Package: Ritable (via r-universe)

October 26, 2026

Title Pedigree-Based Heritability Estimation for Family Cohort Studies

Version 0.2.1

Description Provides profile-likelihood and general maximum-likelihood variance-components estimation of narrow-sense heritability (h2), household/common-environment effects (c2), and bivariate genetic and environmental correlations for quantitative traits in family cohort and twin studies. Additive genetic relationship matrices are built from pedigrees via 'kinship2', including monozygotic-twin relatedness overrides. Phenotypes are inverse-normal transformed internally. Likelihood-ratio tests use a one-sided chi-squared boundary correction equivalent to SOLAR Eclipse. Ninety-five percent confidence intervals are derived from the profile likelihood rather than Wald approximations. Batch estimation over many traits or trait pairs returns tidy data frames ready for downstream visualisation (forest plots, heatmaps) and includes Benjamini-Hochberg FDR correction for bivariate correlation batches.

License MIT + file LICENSE

URL https://r-itable.circadia-lab.uk, https://github.com/circadia-bio/R-itable

BugReports https://github.com/circadia-bio/R-itable/issues

Encoding UTF-8

Roxygen list(markdown = TRUE)

Depends R (>= 4.1.0)

Imports kinship2, stats, utils, rlang, cli

Suggests ggplot2, scales, pkgdown, testthat (>= 3.0.0), knitr, rmarkdown

Config/testthat/edition 3

VignetteBuilder knitr

Config/roxygen2/version 8.0.0

Repository https://circadia-bio.r-universe.dev

Date/Publication 2026-08-27 00:54:08 UTC

RemoteUrl https://github.com/circadia-bio/R-itable

RemoteRef v0.2.1

RemoteSha 7392a002f82b77bfaaf7c8f78b771fb00a54d9f0

Contents

build_grm 2
build_household 4
herit_ace 5
herit_batch 7
herit_bivar 8
herit_bivar_batch 10
herit_vc 11
int_transform 13
plot_forest 15
Ritable_colours 16

Index **17**

`build_grm` *Build an additive genetic relationship matrix from a pedigree*

Description

Constructs the additive genetic relationship matrix **A** (= 2 x kinship matrix) for a set of study subjects, using a pedigree that may include additional founder parents not in the study sample.

Usage

```
build_grm(
  ped_df,
  study_ids = NULL,
  id_col = "id",
  pat_col = "pat",
  mom_col = "mom",
  sex_col = "sex",
  mz_col = NULL
)
```

Arguments

ped_df	A data frame with at least the columns id, pat, mom, and sex. Missing parents should be NA or 0. sex should be numeric: 1 = male, 2 = female (any other value is recoded to 1 with a warning).
study_ids	Character or integer vector of IDs for the study subjects whose sub-matrix should be extracted. Defaults to all IDs in ped_df.
id_col, pat_col, mom_col, sex_col	Column names for the four required pedigree fields. Defaults are "id", "pat", "mom", "sex".
mz_col	Optional column name in ped_df identifying monozygotic twin/multiple groups (e.g. a shared family label present only for MZ sets, blank/NA for everyone else, as in a MZTWIN column). All pairwise combinations of individuals sharing the same non-blank, non-NA value of mz_col are passed to kinship2::pedigree() as monozygotic (code = 1), overriding the default full-sibling kinship of 0.25 with the correct MZ kinship of 0.5 (A = 1.0). Groups of size ≥ 3 (e.g. MZ triplets) are supported: every pair within the group is marked. NULL (default) disables this and reproduces prior behaviour exactly.

Details

The function calls kinship2::kinship() on the full pedigree (including founders), then multiplies by 2 to obtain the additive relationship matrix, and finally subsets to study_ids. Including founders in the pedigree ensures that kinship coefficients between study subjects connected only through founders are estimated correctly.

For twin cohorts, pat/mom are typically synthetic per-pair founder labels rather than genotyped individuals; such founders should still be present as their own rows in ped_df (with pat/mom set to NA) so that kinship2::pedigree() can resolve the family structure. Without an mz_col, twins are treated as ordinary full siblings (A = 0.5); supplying mz_col corrects MZ pairs to A = 1.0 while leaving DZ pairs (which already have the correct full-sibling kinship) unchanged.

