Package: irtlink (via r-universe)

October 16, 2026

Title IRT Linking and Linking Error Estimation Across Time Points and Groups

Version 0.1.0

Description Estimation of trends in the mean and standard deviation of latent ability distributions across the time points of educational assessments, and of ability distribution differences across linked groups, for unidimensional item response theory (IRT) models (1PL, 2PL, 3PL, partial credit, and generalized partial credit). Groups connected through common items, with independent or dependent person samples, are calibrated on an internal marginal maximum likelihood EM engine, separately, concurrently, with fixed item parameters, or with regularized calibration. Linking methods include mean-mean and mean-geometric mean linking, Haberman linking (Haberman, 2009, <doi:10.1002/j.2333-8504.2009.tb02197.x>) with robust Lp loss and a pairwise variant (Robitzsch, 2025, <doi:10.3390/appliedmath5010007>), and the response function methods of Haebara and Stocking-Lord. Under differential item functioning (DIF), the linked estimates depend on the selected common items, and the package estimates this linking error together with the standard error by jackknife and sandwich estimators, including bias-corrected variants (Robitzsch, 2024, <doi:10.3390/stats7030036>). RMSD and likelihood-ratio DIF diagnostics with iterative purification are included.

License GPL (>= 3)

URL https://github.com/oskarengels/irtlink

BugReports https://github.com/oskarengels/irtlink/issues

Encoding UTF-8

Imports graphics, grDevices, parallel, Rcpp, stats, utils

LinkingTo Rcpp

Suggests knitr, rmarkdown, testthat (>= 3.1.7)

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation yes

Author Oskar Engels [aut, cre] (ORCID: <https://orcid.org/0009-0008-3325-0352>)

Maintainer Oskar Engels <engels@leibniz-ipn.de>

Repository https://cran.r-universe.dev

Date/Publication 2026-10-16 13:30:02 UTC

RemoteUrl https://github.com/cran/irtlink

RemoteRef HEAD

RemoteSha 3c2dd8ba15a445c78dea7033f2949629d7e42db9

Contents

calibrate 2
dependent_vcov 5
detect_dif 7
estimate_trend 9
irtlink-plots 11
link_groups 12
linking_design 15
linking_error 16
person_scores 18
sim_trend_data 19

Index **22**

`calibrate` *Calibrate IRT models per group*

Description

Fits a unidimensional IRT model for each group on the package's internal marginal maximum likelihood EM engine (Bock & Aitkin, 1981, with update formulas following Hanson, 1998) and collects the item parameters in a unified long format. The response model is

$$P(X_{pi} = 1 \mid \theta_p) = c_i + (1 - c_i)\,\mathrm{logit}^{-1}\big(a_i(\theta_p - b_i)\big),$$

with guessing $c_i = 0$ for the 1PL and 2PL and discrimination $a_i = 1$ for the 1PL. Separate calibration fits one model per group. Concurrent calibration fits a single multiple group model (Hanson & Béguin, 2002) and extracts the latent trend directly from the group distribution parameters. Fixed item parameter calibration (FIPC, e.g., Kim, 2006) fixes the group-1 item parameters in the later groups. Regularized calibration frees a DIF effect per item and group and shrinks it under a smooth Bayesian information criterion (SBIC, O'Neill & Burke, 2023, and Robitzsch, 2024).

Usage

```
calibrate(
  data_list,
  model = c("2PL", "1PL", "3PL", "GPCM", "PCM"),
  calibration = c("separate", "concurrent", "fixed", "regularized"),
  engine = "em",
  free_items = NULL,
  eps = 0.001,
  pweights = NULL,
  keep_models = TRUE,
  keep_vcov = FALSE,
  verbose = FALSE,
  ...
)
```

Arguments

`data_list` List of data.frames or matrices (one per group) with dichotomous responses (0/1/NA), or integer categories 0, 1, ... for the polytomous models, and item names as column names.

`model` `"2PL"` (default), `"1PL"`, `"3PL"`, `"GPCM"`, or `"PCM"`. The 3PL estimates the guessing parameter with a beta prior and requires `calibration = "separate"`. The guessing estimates are carried in the `c` column of the canonical `$ipars` table and enter the Haebara and Stocking-Lord linking criteria. `"GPCM"` is the generalized partial credit model (Muraki, 1992) for integer response categories 0, 1, ..., and `"PCM"` its partial credit restriction with all discriminations fixed at 1 (Masters, 1982). Both require `calibration = "separate"`. Their `$ipars` rows carry the discrimination `a`, the item location `b` (the mean of the step difficulties), and centered threshold columns `tau1, tau2, ...` with `b + tau_v` the step difficulties.

`calibration` Calibration strategy. `"separate"` (default) fits one model per group and the trend is recovered afterwards by linking. The other three yield the trend directly: `"concurrent"` fits a multiple group model with invariant item parameters (full, or partial via `free_items`), `"fixed"` is sequential fixed-item-parameter calibration (FIPC), and `"regularized"` is SBIC with a smooth-L0 penalty on differential item functioning (DIF).

`engine` Estimation backend. The internal EM engine (`"em"`) is the only backend and covers all four calibration strategies, with control options via `control = list(...)` in `....`

`free_items` Character vector of item names whose difficulty is allowed to differ across groups (partial invariance). Requires `calibration = "concurrent"`. Default `NULL` (full invariance).

