i>Write a Diagnostic Dashboard Report</i>

Description

Write a Diagnostic Dashboard Report

Usage

```
write_diagnostic_dashboard_report(x, file, overwrite = FALSE)
```

Arguments

- | | |
|-----------|---|
| x | A diagnostic dashboard. |
| file | Explicit Markdown output path. |
| overwrite | Whether an existing file may be replaced. |

Value

Invisibly, normalized output path.

write_model_card	<i>Write a Model Card</i>
------------------	---------------------------

Description

Write a Model Card

Usage

```
write_model_card(x, file, overwrite = FALSE)
```

Arguments

- x A gp3bayes_model_card.
- file Explicit Markdown output path.
- overwrite Whether an existing file may be replaced.

Value

Invisibly, the normalized written path.

write_publication_registry	<i>Write a Publication Registry</i>
----------------------------	-------------------------------------

Description

Write a Publication Registry

Usage

```
write_publication_registry(x, file, overwrite = FALSE)
```

Arguments

- x A publication registry.
- file Explicit Markdown output path.
- overwrite Whether an existing file may be replaced.

Value

Invisibly, the normalized output path.

`write_reproducibility_report`*Write a Reproducibility Report*

Description

Writes a conservative Markdown provenance report to an explicit path.

Usage

```
write_reproducibility_report(manifest, file, overwrite = FALSE)
```

Arguments

<code>manifest</code>	An analysis manifest.
<code>file</code>	Explicit Markdown output path.
<code>overwrite</code>	Whether an existing file may be replaced.

Value

The normalized output path, invisibly.

Index

advanced_pupil_trajectory_table, 12
analysis_bundle_table, 12
analysis_manifest_table, 13
apply_transformation_recipe, 13
as.data.frame.gp3bayes_pupil_contract,
14
as.data.frame.gp3bayes_pupil_estimand,
14
as_pupil_prediction_draws, 17
assess_binary_prior_sensitivity, 15
assess_binary_prior_sensitivity(), 63
assess_duration_prior_sensitivity, 16
assess_duration_prior_sensitivity(),
69
assess_powerscaled_sensitivity, 17
audit_advanced_pupil_identifiability,
18
audit_backend_parity, 19
audit_binocular_pupil_readiness, 20
audit_design_support, 20
audit_design_support(), 224
audit_duration_boundaries, 21
audit_duration_unit_invariance, 22
audit_duration_unit_invariance(), 94
audit_estimand_invariance, 22
audit_fixed_effect_design, 23
audit_missingness_structure, 24
audit_model_readiness, 25
audit_model_readiness(), 26
audit_model_readiness_strict, 26
audit_model_readiness_strict(), 21
audit_prediction_support, 27
audit_pupil_computational_budget, 28
audit_pupil_measurement_context, 28
audit_pupil_measurement_model, 29
audit_pupil_missingness, 29
audit_pupil_predictive_calibration, 30
audit_pupil_readiness, 30
audit_pupil_temporal_dependence, 31
audit_random_effects_support, 32
backend_capabilities, 32
backend_environment_table, 33
backend_parity_table, 33
bayesian_backend_capabilities, 34
binary_calibration_error, 34
binary_calibration_table, 35
binary_calibration_uncertainty, 35
binary_calibration_uncertainty_table,
36
binary_confusion_table, 37
binary_group_calibration, 37
binary_precision_recall_curve, 38
binary_prediction_scores, 38
binary_roc_curve, 39
binary_threshold_metrics, 40
build_model_formula, 40
capture_gp3bayes_schema, 41
check_advanced_pupil_prior_predictive,
42
check_binary_posterior_predictive, 43
check_binary_posterior_predictive(),
63
check_binary_ppc_details, 44
check_binary_prior_predictive, 44
check_cmdstan_backend, 46
check_duration_posterior_predictive,
47
check_duration_posterior_predictive(),
69
check_duration_ppc_details, 48
check_duration_prior_predictive, 48
check_model_ppc, 49
check_pupil_posterior_predictive, 50
check_pupil_prior_predictive, 51
collect_model_evidence, 52
compare_analysis_manifests, 53
compare_estimand_sensitivity, 53

