

Package ‘gtheoryr’

October 7, 2026

Type Package

Title Simple Generalizability Theory for Crossed and Nested Designs

Version 0.2.0

Author Ujjwal Tyagi [aut, cre]

Maintainer Ujjwal Tyagi <ujjwaltyagiii@gmail.com>

Description Provides a small, beginner-friendly interface for estimating variance components in simple generalizability theory designs. The package currently supports a fully crossed persons-by-items design, generic balanced crossed designs with one or more additional facets such as raters, occasions, or forms, and a simple items-within-person nested design, along with design-study summaries for relative and absolute decisions. Includes data diagnostics, measurement error intervals, design comparison and cost planning, sensitivity analysis, Gaussian simulation and parametric bootstrap uncertainty estimates for balanced crossed designs.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

NeedsCompilation no

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Repository CRAN

Date/Publication 2026-10-07 05:30:02 UTC

Contents

gtheoryr-package	2
anova_table	3
bootstrap_gstudy	4
check_gstudy_design	5
dstudy_grid	6
dstudy_nested_ip	6

dstudy_pxi	7
dstudy_pxif	8
dstudy_pxir	9
dstudy_sensitivity	10
error_budget	11
gstudy_crossed	11
gstudy_nested_ip	12
gstudy_pxi	13
gstudy_pxif	14
gstudy_pxir	15
gstudy_pxiro	16
optimize_dstudy	17
score_interval	18
sem_gtheory	19
simulate_gstudy	20
variance_proportions_table	21
Index	22

gtheoryr-package

gtheoryr: Simple Generalizability Theory for R

Description

Small, beginner-friendly helpers for estimating variance components in simple generalizability theory designs. The package currently supports:

- a crossed persons-by-items design via `gstudy_pxi()`
- generic balanced crossed designs with one or more facets via `gstudy_crossed()`
- a crossed persons-by-items-by-facet design via `gstudy_pxif()`
- convenience wrappers for raters and occasions via `gstudy_pxir()` and `gstudy_pxio()`
- convenience wrappers for persons-by-items-by-raters-by-occasions designs via `gstudy_pxiro()`
- a nested items-within-person design via `gstudy_nested_ip()`
- simple D-studies via `dstudy_pxi()`, `dstudy_crossed()`, `dstudy_pxif()`, `dstudy_pxir()`, `dstudy_pxio()`, `dstudy_pxiro()`, and `dstudy_nested_ip()`
- table extractors via `anova_table()`, `mean_squares_table()`, `variance_components_table()`, and `variance_proportions_table()`

anova_table	<i>Extract Tables from a G-study Object</i>
-------------	---

Description

Convenience extractors for sums of squares, mean squares, and variance components from a "gstudy_gtheoryr" object.

Usage

```
anova_table(x, ...)  
  
mean_squares_table(x, ...)  
  
variance_components_table(x, ...)
```

Arguments

x	A "gstudy_gtheoryr" object.
...	Unused.

Value

A data frame.

Examples

```
scores <- data.frame(  
  person = rep(c("P1", "P2", "P3"), each = 3),  
  item = rep(c("I1", "I2", "I3"), times = 3),  
  score = c(8, 7, 9, 5, 4, 6, 7, 6, 8)  
)  
  
gs <- gstudy_pxi(scores, person = "person", item = "item", score = "score")  
  
anova_table(gs)  
mean_squares_table(gs)  
variance_components_table(gs)
```

bootstrap_gstudy

*Parametric Bootstrap Uncertainty for a Crossed G-study***Description**

Parametric Bootstrap Uncertainty for a Crossed G-study

Usage

```
bootstrap_gstudy(
  gstudy,
  B = 1000,
  design_levels = NULL,
  conf = 0.95,
  seed = NULL,
  negative = c("error", "zero")
)
```

Arguments

<code>gstudy</code>	A fitted G-study from any supported estimator.
<code>B</code>	Number of bootstrap replicates, an integer at least two. Use at least 1000 for substantive work and inspect Monte Carlo stability.
<code>design_levels</code>	Optional named integer vector of proposed facet counts. Use singular names, for example <code>c(item = 10, rater = 3)</code> .
<code>conf</code>	Confidence level strictly between zero and one.
<code>seed</code>	Optional integer seed, restoring the caller's RNG state afterward.
<code>negative</code>	Either "error" (default) or "zero". The latter explicitly truncates negative component estimates for planning and records their names in the <code>adjusted_components</code> attribute. It does not change the fitted object.