Value

A symmetric numeric matrix of dimension length(study_ids) x length(study_ids), with diagonal entries equal to 1 for non-inbred individuals and row/column names matching study_ids.

Examples

```
# Minimal two-generation pedigree: two couples, four offspring
ped <- data.frame(
  id  = 1:8,
  pat = c(0, 0, 0, 0, 1, 1, 3, 3),
  mom = c(0, 0, 0, 0, 2, 2, 4, 4),
  sex = c(1, 2, 1, 2, 1, 2, 1, 2)
)
A <- build_grm(ped, study_ids = 5:8)
round(A, 3)

# With an MZ-twin column: subjects 7 and 8 are MZ (forced to the same sex,
```

```
# since kinship2 requires MZ pairs to share sex -- ordinarily they would
# already match)
ped$sex[ped$id %in% c(7, 8)] <- 1
ped$mztwin <- c(NA, NA, NA, NA, NA, NA, "MZ1", "MZ1")
A_mz <- build_grm(ped, study_ids = 5:8, mz_col = "mztwin")
round(A_mz, 3)  # A[7,8] = 1.0 instead of 0.5
```

`build_household` *Build a household (shared-environment) indicator matrix*

Description

Constructs the block-diagonal indicator matrix **C** used as the shared /common-environment ("household") random-effect design in `herit_ace()`: `C[i, j] = 1` if individuals `i` and `j` belong to the same household group, `0` otherwise (including the diagonal convention `C[i, i] = 1`).

Usage

```
build_household(hhid, ids)
```

Arguments

`hhid`	Vector of household group labels, one per individual, in the same order as `ids`. Individuals sharing an `hhid` value are treated as co-resident (e.g. twin pairs reared together).
`ids`	Vector of individual IDs, the same length as `hhid`. Used to name the rows/columns of the returned matrix so it aligns with a GRM from `build_grm()`.

Value

A symmetric numeric matrix of dimension `length(ids) x length(ids)`, with row/column names `as.character(ids)`.

Examples

```
hh  <- c("A", "A", "B", "B", "C")
ids <- 1:5
build_household(hh, ids)
```

herit_ace *A vs C identifiability check (ACE model comparison)*

Description

Fits four nested variance-component models for a single quantitative trait – sporadic (no familial effect), polygenic/AE (additive genetic only), household/CE (common environment only), and full ACE (A and C estimated jointly) – and returns their log-likelihoods together with the two likelihood-ratio tests needed to judge whether A and C are separately identifiable in the sample: CE vs sporadic (is there a familial signal at all?) and ACE vs AE (does C add anything once A is in the model?). This mirrors SOLAR Eclipse's `house` / `polygenic -screen` model-comparison workflow (see `solar_full_analysis.tcl`, Part 3).

Usage

```
herit_ace(
  trait,
  grm,
  household,
  data,
  covs = NULL,
  screen = TRUE,
  prob_level = 0.1,
  id_col = "IID",
  transform = TRUE,
  min_n = 80L,
  verbose = TRUE
)
```

Arguments

`trait`	Character string: name of the trait column in `data`.
`grm`	Numeric matrix: additive genetic relationship matrix, as returned by `build_grm()`.
`household`	Numeric matrix: household indicator matrix, as returned by `build_household()`. Must have the same row/column names as `grm`.
`data`	Data frame containing `id_col`, `trait`, and any `covs`.
`covs`	Optional character vector of covariate column names in `data` (e.g. `c("age", "sex", "age_sex", "age2", "age2_sex")` for SOLAR's standard age/sex/age*sex/age*^2/*age*^2sex set – construct interaction and polynomial terms as ordinary columns before calling, same convention as `herit_vc()`). Covariates are standardised internally. Default `NULL` (no covariates, matching prior behaviour exactly).
`screen`	Logical. If `covs` is given, apply SOLAR-style automatic covariate screening (see Details) rather than including all of `covs` unconditionally. Default `TRUE`. Ignored if `covs` is `NULL`.
`prob_level`	Screening significance threshold. Default `0.10`, matching SOLAR Eclipse's own default (confirmed from source, `lib/solar.tcl` `proc polygenic`: `set probability_level 0.1`).

id_col	Name of the individual ID column in data. Default "IID".
transform	Logical. Apply int_transform() to trait before fitting. Default TRUE, matching SOLAR Eclipse convention for skewed phenotypes.
min_n	Minimum number of complete observations required. Default 80L.
verbose	Logical. Print progress via cli. Default TRUE.