`eps` Smoothing constant(s) for the smooth-L0 BIC penalty in `calibration = "regularized"`. Smaller values yield a sharper L0 approximation. A single value fits once with that constant. A numeric vector (e.g. `c(0.01, 0.001, 0.0001)`) fits a warm-started grid (from large to small eps) and selects the eps with minimum BIC. Default `0.001`. Ignored for other calibration strategies.

pweights	Optional person sampling weights (design weights, e.g. senate weights): a list with one positive numeric vector per group, in the row order of data_list. The weights enter the likelihood of every calibration strategy, so persons contribute proportionally to their weight. With keep_vcov = TRUE the stored covariances are the inverted weighted information. detect_dif() and dependent_vcov() do not use the weights yet.
keep_models	Keep the fitted model objects? Default TRUE. Set FALSE when memory is a concern (the stored fits carry the collapsed response patterns).
keep_vcov	Compute and store item parameter covariance matrices in the package's a,b parameterization (a,b,c for the 3PL). The guessing parameter of the 3PL is estimated with a beta prior that acts like a penalty, so its covariance describes this penalized estimator. Currently supported for calibration = "separate". Default FALSE because the Hessian computation is costly.
verbose	Print estimation progress from the EM engine? Default FALSE.
...	The only recognized argument is control, a list of EM options (e.g. theta_nodes, theta_range, maxit, conv, or lower_a, which overrides the default lower bound 0.1 of the discrimination parameter).

Details

The strategies constrain the common items differently. "concurrent" and "fixed" hold all common item parameters equal across groups, so DIF or item parameter drift (IPD) in a common item biases their trend estimates. The item bank of FIPC is the first group by construction, and the reporting metric remains selectable through the ref argument of estimate_trend(). "regularized" frees a difficulty effect per item and group and shrinks it toward zero, which keeps functionally invariant items constrained while clearly drifting items escape the penalty. The discriminations are not regularized. "separate" places no constraints during calibration, and DIF/IPD is then handled in the linking step or by detection and purification (detect_dif()).

Value

An object of class irtlink_calib: list with ipars (data.frame group, item, a, b, c), item_parameters (data.frame group, item, a, b, d, nu, estimated_parameterization where available), model, calibration, engine, item_parameterization, n_groups, N, converged, models, vcov_ipars, design, trend, free_items, eps, and call. $trend is NULL for separate calibration and a data.frame group, mu, sigma (group 1 = reference: mu 0, sigma 1) for concurrent, fixed, and regularized calibration. $models is a per-group list of fitted model objects for calibration = "separate", a single multiple group fit (in a length-1 list) for calibration = "concurrent", and NULL for "fixed" or "regularized" (or whenever keep_models = FALSE). $vcov_ipars is a per-group list of covariance matrices for separate calibrations when keep_vcov = TRUE, otherwise NULL. Row names append _a / _b (plus _c for the 3PL) to the item name (e.g., I01_a), and linking_error() extracts the a,b blocks it needs by name. $free_items is the character vector of items freed under partial invariance (calibration = "concurrent"), or NULL when no items were freed. $eps is the selected SBIC smoothing constant for calibration = "regularized", or NULL otherwise.

References

Bock, R. D., & Aitkin, M. (1981). Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm. Psychometrika, 46(4), 443-459. doi:10.1007/BF02293801

Hanson, B. A. (1998). IRT parameter estimation using the EM algorithm. Technical note.

Hanson, B. A., & Béguin, A. A. (2002). Obtaining a common scale for item response theory item parameters using separate versus concurrent estimation in the common-item equating design. Applied Psychological Measurement, 26(1), 3-24. doi:10.1177/0146621602026001001

Kim, S. (2006). A comparative study of IRT fixed parameter calibration methods. Journal of Educational Measurement, 43(4), 355-381. doi:10.1111/j.17453984.2006.00021.x

Masters, G. N. (1982). A Rasch model for partial credit scoring. Psychometrika, 47(2), 149-174. doi:10.1007/BF02296272

Muraki, E. (1992). A generalized partial credit model: Application of an EM algorithm. Applied Psychological Measurement, 16(2), 159-176. doi:10.1177/014662169201600206

O'Neill, M., & Burke, K. (2023). Variable selection using a smooth information criterion for distributional regression models. Statistics and Computing, 33, 71. doi:10.1007/s11222023102048

Robitzsch, A. (2024). Smooth information criterion for regularized estimation of item response models. Algorithms, 17(4), 153. doi:10.3390/a17040153

Examples

```
d <- sim_trend_data(n_groups = 2, N = 300, I = 10)
cal <- calibrate(d)
head(coef(cal))
```

`dependent_vcov` *Item parameter covariance for dependent person samples*

Description

Joint covariance matrix of the item parameter estimates of all separately calibrated groups when the same persons were tested in more than one group, as in longitudinal designs that follow a cohort. Persons shared between groups make the item parameter estimates of these groups dependent, and the cross-group blocks of this covariance carry exactly that dependence into `linking_error()`. The stacked method treats the separate calibrations as one stacked M-estimation problem with the block-diagonal observed information as bread and the crossproduct of the person-level scores as meat. The bootstrap method redraws persons with their complete longitudinal records and recalibrates every group per replicate.

Usage

```
dependent_vcov(x, ids, method = c("stacked", "bootstrap"), ...)
```

Arguments

x	For method = "stacked" an irtlink_calib object from calibrate() with calibration = "separate" and keep_models = TRUE. For method = "bootstrap" the list of response matrices as passed to calibrate().
ids	List with one vector of person identifiers per group, in the row order of the response data. Identifiers shared between groups mark the same person. Each identifier may occur at most once per group. sim_trend_data() stores suitable identifiers in its person_ids attribute.
method	"stacked" (default) computes the closed-form stacked sandwich covariance. "bootstrap" estimates the same matrix by a nonparametric bootstrap over persons (e.g., Efron, 1979).
...	Passed to the chosen estimator. The stacked sandwich accepts h (step size of the observed information, default 1e-5). The bootstrap accepts B (number of replicates, default 200), seed, model, cores, and draws (previously computed draws).