- compare_gp3bayes_schemas, 54
- compare_psis_loo, 54
- compare_pupil_autocorrelation, 55
- compare_pupil_lfo, 55
- compare_pupil_models, 56
- compare_pupil_sensitivity_estimands, 56
- compute_kfold_cv, 57
- compute_loo_model_weights, 58
- compute_psis_loo, 58
- compute_psis_loo_from_log_lik, 59
- create_advanced_pupil_prior_specification, 59
- create_analysis_bundle, 60
- create_analysis_figure_set, 61
- create_analysis_manifest, 61
- create_binary_model_report, 63
- create_brms_sbc_plan, 64
- create_complete_evidence_inventory, 65
- create_contrast_coding_sensitivity_specification, 65
- create_custom_sbc_plan, 66
- create_diagnostic_dashboard, 67
- create_diagnostic_dashboard_figures, 68
- create_duration_model_report, 68
- create_duration_unit_sensitivity_specification, 69
- create_figure_set, 70
- create_group_deletion_sensitivity_plan, 70
- create_group_deletion_sensitivity_plan(), 94
- create_loo_influence_atlas, 71
- create_model_card, 72
- create_model_contract, 72
- create_model_contract(), 25, 40, 41, 81
- create_model_evidence_report, 74
- create_model_specification, 75
- create_prediction_contrast_profile, 75
- create_prediction_grid, 76
- create_prediction_profile, 77
- create_prediction_surface, 78
- create_predictive_distribution_atlas, 79
- create_predictor_scaling_sensitivity_specification, 80
- create_prior_specification, 81
- create_prior_specification(), 300
- create_publication_registry, 82
- create_publication_table_set, 83
- create_pupil_advanced_sensitivity_suite, 83
- create_pupil_arma_spec, 84
- create_pupil_arma_spec(), 278
- create_pupil_contract, 84
- create_pupil_gp_spec, 87
- create_pupil_gp_spec(), 278
- create_pupil_lfo_plan, 88
- create_pupil_lfo_plan(), 56
- create_pupil_measurement_model, 88
- create_pupil_missingness_spec, 89
- create_pupil_model_set, 90
- create_pupil_sensitivity_suite, 90
- create_pupil_validation_plan, 92
- create_random_slope_sensitivity_plan, 93
- create_random_slope_sensitivity_plan(), 94
- create_reporting_checklist, 93
- create_sensitivity_suite_plan, 94
- create_sensitivity_suite_plan(), 262
- create_transformation_recipe, 95
- design_support_table, 95
- detect_binary_separation, 96
- detect_binary_separation(), 21
- diagnose_advanced_pupil_fit, 96
- diagnose_binary_fit, 97
- diagnose_binary_fit(), 63, 99
- diagnose_duration_fit, 98
- diagnose_duration_fit(), 69, 99
- diagnose_model_fit, 99
- diagnose_pupil_fit, 99
- diagnostic_dashboard_table, 100
- duration_fit_table, 100
- duration_prediction_scores, 101
- duration_qq_table, 101
- duration_quantile_calibration, 102
- duration_tail_check, 102
- estimand_sensitivity_table, 103
- estimate_binocular_pupil_trajectory, 103
- estimate_model_estimands, 104
- estimate_model_estimands(), 262
- estimate_pupil_auc, 104

- estimate_pupil_dynamic_contrast, 105
- estimate_pupil_peak, 106
- estimate_pupil_peak_latency, 106
- estimate_pupil_residual_scale, 107
- estimate_pupil_response_parameters, 108
- estimate_pupil_threshold_duration, 108
- estimate_pupil_trajectory, 109
- estimate_pupil_trajectory_derivative, 110
- estimate_pupil_window, 110
- estimate_standardized_duration_estimands, 111
- estimate_standardized_probability_contrast, 112
- evaluate_pathological_simulation, 112
- evidence_inventory_table, 113
- extract_expected_predictions, 113
- extract_linear_predictions, 114
- extract_log_likelihood, 115
- extract_posterior_draws, 115
- extract_posterior_predictions, 116
- extract_sampler_diagnostics, 117