Value

A list with percentile intervals (`estimate`, `lower`, `upper`, `n_valid`), replicate estimates, `B`, `conf`, `method`, `adjusted_components` and `replicate_adjustments`. Simulates all random facets at their original sample sizes, then refits ANOVA. Proposed `design_levels` affect D-study results only. Refit negative components are truncated to zero and counted. Raw variance estimates are retained in replicate output; coefficient estimates use the truncated components. Undefined coefficients remain NA and are counted. This Gaussian parametric method is not the nonparametric bias-corrected procedure described by Tong and Brennan. Coverage is approximate, especially near zero variance boundaries. Simple nested designs are not supported.

References

Tong and Brennan (2007), doi:10.1177/0013164407301533.

check_gstudy_design	<i>Check Data Before Fitting a G-study</i>
---------------------	--

Description

Reports missing scores and identifiers, duplicate cells, missing crossed cells and insufficient facet levels without fitting a model or dropping rows.

Usage

```
check_gstudy_design(  
  data,  
  person,  
  facets,  
  score,  
  design = c("crossed", "nested")  
)
```

Arguments

data	A data frame in long format.
person	Name of the person column.
facets	Character vector of facet columns. For nested designs supply the single item column, with globally unique item labels.
score	Name of the numeric score column.
design	Either "crossed" or "nested" (items within persons).

Value

A list with valid, issues, level_counts and diagnostic row/cell counts. Missing cells refer to the Cartesian product of observed nonmissing levels.

Examples

```
d <- data.frame(p = rep(1:3, each = 2), i = rep(1:2, 3), y = 1:6)  
check_gstudy_design(d, "p", "i", "y")
```

dstudy_grid	<i>Compare a Grid of Candidate D-study Designs</i>
-------------	--

Description

Compare a Grid of Candidate D-study Designs

Usage

```
dstudy_grid(gstudy, candidates, negative = c("error", "zero"))
```

Arguments

gstudy	A fitted G-study from any supported estimator.
candidates	Named list of positive integer vectors, one for each facet.
negative	Either "error" (default) or "zero". The latter explicitly truncates negative component estimates for planning and records their names in the adjusted_components attribute. It does not change the fitted object.

Value

A data frame with facet counts, observations per person, error variances, G and Phi coefficients and SEMs. At most 100000 combinations. Undefined coefficients (zero universe variance and zero error) are NA.

dstudy_nested_ip	<i>Design Study for a Nested Items-within-Person Design</i>
------------------	---

Description

Computes a simple reliability summary for a proposed number of nested items per person.

Usage

```
dstudy_nested_ip(gstudy, n_items = gstudy$n_items)
```

Arguments

gstudy	A result from gstudy_nested_ip().
n_items	Number of nested items per person in the proposed design.

Value

An object of class "dstudy_gtheoryr".

Examples

```
nested_scores <- data.frame(
  person = c("P1", "P1", "P2", "P2", "P3", "P3"),
  item = c("P1_I1", "P1_I2", "P2_I1", "P2_I2", "P3_I1", "P3_I2"),
  score = c(8, 6, 5, 4, 9, 7)
)

gs_nested <- gstudy_nested_ip(
  nested_scores,
  person = "person",
  item = "item",
  score = "score"
)
dstudy_nested_ip(gs_nested, n_items = 4)
```

dstudy_pxi

*Design Study for a Crossed Persons-by-Items Design***Description**

Computes relative error, absolute error, a generalizability coefficient, and a phi coefficient for a proposed number of items.

Usage

```
dstudy_pxi(gstudy, n_items = gstudy$n_items)
```

Arguments

gstudy A result from `gstudy_pxi()`.
n_items Number of items in the proposed design.