Value

A list whose element vcov holds the joint covariance of the stacked item parameters, with row names that join group number, item name, and parameter as in "g1:I01:a" and "g1:I01:b". It can be passed directly as vcov to linking_error() or as le_vcov to estimate_trend(). Both methods also return vcov_items (per-item covariance blocks) and overlap, a matrix whose entry for two groups counts the persons who appear in both. The stacked method additionally returns reference matrices without the dependence (vcov_nocross, vcov_independence) and a score_check. The bootstrap additionally returns its draws and records replicates that were excluded: a replicate is excluded when a calibration did not converge or when a resampled group answered an item entirely correct or entirely wrong, which leaves no response variance to calibrate.

References

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. The Annals of Statistics, 7(1), 1-26. doi:10.1214/aos/1176344552

Examples

```
d <- sim_trend_data(n_groups = 2, N = 300, I = 10,
                    person_overlap = 0.5)
cal <- calibrate(d, keep_models = TRUE)
dv <- dependent_vcov(cal, attr(d, "person_ids"))
dv$overlap
```

`detect_dif` *Detect differential item functioning*

Description

Flags items showing differential item functioning (DIF) across groups, known in longitudinal assessments as item parameter drift (IPD). Two detection methods are available. The root mean square deviation (RMSD) method (`method = "rmsd"`) measures the deviation between an item's group-specific response function and its invariance-constrained reference function,

$$RMSD_i = \sqrt{\int (P_i(\theta) - P_i^{ref}(\theta))^2 \, w(\theta) \, d\theta},$$

with either a parameter-based (population) estimator from linked separate calibrations or a data-based (empirical) estimator computed from a full-invariance multiple group fit (see Köhler, Robitzsch, & Hartig, 2020, and Tijmstra et al., 2020, for the two estimators). The data-based tables also report bias-corrected RMSD values that remove the positive finite sample bias of the sample RMSD (Köhler et al., 2020), using the analytic estimator T4 of Robitzsch (2025). Items are flagged with fixed cutoffs or with the data-driven robust z cutoff of von Davier and Bezirhan (2023), based on the median absolute deviation (MAD),

$$z_i = \frac{RMSD_i - \mathrm{median}(RMSD)}{1.4826 \cdot \mathrm{MAD}(RMSD)} > \tau,$$

in one step or with an iterative forward purification in which flagged items leave the anchor and stay excluded. Operational large-scale assessments flag items at RMSD cutoffs around 0.12 (e.g., OECD, 2024), and cutoff choice is discussed in Köhler et al. (2020) and von Davier and Bezirhan (2023). In trend estimation simulations, the fixed cutoff 0.05 performed well across conditions, 0.03 lost some precision in small samples, 0.08 was too lenient under strong balanced drift, and for the data-driven cutoff `tau = 2.7` kept the better efficiency while `tau = 1.7` with iterative purification minimized bias in extreme conditions (Engels, Lüdtke, & Robitzsch, 2026). The likelihood-ratio test (`method = "lrt"`, Thissen, Steinberg, & Wainer, 1988) splits each item across groups and compares the deviance of a full-invariance multiple group model to that of the item-split model, with all other items as anchor (Wang & Yeh, 2003). `dif_type` chooses whether to test uniform (difficulty), non-uniform (discrimination), or both kinds of DIF/IPD.

Usage

```
detect_dif(
  calib,
  method = c("rmsd", "lrt"),
  rmsd_method = c("parameter", "data", "both"),
  cutoff_type = c("fixed", "data_driven"),
  cutoff = 0.05,
  tau = 2.7,
  flag_on = c("parameter", "data"),
  link_method = "mgm",
```

```
  procedure = c("one_step", "iterative_forward"),
  max_iter = 7,
  min_anchor = 3,
  alpha = 0.05,
  alpha_adjust = c("none", "bonferroni"),
  dif_type = c("both", "uniform", "nonuniform"),
  ...
)
```

Arguments

calib	An irtlink_calib object (separate calibration, keep_models = TRUE).
method	"rmsd" (default) or "lrt" (likelihood-ratio test).
rmsd_method	"parameter" (default), "data", or "both" (RMSD).
cutoff_type	"fixed" (default) or "data_driven" (RMSD).
cutoff	Fixed RMSD cutoff(s). A vector is allowed with procedure = "one_step". Default 0.05.
tau	Robust z cutoff for cutoff_type = "data_driven". Default 2.7.
flag_on	Which estimator drives flagging when rmsd_method = "both".
link_method	Internal linking method for the parameter-based RMSD.
procedure	"one_step" (default) or "iterative_forward". The LRT is always one-step.
max_iter	Maximum purification iterations. Default 7.
min_anchor	Minimum anchor items to retain. Default 3.
alpha	Significance level for method = "lrt". Default 0.05.
alpha_adjust	Multiple-comparison adjustment for the LRT p-values: "none" (default) or "bonferroni".
dif_type	For method = "lrt", the kind of DIF/IPD tested per item. "both" (default) frees discrimination and difficulty with df = 2 per extra group, "uniform" the difficulty only (df = 1), and "nonuniform" the discrimination only (df = 1). Ignored for method = "rmsd".
...	Passed on to the internal fitting functions.