- fit_advanced_pupil_model, 117
- fit_advanced_pupil_model_backend, 118
- fit_advanced_pupil_model_cmdstanr, 118
- fit_binary_model, 119
- fit_binary_model(), 15
- fit_binary_model_backend, 121
- fit_binary_model_cmdstanr (fit_binary_model_backend), 121
- fit_binocular_pupil_model, 122
- fit_duration_model, 123
- fit_duration_model(), 16
- fit_duration_model_backend, 124
- fit_duration_model_cmdstanr (fit_duration_model_backend), 124
- fit_pupil_model, 125
- fit_pupil_model_backend, 126
- fit_pupil_model_cmdstanr, 127
- fit_pupil_response_shape_model, 128
- freeze_analysis_manifest, 128
- freeze_gp3bayes_schema, 129

- gazeport_pupil_mapping_table, 129
- gp3bayes_specification_traceability, 130

- group_deletion_sensitivity_table, 131
- group_effect_draws_table, 131
- group_effect_rank_probability_table, 132
- group_effect_table, 133
- group_prediction_summary, 134
- grouped_prediction_check, 130

- identify_identifier_like_predictors, 134
- identify_loo_influential_observations, 135
- identify_mcmc_issues, 136
- inspect_gazeport_pupil_schema, 137
- interaction_prior_summary, 138
- invert_transformation_recipe, 138

- loo_diagnostic_table, 139
- loo_flagged_data, 139
- loo_group_influence_table, 140
- loo_influence_atlas_table, 140
- loo_influence_summary, 141
- loo_pointwise_table, 141
- loo_summary_table, 142

- manifest_comparison_table, 142
- materialize_pupil_advanced_sensitivity_scenario, 143
- materialize_pupil_sensitivity_scenario, 143
- mcmc_diagnostic_table, 144
- missingness_audit_table, 144
- model_card_table, 145
- model_comparison_table, 145
- model_evidence_table, 146
- model_weights_table, 146
- model_workflow_status, 147

- plot_gp3bayes_kfold_cv, 147
- plot_gp3bayes_strict_readiness_audit, 148
- plot_advanced_pupil_simulation, 148
- plot_advanced_pupil_trajectory, 149
- plot_autocorrelation, 149
- plot_backend_environment_gg, 150
- plot_backend_parity_gg, 150
- plot_binary_calibration, 151
- plot_binary_calibration_uncertainty, 151

- plot_binary_group_calibration, 152
- plot_binary_precision_recall, 152
- plot_binary_roc, 153
- plot_binary_threshold_metrics, 153
- plot_binocular_pupil_trajectory, 154
- plot_design_support_gg, 154
- plot_diagnostic_dashboard, 155
- plot_duration_fit, 155
- plot_duration_qq, 156
- plot_duration_quantile_calibration, 156
- plot_duration_tail, 157
- plot_estimand_intervals, 157
- plot_estimand_sensitivity_gg, 158
- plot_exceedance_probability, 158
- plot_group_deletion_sensitivity, 159
- plot_group_effect_distribution, 160
- plot_group_effect_rank_probability, 161
- plot_group_effects, 160
- plot_group_predictions, 161
- plot_grouped_prediction_check, 159
- plot_loo_group_elpd, 162
- plot_loo_group_influence, 162
- plot_loo_influence, 163
- plot_loo_influence_rank, 163
- plot_loo_pareto_vs_elpd, 164
- plot_loo_pointwise_elpd, 164
- plot_manifest_comparison_gg, 165
- plot_mcmc_quality, 165
- plot_missingness_gg, 166
- plot_model_comparison, 166
- plot_model_evidence_gg, 167
- plot_model_weights, 167
- plot_posterior_areas, 168
- plot_posterior_correlations, 168
- plot_posterior_density, 169
- plot_posterior_intervals, 169
- plot_posterior_pairs, 170
- plot_powerscale_sensitivity_gg, 171
- plot_ppc_statistic, 171
- plot_prediction_contrast_profile, 172
- plot_prediction_draws, 172
- plot_prediction_gradient, 173
- plot_prediction_interval_width, 174
- plot_prediction_intervals, 173
- plot_prediction_profile, 174
- plot_prediction_rank_probabilities, 175
- plot_prediction_score_uncertainty, 175
- plot_prediction_support, 176
- plot_prediction_surface, 176
- plot_prediction_surface_uncertainty, 177
- plot_predictive_atlas_statistics, 177
- plot_predictive_coverage, 178
- plot_predictive_quantile_envelope, 178
- plot_predictive_residuals, 179
- plot_prior_posterior_contraction, 179
- plot_prior_posterior_density, 180
- plot_prior_posterior_intervals, 180
- plot_prior_posterior_shift, 181
- plot_prior_sensitivity, 181
- plot_prior_sensitivity_scenarios, 182
- plot_pupil_autocorrelation_comparison, 182
- plot_pupil_dynamic_contrast, 183
- plot_pupil_estimand, 183
- plot_pupil_gp_hyperparameters, 184
- plot_pupil_identifiability_audit, 184
- plot_pupil_lfo, 185
- plot_pupil_measurement_audit, 185
- plot_pupil_measurement_uncertainty, 186
- plot_pupil_missingness, 186
- plot_pupil_model_comparison, 187
- plot_pupil_model_complexity, 187
- plot_pupil_observed_trajectory, 188
- plot_pupil_posterior_trajectory, 188
- plot_pupil_ppc, 189
- plot_pupil_predictive_calibration, 189
- plot_pupil_readiness, 190
- plot_pupil_residual_acf, 190
- plot_pupil_residual_scale, 191
- plot_pupil_residual_spectrum, 191
- plot_pupil_response_parameters, 192
- plot_pupil_sensitivity, 192
- plot_pupil_temporal_dependence, 193
- plot_pupil_trajectory_derivative, 193
- plot_pupil_validation, 194
- plot_random_intercept_variance_partition, 194
- plot_random_slope_sensitivity, 195
- plot_rank_diagnostics, 195
- plot_recovery_bias, 196
- plot_recovery_coverage, 196