Details

Unlike herit_vc(), which diagonalises a single variance component via eigen(), this function estimates one or two variance components jointly by direct numerical maximum likelihood over the full n x n covariance matrix (Cholesky-based GLS), since A and household indicator C do not in general share eigenvectors.

Model: Omega = sigma2_p * [h2*A + c2*C + (1 - h2 - c2)*I], jointly estimated for the ACE model via multi-start Nelder-Mead on a logit-reparametrised (h2, c2) (guaranteeing h2, c2 >= 0 and h2 + c2 < 1); AE and CE are 1-D special cases. Covariates (if any) enter the mean structure: E[y] = X %*% beta, with X including an intercept plus the (standardised) columns named in covs.

Covariate screening (when covs is given and screen = TRUE): matches SOLAR Eclipse's polygenic -screen, confirmed from source (lib/solar.tcl proc polygenic) rather than assumed. SOLAR's algorithm is a **single pass**, not iterative backward elimination: fit the AE model with *all* candidate covariates, then for each covariate individually, fit the AE model with just that one covariate removed and compute a likelihood-ratio test against the full-covariate model (chi-squared(1), not boundary-corrected, since a covariate coefficient is not a boundary parameter). Any covariate with p >= prob_level is dropped; all remaining covariates are kept. This happens once, against the AE model specifically (not separately for CE/ACE) – SOLAR's own script can in principle screen separately per model context, but using one shared covariate set for all four nested models here keeps the sporadic/AE/CE/ACE comparisons validly nested (a valid LRT requires the compared models to share the same mean structure).

LRTs for h2/c2: one-sided chi-squared(1) boundary correction, as in herit_vc(), since both null hypotheses (c2 = 0, or h2 = 0) sit on the boundary of the parameter space.

Value

A named list with elements:

trait Trait name.

n Sample size after dropping missing values.

loglik_sporadic, loglik_ae, loglik_ce, loglik_ace Log-likelihoods of the four nested models (all fit with the same final covariate set – see Details).

h2_ae Heritability under the AE model.

c2_ce Common-environment proportion under the CE model.

h2_ace, c2_ace Joint estimates under the full ACE model.

chisq_c_vs_sporadic, p_c_vs_sporadic One-sided boundary LRT of CE vs sporadic (evidence of a familial effect at all).

`chisq_ace_vs_ae`, `p_ace_vs_ae` One-sided boundary LRT of ACE vs AE (evidence C adds anything once A is estimated). A near-zero statistic with `c2_ace` at or near the 0 boundary indicates A and C are not jointly identifiable in this sample – report the simpler AE model (see package vignette / Leocadio-Miguel et al. companion analyses).

`covs_tested`, `covs_kept`, `covs_dropped` Present only if `covs` was supplied and `screen = TRUE`: the full candidate list, the ones retained ($p <$ `prob_level`), and the ones dropped.

`covariate_lrt_p` Present only if `covs` was supplied and `screen = TRUE`: named numeric vector of each candidate covariate's individual LRT p-value against the full-covariate AE model.

Returns `NULL` if `n < min_n`.

See Also

`build_grm()`, `build_household()`, `herit_vc()`, `herit_bivar()`

`herit_batch` *Batch heritability estimation over multiple traits*

Description

A convenience wrapper around `herit_vc()` that iterates over a vector of trait names, optionally across multiple covariate models, and returns a tidy data frame.

Usage

```
herit_batch(
  traits,
  grm,
  data,
  covs_list = NULL,
  id_col = "IID",
  min_n = 80L,
  ci_level = 0.95,
  .progress = TRUE
)
```

Arguments

`traits`	Character vector of trait column names in `data`.
`grm`	Numeric matrix: additive genetic relationship matrix as returned by `build_grm()`.
`data`	Data frame containing ID, trait, and covariate columns.
`covs_list`	A named list of covariate vectors, where each element defines one covariate model. If `NULL`, a single unadjusted model is run. Example: `list(unadj = NULL, cov1 = c("age", "sex"), cov2 = c("age", "sex", "age2"))`.
`id_col`	Name of the individual ID column. Default `"IID"`.
`min_n`	Minimum sample size to attempt estimation. Default `80`.
`ci_level`	Profile-likelihood CI level. Default `0.95`.
`.progress`	Logical. Show a cli progress bar. Default `TRUE`.