Details

Both methods require a separate calibration with keep_models = TRUE (the per-group responses are reconstructed from the stored models). At least min_anchor items always remain in the anchor (see Kopf, Zeileis, & Strobl, 2015, for a framework of anchor selection methods).

Value

An object of class irtlink_dif: a list with flagged (names of the flagged items), anchor (retained items), rmsd (the driver RMSD table, NULL for the LRT), rmsd_parameter/rmsd_data (per-estimator tables, possibly NULL), lrt (the per-item LRT table item, chisq, df, p, p_adj for method = "lrt", else NULL), flagged_by_cutoff, linkable, history (per-iteration, or NULL), converged, and the call settings (method, dif_type, alpha, etc.).

References

Engels, O., Lüdtke, O., & Robitzsch, A. (2026). Estimating trends with differential item functioning: A comparison of five IRT-based approaches. Educational and Psychological Measurement. doi:10.1177/00131644251408818

Köhler, C., Robitzsch, A., & Hartig, J. (2020). A bias-corrected RMSD item fit statistic: An evaluation and comparison to alternatives. Journal of Educational and Behavioral Statistics, 45(3), 251-273. doi:10.3102/1076998619890566

Kopf, J., Zeileis, A., & Strobl, C. (2015). A framework for anchor methods and an iterative forward approach for DIF detection. Applied Psychological Measurement, 39(2), 83-103. doi:10.1177/0146621614544195

OECD (2024). PISA 2022 technical report. OECD Publishing. doi:10.1787/01820d6den

Robitzsch, A. (2025). Bias-corrected root mean square deviation estimators. Foundations, 5(4), 36. doi:10.3390/foundations5040036

Thissen, D., Steinberg, L., & Wainer, H. (1988). Use of item response theory in the study of group difference in trace lines. In H. Wainer & H. Braun (Eds.), Test validity (pp. 147-170). Lawrence Erlbaum.

Tijmstra, J., Bolsinova, M., Liaw, Y.-L., Rutkowski, L., & Rutkowski, D. (2020). Sensitivity of the RMSD for detecting item-level misfit in low-performing countries. Journal of Educational Measurement, 57(4), 566-583. doi:10.1111/jedm.12263

von Davier, M., & Bezirhan, U. (2023). A robust method for detecting item misfit in large-scale assessments. Educational and Psychological Measurement, 83(4), 740-765. doi:10.1177/00131644221105819

Wang, W.-C., & Yeh, Y.-L. (2003). Effects of anchor item methods on differential item functioning detection with the likelihood ratio test. Applied Psychological Measurement, 27(6), 479-498. doi:10.1177/0146621603259902

Examples

```
set.seed(1)
d <- sim_trend_data(n_groups = 2, N = 400, I = 12, dif = "unbalanced",
                    dif_pct = 0.2, dif_effect = 0.8)
cal <- calibrate(d)
dif <- detect_dif(cal, rmsd_method = "parameter", cutoff = 0.1)
dif$flagged
dif$anchor
```

`estimate_trend` *Estimate a longitudinal IRT trend*

Description

Runs the complete pipeline in one call: calibrates the groups, optionally detects and removes DIF/IPD items, links the groups, and attaches the requested error estimates. `calibration` chooses the calibration, `approach` the composition of the groups (also for concurrent calibration), and `link` chooses the transformation when the groups were calibrated separately. With a bias-corrected linking error method the trend table decomposes the uncertainty into standard error (se), linking error (le and le_bc), and total error (te and te_bc).

Usage

```
estimate_trend(
  data_list,
  calibration = c("separate", "concurrent", "fixed", "regularized"),
  approach = c("chain", "joint", "joint_restricted"),
  link = c("haberman", "haebara", "sl", "mm", "mgm"),
  variant = c("simultaneous", "pairwise"),
  pow = 2,
  weights = "uniform",
  use_intercepts = TRUE,
  item_weights = c("inverse_admin", "uniform"),
  dif = c("none", "purify"),
  le = c("none", "jackknife", "jackknife_bc", "ajk", "ajk_bc", "sandwich_esw",
    "sandwich_osw", "sandwich_bosw", "sandwich_esw_bc", "sandwich_osw_bc",
    "sandwich_bosw_bc", "sandwich_osw_analytic"),
  le_vcov = NULL,
  le_h = 1e-05,
  ref = NULL,
  model = c("2PL", "1PL", "GPCM", "PCM"),
  eps = 0.001,
  ...
)
```

Arguments

`data_list`	List of per-group data.frames or matrices with dichotomous responses.
`calibration`	How the groups are calibrated. `"separate"` fits every group on its own metric and links afterwards. `"concurrent"` fits one multiple group model under full or partial invariance. `"fixed"` is fixed item parameter calibration (FIPC). `"regularized"` penalizes group differences of the item difficulties with a smooth information criterion.
`approach`	How the groups are composed: `"chain"` over successive pairs, `"joint"` in one fit over all groups, or `"joint_restricted"` over the growing window of the first t groups. The approach applies to `"separate"` and `"concurrent"` calibration alike. It is ignored for `"fixed"`, which is sequential by construction.
`link`	Linking method for `calibration = "separate"`: `"haberman"`, `"haebara"`, `"sl"`, `"mm"`, or `"mgm"`. `"mm"` and `"mgm"` need `approach = "chain"`.
`variant`	`"simultaneous"` (default) or `"pairwise"` form of the joint linking methods, as in `link_groups()`.

pow, weights, use_intercepts, item_weights
: Method options passed to the linking step (see link_groups()).

dif
: "none" (default) keeps all common items in the link. "purify" first runs the iterative purification of detect_dif(): items flagged as showing differential item functioning (DIF) or item parameter drift (IPD) leave the link, and the trend is estimated on the remaining items.