- plot_recovery_estimates, 197
- plot_recovery_fit_status, 197
- plot_recovery_rmse, 198
- plot_reporting_checklist, 198
- plot_sampler_diagnostics, 199
- plot_sampling_diagnostics, 199
- plot_sbc_coverage_gg, 200
- plot_sbc_ecdf_gg, 200
- plot_sbc_rank_gg, 201
- plot_sbc_simulated_vs_estimated_gg, 201
- plot_schema_comparison_gg, 202
- plot_sensitivity_suite_gg, 202
- plot_uncertainty_decomposition, 203
- plot_variance_components, 203
- posterior_correlation_table, 204
- posterior_interval_table, 204
- posterior_predictive_statistic, 205
- posterior_predictive_summary_table, 206
- posterior_probability_table, 206
- powerscale_sensitivity_table, 207
- powerscale_sequence_for_fit, 207
- ppc_statistic_table, 208
- predict_advanced_pupil_trajectory, 220
- predict_binary_probability, 221
- predict_duration, 221
- predict_model, 222
- predict_pupil_trajectory, 223
- prediction_contrast, 208
- prediction_contrast_profile_table, 209
- prediction_draws_long, 210
- prediction_exceedance_probability, 210
- prediction_gradient_table, 211
- prediction_interval_width, 211
- prediction_pairwise_contrasts, 212
- prediction_profile_table, 212
- prediction_rank_probabilities, 213
- prediction_score_uncertainty, 213
- prediction_score_uncertainty_table, 214
- prediction_support_table, 215
- prediction_surface_table, 215
- prediction_table, 216
- prediction_uncertainty_decomposition, 216
- predictive_coverage_table, 217
- predictive_distribution_atlas_table, 218
- predictive_quantile_envelope, 218
- predictive_residuals, 219
- preflight_model_specification, 224
- prepare_binocular_pupil_timecourse, 225
- prepare_hierarchical_binary_data, 226
- prepare_hierarchical_duration_data, 227
- prepare_pupil_timecourse, 228
- prior_posterior_bridge, 230
- prior_posterior_distance_table, 231
- prior_posterior_draws_long, 231
- prior_posterior_summary_table, 232
- prior_sensitivity_scenario_table, 232
- prior_sensitivity_table, 233
- prior_specification_table, 233
- publication_registry_table, 234
- pupil_advanced_capabilities, 234
- pupil_advanced_compatibility_table, 235
- pupil_advanced_mapping_table, 235
- pupil_advanced_specification_table, 236
- pupil_autocorrelation_table, 236
- pupil_binocular_agreement_table, 237
- pupil_binocular_correlation, 237
- pupil_binocular_difference, 238
- pupil_condition_contrast, 238
- pupil_distribution_table, 239
- pupil_dynamic_contrast_table, 239
- pupil_gp_hyperparameters, 240
- pupil_gp_table, 240
- pupil_identifiability_table, 241
- pupil_measurement_audit_table, 241
- pupil_measurement_uncertainty_table, 242
- pupil_missingness_table, 242
- pupil_model_card, 243
- pupil_model_card_table, 243
- pupil_model_comparison_table, 244
- pupil_model_weights, 244
- pupil_ppc_table, 245
- pupil_readiness_table, 245
- pupil_residual_acf, 246
- pupil_residual_scale_table, 246
- pupil_residual_spectrum, 247
- pupil_response_parameter_table, 247