Value

A data frame (tibble-compatible) with one row per successfully fitted model and columns: `label`, `trait`, `covariates`, `n`, `h2`, `se`, `ci_lo`, `ci_hi`, `pval`, `sigma2_a`, `sigma2_e`. Failed / skipped models are silently omitted.

See Also

`herit_vc()`, `build_grm()`, `plot_forest()`

Examples

```
## Not run:
A    <- build_grm(my_pedigree, study_ids = my_data$IID)

res  <- herit_batch(
  traits     = c("bmi", "systolic_bp", "hdl"),
  grm        = A,
  data       = my_data,
  covs_list  = list(
    unadj = NULL,
    cov1  = c("age", "sex"),
    cov2  = c("age", "sex", "age2")
  )
)

# Significant adjusted models
subset(res, grepl("cov2", label) & pval < 0.05)

## End(Not run)
```

`herit_bivar` *Bivariate genetic and environmental correlation between two traits*

Description

Fits a bivariate AE variance-components model for two quantitative traits and estimates the additive-genetic correlation (rhoG), the non-shared environmental correlation (rhoE), and the derived phenotypic correlation (rhoP = rhoG*sqrt(h2_1*h2_2) + rhoE*sqrt((1-h2_1)*(1-h2_2))), together with likelihood-ratio tests of rhoG = 0, rhoE = 0, and the joint test of rhoP = 0 (rhoG = rhoE = 0). This mirrors SOLAR Eclipse's `polygenic -testrhog` / `-testrhoe` bivariate output (see `solar_full_analysis.tcl`, Part 5).

Usage

```
herit_bivar(
  trait1,
  trait2,
```

```
  grm,
  data,
  id_col = "IID",
  transform = TRUE,
  min_n = 80L,
  verbose = TRUE
)
```

Arguments

`trait1`, `trait2`	Character strings: names of the two trait columns in `data`.
`grm`	Numeric matrix: additive genetic relationship matrix, as returned by `build_grm()`.
`data`	Data frame containing `id_col`, `trait1`, and `trait2`.
`id_col`	Name of the individual ID column in `data`. Default `"IID"`.
`transform`	Logical. Apply `int_transform()` to both traits before fitting. Default `TRUE`.
`min_n`	Minimum number of individuals contributing to the analysis sample required (an individual counts if at least one of `trait1`, `trait2` is observed – see Details). Default `80L`.
`verbose`	Logical. Print progress via `cli`. Default `TRUE`.

Details

Model: traits are stacked into a 2n-vector with block covariance `Sigma = [[sigma2_g1*A + sigma2_e1*I, rhoG*sqrt(sig` where `sigma2_g_i = h2_i * sigma2_p_i` and `sigma2_e_i = (1-h2_i) * sigma2_p_i`. All six free parameters (`h2_1, h2_2, rhoG, rhoE, sigma2_p1, sigma2_p2`) are estimated jointly by multi-start Nelder-Mead ML on a reparametrised scale (logit for `h2`, `atanh` for the correlations, `log` for the variances) so that all constraints hold automatically.

Missing data: if either trait has missing values, individuals missing exactly one of the two traits are still included in the analysis sample, contributing their observed trait's marginal likelihood, matching SOLAR Eclipse's default `UnbalancedTraits` behaviour (individuals are only excluded if *both* traits, or the ID, are missing). This is both more statistically appropriate than dropping them (more information used under the usual MAR assumption) and matches SOLAR exactly once combined with the SOLAR-exact `inormal` tie handling in `int_transform()` – see NEWS.md for validation. When neither trait has missing data (the common case), this reduces to the same fast eigenbasis computation as before – the unbalanced path is only used when needed, since it cannot exploit the eigenbasis shortcut (the missingness pattern is individual-specific, so the joint covariance no longer shares a common eigenbasis with `A` across all n individuals).