le
: Linking error estimator passed to linking_error(), or "none" (default). The *_bc methods need an item parameter covariance (see le_vcov).

le_vcov
: Item parameter covariance for the *_bc methods. When omitted, the per-group covariances of the calibration are used, which treat the person samples as independent. For dependent person samples pass the covariance from dependent_vcov().

le_h
: Step size passed to linking_error().

ref
: Reference group whose metric carries the reported trend (mu = 0, sigma = 1 for this group). Default NULL uses the first group. All linking error estimators report their errors in the reference metric.

model
: "2PL" (default), "1PL", "GPCM", or "PCM". The polytomous models run the separate path with dif = "none". The 3PL is currently available in calibrate() with separate calibration and not yet in this convenience wrapper.

eps
: Tuning constant of the regularized calibration.

...
: Passed to calibrate() and detect_dif(), e.g. person sampling weights as pweights.

Value

An object of class irtlink_trend: a list with trend (data.frame group, mu, sigma, se, le, te and, for the bias-corrected methods, le_bc, te_bc), calibration, approach, link, calib, dif, link_obj, le_obj, and call.

Examples

```
set.seed(1)
d <- sim_trend_data(n_groups = 2, N = 200, I = 10, overlap = 1,
                    dif = "none", mu = c(0, 0.3), sigma = c(1, 1))
tr <- estimate_trend(d, calibration = "separate", approach = "chain",
                     link = "mgm", le = "jackknife")
tr$trend
```

---

irtlink-plots *Plot methods for irtlink objects*

Description

One plot method per result type. For a differential item functioning (DIF) detection result the items are sorted by their statistic (maximum root mean square deviation (RMSD) or likelihood-ratio test (LRT) chi-square) and the detection threshold is drawn as a reference line, with flagged items highlighted. For a linking result each pair of successive groups gets one panel that plots the common items' parameters in the first group against the parameters of the second group transformed onto the first group's metric. Items without DIF or item parameter drift (IPD) scatter around the identity line and outliers stick out. For a trend result the estimated group means are drawn against the group index.

Usage

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

## S3 method for class 'irtlink_link'
plot(x, pars = c("b", "a"), ...)

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

Arguments

`x`	An `irtlink_dif`, `irtlink_link`, or `irtlink_trend` object.
`pars`	Linking plot only. Plot difficulties (`"b"`, default) or discriminations (`"a"`).
`...`	Passed on to `graphics::plot()`.

Value

`x`, invisibly.

`link_groups` *Linking across groups*

Description

Places separately calibrated groups on one common scale. The chain approach accumulates one step per pair of successive groups (e.g., Kolen & Brennan, 2014),

$$\mu_{t+1} = \mu_t + \sigma_t \, \hat{\mu}_{step}, \qquad \sigma_{t+1} = \sigma_t \, \hat{\sigma}_{step}.$$

The joint approaches estimate all transformations in one fit. The methods are the closed-form mean-mean and mean-geometric mean linking, Haberman linking (Haberman, 2009) with the robust Lq loss (Robitzsch, 2020) and a pairwise variant (Robitzsch, 2025), and the response function methods

of Haebara (1980) and Stocking and Lord (1983). The Haebara criterion minimizes the discrepancy of the item response functions over the common items,

$$\sum_i \int \rho\big(P_i(\theta) - P_i^*(\theta)\big)\, w(\theta)\, d\theta,$$

and the Stocking-Lord criterion the discrepancy of the test response functions,

$$\int \rho\Big(\sum_i P_i(\theta) - \sum_i P_i^*(\theta)\Big)\, w(\theta)\, d\theta,$$

where P_i^* is the second group's item response function under the candidate transformation, w the quadrature weights, and $\rho(x) = |x|^p$ the Lp loss, smoothed as $(x^2 + \varepsilon)^{p/2}$. With 3PL item parameters the response functions are the full $c_i + (1 - c_i)\,\mathrm{logit}^{-1}(a_i(\theta - b_i))$ curves (Kolen & Brennan, 2014). The guessing parameter enters the criteria but is itself unaffected by the transformation. The simultaneous joint Haebara fit re-estimates free item parameters and requires $c_i = 0$, its pairwise variant supports the 3PL.

For polytomous item parameters (generalized partial credit model, threshold columns `tau1, tau2, ...`) the chain Haebara criterion compares the category response functions and the chain Stocking-Lord criterion the expected score functions (Kim & Lee, 2006). The pairwise joint criteria use the same polytomous response functions. Mean-mean, mean-geometric mean, Haberman, and pairwise Haberman linking operate on the discriminations and item locations and need no further input. The simultaneous joint Haebara criterion does not yet support polytomous items.

Usage

```
link_groups(
  x,
  approach = c("chain", "joint", "joint_restricted"),
  method = NULL,
  variant = c("simultaneous", "pairwise"),
  ref = NULL,
  ...
)
```

Arguments

`x`	An `irtlink_calib` object or a data.frame with columns `group, item, a, b` (identified item parameters), optionally with a column `c` of 3PL guessing parameters or centered threshold columns `tau1, tau2, ...` for polytomous items.
`approach`	`"chain"` estimates the linear transformation between successive groups and accumulates the steps. `"joint"` estimates the transformations of all groups in one fit. `"joint_restricted"` repeats the joint fit on the growing window of the first t groups and takes each group's result from the fit up to that group.
`method`	`"mgm"` (mean-geometric mean), `"mm"` (mean-mean), `"haberman"`, `"haebara"`, or `"sl"` (Stocking-Lord). The default is `"mgm"` for the chain and `"haberman"` for the joint approaches. `"mm"` and `"mgm"` are available for the chain approach only.

variant — Form of the joint methods. "simultaneous" (default) fits one criterion over all groups at once. "pairwise" fits the criterion over all group pairs with common items. For method = "haberman" the pairwise variant is pairwise Haberman linking (PHL, Robitzsch, 2025).

ref — Reference group whose metric carries the reported trend (mu = 0, sigma = 1 for this group). Default NULL uses the first group. The estimation is unchanged and the trend is re-expressed in the reference metric. linking_error() reports all errors in that metric, so the reference group has linking error 0 and the first group then carries its own.