Value

An object of class "dstudy_gtheoryr".

Examples

```
scores <- data.frame(
  person = rep(c("P1", "P2", "P3"), each = 3),
  item = rep(c("I1", "I2", "I3"), times = 3),
  score = c(8, 7, 9, 5, 4, 6, 7, 6, 8)
)

gs <- gstudy_pxi(scores, person = "person", item = "item", score = "score")
dstudy_pxi(gs, n_items = 6)
```

dstudy_pxif

*Design Study for a Crossed Persons-by-Items-by-Facet Design***Description**

Computes relative and absolute error variances, the generalizability coefficient, and the phi coefficient for the current or proposed number of items and levels of a third crossed facet such as raters, occasions, or forms.

Usage

```
dstudy_pxif(gstudy, n_items = gstudy$n_items, n_facets = gstudy$n_facets)
```

Arguments

<code>gstudy</code>	A result from <code>gstudy_pxif()</code> .
<code>n_items</code>	Number of items in the design. Defaults to the current number of items in the supplied <code>gstudy</code> object.
<code>n_facets</code>	Number of levels of the third facet. Defaults to the current number in the supplied <code>gstudy</code> object.

Value

An object of class "dstudy_gtheoryr".

Examples

```
scores <- expand.grid(
  person = c("P1", "P2", "P3"),
  item = c("I1", "I2"),
  rater = c("R1", "R2"),
  stringsAsFactors = FALSE
)
scores$score <- c(8, 7, 7, 6, 5, 4, 6, 5, 7, 6, 8, 7)

gs <- gstudy_pxif(
  scores,
  person = "person",
  item = "item",
  facet = "rater",
  score = "score",
  facet_name = "rater"
)

dstudy_pxif(gs)
```

Description

Convenience wrappers around `dstudy_pxif()` for crossed persons-by-items-by-raters and crossed persons-by-items-by-occasions designs.

Usage

```
dstudy_pxir(gstudy, n_items = gstudy$n_items, n_raters = gstudy$n_facets)

dstudy_pxio(gstudy, n_items = gstudy$n_items, n_occasions = gstudy$n_facets)
```

Arguments

<code>gstudy</code>	A result from <code>gstudy_pxir()</code> or <code>gstudy_pxio()</code> .
<code>n_items</code>	Number of items in the design. Defaults to the current number in the supplied <code>gstudy</code> object.
<code>n_raters</code>	Number of raters in the design. Defaults to the current number in the supplied <code>gstudy</code> object.
<code>n_occasions</code>	Number of occasions in the design. Defaults to the current number in the supplied <code>gstudy</code> object.

Value

An object of class "dstudy_gtheoryr".

Examples

```
scores_rater <- expand.grid(
  person = c("P1", "P2", "P3"),
  item = c("I1", "I2"),
  rater = c("R1", "R2"),
  stringsAsFactors = FALSE
)
scores_rater$score <- c(8, 7, 7, 6, 5, 4, 6, 5, 7, 6, 8, 7)

gs_rater <- gstudy_pxir(
  scores_rater,
  person = "person",
  item = "item",
  rater = "rater",
  score = "score"
)
dstudy_pxir(gs_rater)

scores_occasion <- expand.grid(
```

```

person = c("P1", "P2", "P3"),
item = c("I1", "I2"),
occasion = c("T1", "T2"),
stringsAsFactors = FALSE
)
scores_occasion$score <- c(8, 8, 7, 7, 5, 5, 6, 6, 7, 7, 8, 8)

gs_occasion <- gstudy_pxio(
  scores_occasion,
  person = "person",
  item = "item",
  occasion = "occasion",
  score = "score"
)
dstudy_pxio(gs_occasion)

```

dstudy_sensitivity	<i>Examine Gains from Increasing One Facet at a Time</i>
--------------------	--

Description

Examine Gains from Increasing One Facet at a Time

Usage

```

dstudy_sensitivity(
  gstudy,
  design_levels = NULL,
  increment = 1,
  negative = c("error", "zero")
)

```

Arguments

<code>gstudy</code>	A fitted G-study from any supported estimator.
<code>design_levels</code>	Optional named integer vector of proposed facet counts. Use singular names, for example <code>c(item = 10, rater = 3)</code> .
<code>increment</code>	Positive integer increase applied separately to each facet.
<code>negative</code>	Either "error" (default) or "zero". The latter explicitly truncates negative component estimates for planning and records their names in the <code>adjusted_components</code> attribute. It does not change the fitted object.