Numerical robustness: both traits are standardised to unit variance before fitting. h2 and the correlations are scale-invariant, so this changes nothing about the true ML answer, but avoids a real failure mode for raw (non-INT-transformed) traits with an extreme variance – e.g. a data-entry problem producing values in the billions – where the unstandardised covariance matrix loses all floating-point precision. rhoG/rhoE are also bounded to `[-0.9, 0.9]`, matching SOLAR's actual default parameter bounds (confirmed from source, not just documentation – see NEWS.md) rather than the mathematical `[-1, 1]`.

As with `herit_ace()`, the unbalanced path uses direct numerical ML over the full covariance matrix rather than an eigendecomposition shortcut.

Validated on the Gomes et al. twin dataset (31 trait pairs), combining correct per-trait inverse-normal transform selection, SOLAR-exact `inormal` tie handling (see `int_transform()`), unbalanced-trait handling, and the numerical robustness fixes above: 28 of 31 pairs match SOLAR Eclipse to 4 decimal places exactly on both rhoG and rhoE, and all 31 pairs agree on nominal significance for both. The 2 remaining pairs both involve a single phenotype (`CRONOTIPO_MCTQ_m` in the validation dataset) that produces wildly different results from SOLAR's reported values even under a fully numerically-robust fit – strong evidence that the underlying data values differ between this package's copy and whatever SOLAR was actually run on, not a modelling or numerical gap in this package. See NEWS.md.

Value

A named list with elements:

`trait1`, `trait2` Trait names.

`n` Number of individuals contributing to the analysis (union of those with `trait1` and/or `trait2` observed – see Details).

`n_complete` Number of individuals with both traits observed. Equal to `n` when neither trait has missing data.

`h2_1`, `h2_2` Heritability of each trait under the joint bivariate model.

`rhoG`, `rhoE`, `rhoP` Genetic, environmental, and derived phenotypic correlations.

`p_rhoG`, `p_rhoE` LRT p-values for rhoG = 0 and rhoE = 0 (standard two-sided chi-squared(1); these correlations are not boundary parameters).

`p_rhoP` LRT p-value for the joint null rhoG = rhoE = 0 (chi-squared(2)).

Returns `NULL` if `n` < `min_n`.

See Also

`build_grm()`, `herit_vc()`, `herit_ace()`, `herit_bivar_batch()`

`herit_bivar_batch` *Batch bivariate genetic/environmental correlations with FDR correction*

Description

Iterates `herit_bivar()` over a set of trait pairs and applies Benjamini-Hochberg FDR correction separately within each of the three p-value families (rhoP, rhoG, rhoE), matching `parse_and_fdr.py`'s post-processing of SOLAR output.

Usage

```
herit_bivar_batch(
  pairs,
  grm,
  data,
  id_col = "IID",
  transform = TRUE,
  min_n = 80L,
  .progress = TRUE
)
```

Arguments

pairs	A two-column character matrix or data frame of trait name pairs (column 1 = trait1, column 2 = trait2), or a list of length-2 character vectors.
grm	Numeric matrix: additive genetic relationship matrix, as returned by build_grm().
data	Data frame containing id_col, trait1, and trait2.
id_col	Name of the individual ID column in data. Default "IID".
transform	Logical. Apply int_transform() to both traits before fitting. Default TRUE.
min_n	Minimum number of individuals contributing to the analysis sample required (an individual counts if at least one of trait1, trait2 is observed – see Details). Default 80L.
.progress	Logical. Show a cli progress bar. Default TRUE.

Value

A data frame with one row per successfully fitted pair and columns trait1, trait2, n, n_complete, h2_1, h2_2, rhoG, rhoE, rhoP, p_rhoG, p_rhoE, p_rhoP, q_rhoG, q_rhoE, q_rhoP (BH-adjusted). Failed / skipped pairs are silently omitted.

See Also

herit_bivar()

---

herit_vc *Profile-likelihood variance-components heritability estimation*

Description

Estimates narrow-sense heritability (h^2) for a single quantitative trait using a profile-likelihood variance-components approach equivalent to SOLAR Eclipse. The phenotype is inverse-normal transformed internally.