... — Method options. pow is the Lp loss power (0, 0.25, 0.5, 1, 2) of "haberman", "haebara", and "sl". weights selects the quadrature weight scheme of "haebara" and "sl" ("uniform", "normal_0.5", "normal_1", "normal_2"). use_intercepts makes "haberman" link the item intercepts instead of the difficulties. anchor restricts the linking to a character vector of item names (default: all common items). type and theta_metric select the alignment direction and the quadrature metric of the chain "haebara" and "sl" criteria. item_weights weights the items of the pairwise Haberman variant. eps overrides the smoothing constant of the Lp loss.

Value

An object of class irtlink_link: a list with trend (data.frame group, mu, sigma), steps (chain approach only, one row per successive pair), method, approach, ipars (the normalized item parameters), anchor, le (filled by linking_error()), and call.

References

Haebara, T. (1980). Equating logistic ability scales by a weighted least squares method. Japanese Psychological Research, 22(3), 144-149. doi:10.4992/psycholres1954.22.144

Haberman, S. J. (2009). Linking parameter estimates derived from an item response model through separate calibrations (Research Report RR-09-40). ETS. doi:10.1002/j.23338504.2009.tb02197.x

Kim, S., & Lee, W.-C. (2006). An extension of four IRT linking methods for mixed-format tests. Journal of Educational Measurement, 43(1), 53-76. doi:10.1111/j.17453984.2006.00004.x

Kolen, M. J., & Brennan, R. L. (2014). Test equating, scaling, and linking: Methods and practices (3rd ed.). Springer. doi:10.1007/9781493903177

Robitzsch, A. (2020). Lp loss functions in invariance alignment and Haberman linking with few or many groups. Stats, 3(3), 246-283. doi:10.3390/stats3030019

Robitzsch, A. (2025). Linking error estimation in Haberman linking. AppliedMath, 5(1), 7. doi:10.3390/appliedmath5010007

Stocking, M. L., & Lord, F. M. (1983). Developing a common metric in item response theory. Applied Psychological Measurement, 7(2), 201-210. doi:10.1177/014662168300700208

Examples

```
# Identified parameters on the group-1 metric with a known trend:
# mu = (0, 0.3, 0.6), sigma = (1, 1.1, 1.2)
mu_t <- c(0, 0.3, 0.6)
```

```
sigma_t <- c(1, 1.1, 1.2)
b_ref <- seq(-1.5, 1.5, length.out = 10)
ip <- data.frame(
  group = rep(1:3, each = 10),
  item = rep(sprintf("I%02d", 1:10), 3),
  a = rep(sigma_t, each = 10),
  b = c(b_ref, (b_ref - 0.3) / 1.1, (b_ref - 0.6) / 1.2)
)
link_groups(ip, approach = "chain", method = "mgm")
link_groups(ip, approach = "joint", method = "haberman",
            variant = "pairwise")
```

`linking_design` *Inspect the linking design of a data set list*

Description

Builds the item-by-group incidence matrix (which item is administered in which group) and counts the common items between successive groups. Useful for checking the item overlap before calibration.

Usage

```
linking_design(data_list)
```

Arguments

`data_list`	List of data.frames or matrices, one per group, with item names as column names.

Value

An object of class `irtlink_design`: list with `matrix` (logical, items x groups), `successive_common` (integer vector), and `n_groups`.

Examples

```
d <- sim_trend_data(n_groups = 3, N = 20, I = 10)
linking_design(d)
```

linking_error *Linking error of a linked trend*

Description

Estimates the linking error, the uncertainty of the linked means and standard deviations that stems from the sampling of the link items (e.g., Monseur & Berezner, 2007). The delete-one item jackknife aggregates the refits without each link item around the full estimate,

$$LE^2(\hat{\mu}_t) = \frac{I}{I-1} \sum_{i=1}^{I} \left(\hat{\mu}_{t,(-i)} - \hat{\mu}_t\right)^2,$$

and the same form applies to the linked standard deviation $\hat{\sigma}_t$. The sandwich estimators use the item-additive estimating equations of the closed-form methods (Robitzsch, 2023, 2024): with bread $\mathbf{A}$ and meat $\mathbf{B}$ the covariance of the linking parameters is $\mathbf{V} = \mathbf{A}^{-1}\mathbf{B}(\mathbf{A}^{-1})^\top$, and the linking error is its delta-method projection onto the trend parameters. The bias-corrected variants remove the part of the dispersion that is explained by the person sampling of the item parameters and decompose the result into standard error, linking error, and total error. All errors are reported in the metric of the link's reference group.