Value

One row per facet with old and new counts, added observations per person, new coefficients and changes from the baseline G and Phi. This is a local comparison conditional on the fitted variance components.

error_budget	<i>Decompose D-study Error by Variance Component</i>
--------------	--

Description

Decompose D-study Error by Variance Component

Usage

```
error_budget(gstudy, design_levels = NULL, negative = c("error", "zero"))
```

Arguments

gstudy	A fitted G-study from any supported estimator.
design_levels	Optional named integer vector of proposed facet counts. Use singular names, for example <code>c(item = 10, rater = 3)</code> .
negative	Either "error" (default) or "zero". The latter explicitly truncates negative component estimates for planning and records their names in the <code>adjusted_components</code> attribute. It does not change the fitted object.

Value

A data frame of component estimates, divisors, and contributions to relative and absolute error variance. All facets are random.

Examples

```
d <- data.frame(p = rep(1:3, each = 2), i = rep(1:2, 3), y = 1:6)
gs <- gstudy_pxi(d, "p", "i", "y")
error_budget(gs, negative = "zero")
```

gstudy_crossed	<i>Generic Crossed G-study and D-study Helpers</i>
----------------	--

Description

`gstudy_crossed()` estimates ANOVA mean squares and variance components for a balanced crossed random-effects design with persons as the object of measurement and one or more additional facets.

`dstudy_crossed()` computes relative and absolute error variances, the generalizability coefficient, and the phi coefficient for the current or proposed levels of the additional facets.

Usage

```
gstudy_crossed(data, person, facets, score, facet_labels = facets)

dstudy_crossed(gstudy, design_levels = gstudy$design_levels)
```

Arguments

<code>data</code>	A data frame containing one row per observed cell.
<code>person</code>	Name of the person column.
<code>facets</code>	Character vector of facet column names.
<code>score</code>	Name of the numeric score column.
<code>facet_labels</code>	Optional user-facing labels for the supplied facets.
<code>gstudy</code>	A result from <code>gstudy_crossed()</code> or one of its crossed-design wrappers.
<code>design_levels</code>	Optional named numeric vector giving the facet counts to use in the D-study. Defaults to the current design in <code>gstudy</code> .

Value

For `gstudy_crossed()`, an object of class "gstudy_gtheoryr".

For `dstudy_crossed()`, an object of class "dstudy_gtheoryr".

Examples

```
scores <- read.csv(
  system.file("extdata", "crossed_scores_rater_occasion.csv", package = "gtheoryr"),
  stringsAsFactors = FALSE
)

gs <- gstudy_crossed(
  scores,
  person = "person",
  facets = c("item", "rater", "occasion"),
  score = "score",
  facet_labels = c("item", "rater", "occasion")
)
gs
dstudy_crossed(gs)
```

<code>gstudy_nested_ip</code>	<i>Estimate Variance Components for a Nested Items-within-Person Design</i>
-------------------------------	---

Description

Estimates ANOVA mean squares and variance components for a simple balanced nested design in which each person has their own set of items.

Usage

```
gstudy_nested_ip(data, person, item, score)
```

Arguments

<code>data</code>	A data frame containing one row per observation.
<code>person</code>	Name of the person column.
<code>item</code>	Name of the nested item column.
<code>score</code>	Name of the numeric score column.

Value

An object of class "gstudy_gtheoryr".

Examples

```
nested_scores <- data.frame(
  person = c("P1", "P1", "P2", "P2", "P3", "P3"),
  item = c("P1_I1", "P1_I2", "P2_I1", "P2_I2", "P3_I1", "P3_I2"),
  score = c(8, 6, 5, 4, 9, 7)
)

gs_nested <- gstudy_nested_ip(
  nested_scores,
  person = "person",
  item = "item",
  score = "score"
)
gs_nested
```

<code>gstudy_pxi</code>	<i>Estimate Variance Components for a Crossed Persons-by-Items Design</i>
-------------------------	---

Description

Estimates ANOVA mean squares and variance components for a fully crossed random-effects persons-by-items design.