Usage

```
herit_vc(
  trait,
  grm,
  data,
  covs = NULL,
  id_col = "IID",
  label = NULL,
  min_n = 80L,
  ci_level = 0.95,
  verbose = TRUE
)
```

Arguments

trait	Character string: name of the trait column in data.
grm	Numeric matrix: the additive genetic relationship matrix for all individuals in data, as returned by build_grm(). Row and column names must match the ID column of data.
data	Data frame containing id_col, trait, and any covariate columns.
covs	Character vector of covariate column names, or NULL for an intercept-only model. Covariates are mean-centred and scaled to unit variance before fitting.
id_col	Name of the individual ID column in data. Default "IID".
label	Optional string label for this model (used in batch output). If NULL, defaults to "<trait>_adj" or "<trait>_unadj".
min_n	Minimum number of complete observations required to attempt estimation. Models with fewer observations return NULL silently. Default 80.
ci_level	Confidence level for the profile-likelihood interval. Default 0.95.
verbose	Logical. Print progress to the console. Default TRUE.

Details

Model: Omega = sigma2_p [h2 A + (1 - h2) I_n]

Optimisation: eigendecomposition of A followed by 1-D profile-likelihood optimisation over h2 in (0, 1) using a coarse grid to seed stats::optimize().

LRT: one-sided chi-squared(1) boundary correction – the null is on the boundary of the parameter space (h2 = 0), so the p-value is halved relative to a standard chi-squared test. This matches the SOLAR Eclipse convention.

CIs: derived by uniroot() on the profile log-likelihood, falling back to a Wald interval if the root-finding fails.

Value

A named list with elements:

label Model label.

`trait` Trait name.

`covariates` Covariate names joined by `"+"`, or `""`.

`n` Sample size after dropping missing values.

`h2` MLE of narrow-sense heritability.

`se` Standard error from profile-likelihood curvature (Wald).

`ci_lo`, `ci_hi` Profile-likelihood confidence interval bounds.

`pval` One-sided LRT p-value with chi-squared(1) boundary correction.

`var_covariates` Proportion of phenotypic variance explained by fixed-effect covariates (R^2 on INT-transformed phenotype). `NA` for unadjusted models. Corresponds to the "variance explained" column in Leocadio-Miguel et al. (2025).

`sigma2_a` Additive genetic variance (σ^2_g in SOLAR notation).

`sigma2_e` Residual environmental variance (σ^2_e).

Returns `NULL` if `n` < `min_n` or if the GRM subset is degenerate.

See Also

`build_grm()`, `herit_batch()`, `int_transform()`

Examples

```
## Not run:
# Build GRM then estimate heritability
A   <- build_grm(my_pedigree, study_ids = my_data$IID)
res <- herit_vc("bmi", grm = A, data = my_data,
                covs = c("age", "sex", "age2"))
str(res)

## End(Not run)
```

`int_transform` *Inverse-normal (rank-based) transformation*

Description

Applies a rank-based inverse-normal transformation (INT) to a numeric vector, placing empirical quantiles onto a standard normal scale.

Usage

```
int_transform(x, method = c("vdw", "blom"), ties = "average")
```

Arguments

x	Numeric vector. NAs are preserved.
method	Which INT formula to use. "vdw" (default) exactly replicates SOLAR Eclipse's inormal command, confirmed against SOLAR's actual source (lib/solar.tcl, proc inormal): sort ascending, assign z_i = qnorm(i / (n+1)) to each individual rank position i, and for tied values average the resulting *z-scores* of their individual rank positions (not the z-score of the averaged rank – these differ for nonlinear qnorm whenever a tied block is not symmetric around n/2, e.g. a large block of tied values at one tail, such as zero-inflated phenotypes). "blom" is the Blom (1958) transformation, qnorm((r-0.5)/n) with conventional averaged-rank tie handling, used by R-itable prior to this fix under the (incorrect) belief that it matched SOLAR.
ties	Method passed to base::rank() when method = "blom". Ignored for method = "vdw", which always uses SOLAR's own tie handling (see Details). Default "average".