- pupil_sensitivity_table, 248
- pupil_specification_table, 248
- pupil_trajectory_derivative_table, 249
- pupil_trajectory_table, 249
- pupil_validation_table, 250
- random_intercept_variance_partition, 250
- random_intercept_variance_partition_table, 251
- random_slope_sensitivity_table, 251
- read_analysis_manifest, 252
- read_gp3bayes_schema, 252
- recovery_estimate_table, 253
- recovery_fit_status_table, 253
- recovery_parameter_table, 254
- register_publication_figure, 254
- register_publication_table, 255
- review_duration_extremes, 255
- run_binary_recovery, 256
- run_binary_recovery(), 63
- run_duration_recovery, 257
- run_duration_recovery(), 69
- run_group_deletion_sensitivity, 259
- run_random_slope_sensitivity, 260
- run_sbc_plan, 261
- run_sensitivity_suite, 261
- sampler_diagnostic_table, 262
- save_figure_set, 263
- save_publication_registry_figures, 263
- sbc_overview_table, 264
- sbc_stats_table, 265
- schema_comparison_table, 265
- score_pupil_predictions, 266
- sensitivity_suite_table, 266
- simulate_advanced_pupil_timecourse, 267
- simulate_advanced_pupil_timecourse(), 269
- simulate_binary_pathology, 268
- simulate_binocular_pupil_timecourse, 269
- simulate_declared_prior_draws, 269
- simulate_duration_pathology, 270
- simulate_hierarchical_binary_data, 271
- simulate_hierarchical_duration_data, 272
- simulate_pupil_response_shape, 274
- simulate_pupil_timecourse, 275
- specify_advanced_pupil_timecourse_model, 277
- specify_advanced_pupil_timecourse_model(), 60
- specify_binary_model, 279
- specify_binary_model_with_interaction_prior, 280
- specify_binocular_pupil_model, 281
- specify_duration_model, 282
- specify_duration_model_with_interaction_prior, 283
- specify_pupil_distribution, 284
- specify_pupil_distribution(), 278
- specify_pupil_response_shape_model, 285
- specify_pupil_timecourse_model, 286
- summarise_binary_group_variation, 287
- summarise_binary_posterior, 288
- summarise_binary_posterior(), 63, 291
- summarise_condition_balance, 288
- summarise_duration_posterior, 289
- summarise_duration_posterior(), 69, 291
- summarise_estimand_draws, 290
- summarise_mcmc_quality, 290
- summarise_model_posterior, 291
- summarise_pupil_posterior, 291
- summarise_sbc_result, 292
- summarise_sensitivity_suite, 292
- theme_gp3bayes, 293
- translate_advanced_pupil_model_to_brms, 293
- translate_binary_model_to_brms, 294
- translate_binary_model_with_interaction_prior, 295
- translate_binocular_pupil_model_to_brms, 295
- translate_duration_model_to_brms, 296
- translate_duration_model_with_interaction_prior, 296
- translate_pupil_model_to_brms, 297
- translate_pupil_response_shape_to_brms, 297
- validate_analysis_manifest, 298
- validate_backend_environment, 298
- validate_gp3bayes_object, 299
- validate_gp3bayes_schema, 300

`validate_prior_specification`, [300](#)
`validate_publication_registry`, [301](#)
`validate_pupil_leave_future_out`, [301](#)
`validate_pupil_leave_future_out()`, [56](#)
`validate_pupil_model`, [302](#)
`validate_transformation_replay`, [303](#)
`variance_component_table`, [303](#)

`write_analysis_bundle_report`, [304](#)
`write_diagnostic_dashboard_report`, [304](#)
`write_model_card`, [305](#)
`write_publication_registry`, [305](#)
`write_reproducibility_report`, [306](#)