Usage

```
gstudy_pxi(data, person, item, score)
```

Arguments

<code>data</code>	A data frame containing one row per person-item observation.
<code>person</code>	Name of the person column.
<code>item</code>	Name of the item column.
<code>score</code>	Name of the numeric score column.

Value

An object of class "gstudy_gtheoryr".

Examples

```
scores <- data.frame(
  person = rep(c("P1", "P2", "P3"), each = 3),
  item = rep(c("I1", "I2", "I3"), times = 3),
  score = c(8, 7, 9, 5, 4, 6, 7, 6, 8)
)

gs <- gstudy_pxi(scores, person = "person", item = "item", score = "score")
gs
```

gstudy_pxif	<i>Estimate Variance Components for a Crossed Persons-by-Items-by-Facet Design</i>
-------------	--

Description

Estimates ANOVA mean squares and variance components for a fully crossed random-effects design with persons crossed with items and one additional facet such as raters, occasions, or forms.

Usage

```
gstudy_pxif(data, person, item, facet, score, facet_name = facet)
```

Arguments

data	A data frame containing one row per person-item-facet observation.
person	Name of the person column.
item	Name of the item column.
facet	Name of the third facet column, such as a rater, occasion, or form.
score	Name of the numeric score column.
facet_name	A user-facing label for the third facet. Defaults to the value supplied to facet.

Value

An object of class "gstudy_gtheoryr".

Examples

```

scores <- expand.grid(
  person = c("P1", "P2", "P3"),
  item = c("I1", "I2"),
  rater = c("R1", "R2"),
  stringsAsFactors = FALSE
)
scores$score <- c(8, 7, 7, 6, 5, 4, 6, 5, 7, 6, 8, 7)

gs <- gstudy_pxif(
  scores,
  person = "person",
  item = "item",
  facet = "rater",
  score = "score",
  facet_name = "rater"
)
gs

```

gstudy_pxir

*Convenience Wrappers for Common Three-Facet Designs***Description**

Convenience wrappers around `gstudy_pxif()` for crossed persons-by-items-by-raters and crossed persons-by-items-by-occasions designs.

Usage

```

gstudy_pxir(data, person, item, rater, score)

gstudy_pxio(data, person, item, occasion, score)

```

Arguments

<code>data</code>	A data frame containing one row per observation.
<code>person</code>	Name of the person column.
<code>item</code>	Name of the item column.
<code>rater</code>	Name of the rater column.
<code>occasion</code>	Name of the occasion column.
<code>score</code>	Name of the numeric score column.

Value

An object of class "gstudy_gtheoryr".

Examples

```

scores_rater <- expand.grid(
  person = c("P1", "P2", "P3"),
  item = c("I1", "I2"),
  rater = c("R1", "R2"),
  stringsAsFactors = FALSE
)
scores_rater$score <- c(8, 7, 7, 6, 5, 4, 6, 5, 7, 6, 8, 7)

gstudy_pxir(
  scores_rater,
  person = "person",
  item = "item",
  rater = "rater",
  score = "score"
)

scores_occasion <- expand.grid(
  person = c("P1", "P2", "P3"),
  item = c("I1", "I2"),
  occasion = c("T1", "T2"),
  stringsAsFactors = FALSE
)
scores_occasion$score <- c(8, 8, 7, 7, 5, 5, 6, 6, 7, 7, 8, 8)

gstudy_pxio(
  scores_occasion,
  person = "person",
  item = "item",
  occasion = "occasion",
  score = "score"
)

```

gstudy_pxiro

Helpers for a Crossed Persons-by-Items-by-Raters-by-Occasions Design

Description

Convenience wrappers around `gstudy_crossed()` and `dstudy_crossed()` for a design with persons, items, raters, and occasions.