Usage

```
linking_error(
  link,
  method = c("jackknife", "jackknife_bc", "ajk", "ajk_bc", "sandwich_esw",
    "sandwich_osw", "sandwich_bosw", "sandwich_esw_bc", "sandwich_osw_bc",
    "sandwich_bosw_bc", "sandwich_osw_analytic"),
  vcov = NULL,
  h = 1e-05,
  jk_scale = c("finite_item", "classical"),
  cluster = NULL,
  ...
)
```

Arguments

link	An `irtlink_link` object returned by `link_groups()`.
method	`"jackknife"` refits the linking without each link item in turn. `"ajk"` replaces the refits by approximate one-step updates of the estimating equations (Robitzsch, 2024, 2025a, 2025b). `"sandwich_esw"`, `"sandwich_osw"`, and `"sandwich_bosw"` are sandwich estimators with the expected, observed, and finite-item corrected observed meat, and `"sandwich_osw_analytic"` evaluates the observed sandwich with analytic derivatives. The `"_bc"` variants apply the finite-sample bias correction of Robitzsch (2024, 2025a, 2025b) and require `vcov`.
vcov	Item parameter covariance matrix or per-group list of covariance matrices, required for the `"_bc"` methods. Use the `vcov_ipars` of `calibrate()` with

keep_vcov = TRUE for independent person samples and dependent_vcov() for dependent ones.

h — Step size of the numeric fallback derivatives of the bias-corrected jackknife. The default analytic derivatives do not use it.

jk_scale — Scaling of the squared delete-one differences for method = "jackknife". "finite_item" (default) uses the factor I/(I-1) and "classical" the factor (I-1)/I of the classical jackknife.

cluster — Optional named character vector mapping item names to testlet labels, e.g. c(I01 = "T1", I02 = "T1"). The deleted jackknife units are then whole testlets instead of single items, which keeps the linking error valid when items within a testlet share stimulus material and drift together (cf. the item-unit jackknife in PISA, Monseur & Berezner, 2007). Items without an entry stay their own unit. Available for "jackknife" (every linking method) and "ajk" (chain mean-mean and mean-geometric mean linking and pairwise Haberman linking). n_reps then counts units.

... — Options overriding the stored linking controls when the leave-one-item links are refit for "jackknife". Ignored with a warning by the other methods.

Value

The input irtlink_link object with its le slot filled: a data.frame with group, le_mu, le_sigma, method, and n_reps. The "_bc" methods add se_*, lebc_*, te_*, and tebc_* columns.

Which linking method supports which estimator

The refit jackknife and its bias-corrected variant are available for every linking method and approach. The approximate jackknife (ajk) covers the chain methods mgm, mm, haebara, and sl (sl with pow = 2) and the pairwise Haberman variant (PHL). The sandwich estimators and all remaining "_bc" variants cover the chain methods mgm and mm and the pairwise Haberman variant. With 3PL guessing parameters the approximate jackknife for haebara and sl is unavailable (its closed forms are derived without guessing). Use the refit jackknife there. For polytomous links the refit jackknife and the approximate jackknife (chain mgm and mm, PHL) are available. The sandwich and bias-corrected estimators are derived for dichotomous models.

References

Monseur, C., & Berezner, A. (2007). The computation of equating errors in international surveys in education. Journal of Applied Measurement, 8(3), 323-335.

Robitzsch, A. (2023). Linking error in the 2PL model. J, 6(1), 58-84. doi:10.3390/j6010005

Robitzsch, A. (2024). Estimation of standard error, linking error, and total error for robust and nonrobust linking methods in the two-parameter logistic model. Stats, 7(3), 592-612. doi:10.3390/stats7030036

Robitzsch, A. (2025a). Linking error estimation in Haberman linking. AppliedMath, 5(1), 7. doi:10.3390/appliedmath5010007

Robitzsch, A. (2025b). Linking error estimation in Stocking-Lord linking. Foundations, 5(1), 2. doi:10.3390/foundations5010002

Examples

```
# Identified parameters with a known trend and one DIF item:
mu_t <- c(0, 0.3, 0.6)
sigma_t <- c(1, 1.1, 1.2)
b_ref <- seq(-1.5, 1.5, length.out = 10)
ip <- data.frame(
  group = rep(1:3, each = 10),
  item = rep(sprintf("I%02d", 1:10), 3),
  a = rep(sigma_t, each = 10),
  b = c(b_ref, (b_ref - 0.3) / 1.1, (b_ref - 0.6) / 1.2)
)
ip$b[ip$group == 2 & ip$item == "I01"] <-
  ip$b[ip$group == 2 & ip$item == "I01"] + 0.5
lk <- link_groups(ip, approach = "chain", method = "mgm")
linking_error(lk, method = "jackknife")$le

# Bias-corrected sandwich linking error needs an item parameter
# covariance. A small diagonal matrix stands in for a real one here
# (in practice use calibrate(..., keep_vcov = TRUE)$vcov_ipars).
V <- diag(0.001, 2 * nrow(lk$ipars))
linking_error(lk, method = "sandwich_osw_bc", vcov = V)$le
```

`person_scores` *Person parameter estimates for an em calibration*

Description

Computes person scores from a fitted `calibrate` object of the internal EM engine (`engine = "em"`): `"eap"` gives the expected a posteriori (EAP) score with its posterior standard deviation under the group's fitted latent distribution, `"wle"` the weighted likelihood estimate (WLE, Warm, 1989) with the standard error $1/\sqrt{I(\hat{\theta})}$, where $I(\theta)$ is the test information over the person's observed items. Supported for `calibration = "separate"` and `"concurrent"` with stored models (`keep_models = TRUE`). Polytomous calibrations (GPCM, PCM) support `"eap"`.

Usage

```
person_scores(calib, method = c("eap", "wle"))
```

Arguments

`calib`	An `irtlink_calib` object fitted with `engine = "em"`.
`method`	`"eap"` (default) or `"wle"`.