Details

For method = "vdw" with no ties, this is the Van der Waerden (1952) transformation, qnorm(r_i / (n+1)), and is bit-identical to it. With ties, tied individuals get the mean of qnorm(i/(n+1)) over the tied rank positions i, exactly matching SOLAR's inormal. This distinction only matters for heavily-tied data; it was confirmed to fully resolve the remaining discrepancy between R-itable and SOLAR Eclipse on zero-inflated phenotypes in the Gomes et al. validation dataset ($> 25\%$ tied values) – see NEWS.md.

For method = "blom":

$$\Phi^{-1}\left(\frac{r_i - 0.5}{n}\right)$$

using base::rank()'s ties argument for tie handling (default "average", i.e. the z-score of the averaged rank – the conventional approach for this formula, not SOLAR's).

Value

A numeric vector of the same length as x, with NAs in the same positions and non-missing values transformed to approximate normality.

Examples

```
set.seed(1)
x <- rexp(200, rate = 0.5)   # right-skewed
hist(x,              main = "Raw")
hist(int_transform(x), main = "INT (SOLAR-matching Van der Waerden)")

# Heavy ties (e.g. zero-inflated activity duration): SOLAR's tie handling
# (default) differs meaningfully from naive rank-average tie handling.
y <- c(rep(0, 40), round(runif(60, 1, 100)))
int_transform(y)[1]                       # SOLAR-exact
qnorm(rank(y, ties.method = "average")[1] / (length(y) + 1))  # naive
```

plot_forest *Forest plot of heritability estimates*

Description

Produces a ggplot2 forest plot from the output of herit_batch() or a list of herit_vc() results coerced to a data frame.

Usage

```
plot_forest(
  results,
  model_filter = NULL,
  colour_by = "trait",
  sig_threshold = 0.05,
  title = NULL,
  x_limits = c(0, 1)
)
```

Arguments

results	Data frame as returned by herit_batch(), containing at least columns label, trait, h2, ci_lo, ci_hi, pval.
model_filter	Optional character vector of model name substrings to keep (matched against label). E.g. "cov2" to show only the age + sex + age-squared model.
colour_by	Column to colour points by. Default "trait". Set to NULL for a monochrome plot.
sig_threshold	Numeric. Traits with pval below this threshold are shown with a filled point; others with an open point. Default 0.05.
title	Optional plot title string.
x_limits	Numeric vector of length 2 for the x-axis range. Default c(0, 1).

Details

Requires **ggplot2** (listed in Suggests). An informative error is thrown if it is not installed.

Value

A ggplot2::ggplot object. Colours follow the Ritable_colours palette by default (pink #FE9EC7 and blue #44ACFF as the primary pair).

See Also

herit_batch(), Ritable_colours

Examples

```
## Not run:
res <- herit_batch(c("bmi", "hdl", "systolic_bp"),
                   grm = A, data = my_data,
                   covs_list = list(unadj = NULL,
                                    cov2  = c("age", "sex", "age2")))

plot_forest(res, model_filter = "cov2", title = "Adjusted heritability")

## End(Not run)
```

`Ritable_colours` *R-itable colour palette*

Description

A named character vector of the four brand colours used throughout R-itable figures and documentation.

Usage

```
Ritable_colours
```

Details

`pink #FE9EC7` – primary accent; significant traits, sex effect

`blue #44ACFF` – primary brand colour; estimates, CI lines

`sky #89D4FF` – secondary blue; muted elements, CI shading

`cream #F9F6C4` – soft highlight; code backgrounds

Examples

```
Ritable_colours
```

Index

base::rank(), *14*
build_grm, 2
build_grm(), *4, 5*, *7–13*
build_household, 4
build_household(), *5*, *7*

ggplot2::ggplot, *15*

herit_ace, 5
herit_ace(), *4*, *9, 10*
herit_batch, 7
herit_batch(), *13*, *15*
herit_bivar, 8
herit_bivar(), *7*, *10, 11*
herit_bivar_batch, 10
herit_bivar_batch(), *10*
herit_vc, 11
herit_vc(), *5–8*, *10*, *15*

int_transform, 13
int_transform(), *6*, *9–11*, *13*

kinship2::kinship(), *3*
kinship2::pedigree(), *3*

plot_forest, 15
plot_forest(), *8*

Ritable_colours, *15*, 16

stats::optimize(), *12*