Value

data.frame with `group`, `person` (row number within the group's data), `est`, and `se`.

Examples

```
set.seed(1)
d <- sim_trend_data(n_groups = 2, N = 200, I = 8, overlap = 1,
                    dif = "none")
cal <- calibrate(d)
head(person_scores(cal, method = "eap"))
head(person_scores(cal, method = "wle"))
```

---

sim_trend_data *Simulate longitudinal item response data*

Description

Generates dichotomous (2PL/1PL) or polytomous (GPCM) response data for several groups, such as time points, with a linking design. Group 1 is the reference population: the first elements `mu[1]` and `sigma[1]` of the `mu` and `sigma` vectors are its latent mean and standard deviation (typically 0 and 1). Trends are induced through the latent distribution.

Usage

```
sim_trend_data(
  n_groups,
  N,
  I = 20,
  mu = NULL,
  sigma = NULL,
  overlap = 0.4,
  design = c("all", "successive"),
  dif = c("none", "balanced", "unbalanced", "random"),
  dif_pct = 0.3,
  dif_effect = 0.5,
  dif_sd = 0.3,
  dif_par = c("b", "a", "both"),
  dif_effect_a = 0.25,
  dif_sd_a = 0.15,
  person_overlap = 0,
  person_cor = 0.7,
  model = c("2PL", "1PL", "GPCM"),
  n_cat = 3
)
```

Arguments

`n_groups`	Number of groups (at least 2).
`N`	Persons per group.
`I`	Items administered per group.

mu, sigma — Numeric vectors of length n_groups with the latent means μ_t and standard deviations σ_t of the groups, in group order. Defaults: mu = 0.3 * (0:(n_groups-1)), sigma = rep(1, n_groups).

overlap — Proportion of I used as common (link) items, giving round(I * overlap) items.

design — "all" shares one common item set across all groups, "successive" places common items only between neighboring groups.

dif — Type of differential item functioning (DIF, in longitudinal assessments item parameter drift): "none" (default), "balanced" (affected items shift in alternating directions), "unbalanced" (all affected items shift by +dif_effect), or "random" (per-group N(0, dif_sd) shifts). DIF is applied to link-relevant items at groups >= 2. Note that "random" mode draws its shifts before the responses, so it consumes the RNG stream differently from the other modes.

dif_pct — Proportion of link-relevant items affected by DIF. The count is round(n_link * dif_pct) (round-half-to-even). Affected items are chosen in alphabetical order, so for a successive design the DIF concentrates in the earliest link blocks.

dif_effect — Difficulty shift size for balanced/unbalanced DIF.

dif_sd — Standard deviation of the per-group shifts for random DIF.

dif_par — Parameter carrying the DIF: "b" (default, uniform DIF, difficulty shifts), "a" (non-uniform DIF, shifts on the log discriminations, i.e. multiplicative), or "both". "a"/"both" require model = "2PL" and use dif_effect_a/dif_sd_a for the log-discrimination shifts.

dif_effect_a, dif_sd_a — Analogues of dif_effect/dif_sd on the log-discrimination scale, used when dif_par is "a" or "both".

person_overlap — Proportion of the N persons per group that are shared with the previous group (dependent person samples, default 0, independent samples). Shared persons keep their identifier across groups (row names and the person_ids attribute), so the group fits can be passed to dependent_vcov().

person_cor — Latent correlation between the standardized abilities of a shared person in consecutive groups (default 0.7, only used when person_overlap > 0).

model — "2PL" (default), "1PL" (all discriminations fixed at 1), or "GPCM" (polytomous responses from the generalized partial credit model).

n_cat — Number of response categories per item for model = "GPCM" (default 3). The step difficulties are b + tau_v with centered, equally spaced tau_v between -0.6 and 0.6, identical across items.

Value

List of n_groups data.frames (persons x administered items) with attributes true_mu, true_sigma, true_ipars (data.frame item, a, b), design, dif, dif_items (names of the items with DIF), dif_effects (a pool-by-group matrix of injected difficulty shifts), dif_effects_a (the analogous log-discrimination shifts, all zero for dif_par = "b"), and person_ids (list of per-group person identifiers, equal to the row names).

Note

Subsetting the returned list (e.g., `d[1:2]`) drops the attributes. Keep the full object when the true values are needed.

In `random` mode the DIF shifts are drawn before the response data, so the same seed yields different responses than `dif = "none"`. The modes `balanced` and `unbalanced` leave the response RNG stream unchanged.

The default item pool of ten discrimination and difficulty pairs, recycled to the requested test length, follows the pool used in the linking error simulation studies of Robitzsch (2023).

References

Robitzsch, A. (2023). Linking error in the 2PL model. J, 6(1), 58-84. doi:10.3390/j6010005

Examples

```
d <- sim_trend_data(n_groups = 3, N = 100)
attr(d, "true_mu")
```

Index

calibrate, 2, *18*
calibrate(), *6*, *11*, *16*

dependent_vcov, 5
dependent_vcov(), *4*, *11*, *17*, *20*
detect_dif, 7
detect_dif(), *4*, *11*

estimate_trend, 9
estimate_trend(), *4*, *6*

graphics::plot(), *12*

irtlink-plots, 11

link_groups, 12
link_groups(), *10, 11*, *16*
linking_design, 15
linking_error, 16
linking_error(), *4–6*, *11*, *14*

person_scores, 18
plot.irtlink_dif *(*irtlink-plots*)*, 11
plot.irtlink_link *(*irtlink-plots*)*, 11
plot.irtlink_trend *(*irtlink-plots*)*, 11

sim_trend_data, 19
sim_trend_data(), *6*