Usage

```

gstudy_pxiro(data, person, item, rater, occasion, score)

dstudy_pxiro(gstudy, n_items = gstudy$design_levels[["item"]],
  n_raters = gstudy$design_levels[["rater"]],
  n_occasions = gstudy$design_levels[["occasion"]])

```

Arguments

data	A data frame containing one row per person-item-rater-occasion observation.
person	Name of the person column.
item	Name of the item column.
rater	Name of the rater column.
occasion	Name of the occasion column.
score	Name of the numeric score column.
gstudy	A result from gstudy_pxiro().
n_items	Number of items in the design. Defaults to the current number.
n_raters	Number of raters in the design. Defaults to the current number.
n_occasions	Number of occasions in the design. Defaults to the current number.

Value

For gstudy_pxiro(), an object of class "gstudy_gtheoryr".

For dstudy_pxiro(), an object of class "dstudy_gtheoryr".

Examples

```
scores <- read.csv(
  system.file("extdata", "crossed_scores_rater_occasion.csv", package = "gtheoryr"),
  stringsAsFactors = FALSE
)

gs <- gstudy_pxiro(
  scores,
  person = "person",
  item = "item",
  rater = "rater",
  occasion = "occasion",
  score = "score"
)
gs
dstudy_pxiro(gs)
```

optimize_dstudy

Find the Least Cost Candidate Meeting a Reliability Target

Description

Find the Least Cost Candidate Meeting a Reliability Target

Usage

```
optimize_dstudy(  
  gstudy,  
  candidates,  
  target = 0.8,  
  coefficient = c("phi", "g"),  
  costs = NULL,  
  observation_cost = 1,  
  budget = Inf,  
  negative = c("error", "zero")  
)
```

Arguments

gstudy	A fitted G-study from any supported estimator.
candidates	Named list of positive integer vectors, one for each facet.
target	Required reliability strictly between zero and one.
coefficient	Either "phi" or "g".
costs	Optional nonnegative named per-level costs for every facet.
observation_cost	Nonnegative cost per observed cell per person.
budget	Maximum allowed cost; defaults to Inf.
negative	Either "error" (default) or "zero". The latter explicitly truncates negative component estimates for planning and records their names in the adjusted_components attribute. It does not change the fitted object.

Value

A list with feasible, best (all minimum-cost ties), evaluated and target. Cost equals $\text{sum}(\text{costs} * \text{counts}) + \text{observation_cost} * \text{prod}(\text{counts})$, expressed per person. Without costs, cost is the number of observations. The optimum is only over the supplied candidate grid. No feasible design returns a zero-row best table rather than an invented recommendation.

References

Meyer, Liu and Mashburn (2014), doi:10.1177/0013164413508774.

score_interval	<i>Construct Approximate Measurement Intervals Around Mean Scores</i>
----------------	---

Description

Construct Approximate Measurement Intervals Around Mean Scores

Usage

```
score_interval(  
  gstudy,  
  scores,  
  design_levels = NULL,  
  decision = c("absolute", "relative"),  
  conf = 0.95,  
  negative = c("error", "zero")  
)
```

Arguments

gstudy	A fitted G-study from any supported estimator.
scores	Numeric vector of observed mean scores on the original scale.
design_levels	Optional named integer vector of proposed facet counts. Use singular names, for example c(item = 10, rater = 3).
decision	"absolute" (default) or "relative".
conf	Confidence level strictly between zero and one.
negative	Either "error" (default) or "zero". The latter explicitly truncates negative component estimates for planning and records their names in the adjusted_components attribute. It does not change the fitted object.

Value

A data frame with score, sem, lower and upper. Normal-theory intervals use a common SEM and treat estimated variance components as known. They are not conditional SEMs, prediction intervals or coefficient confidence intervals. Relative intervals exclude systematic facet effects and should not be used for absolute cut-score decisions. Bounds are not clipped.

sem_gtheory	<i>Report Relative and Absolute Standard Errors of Measurement</i>
-------------	--

Description

Report Relative and Absolute Standard Errors of Measurement

Usage

```
sem_gtheory(gstudy, design_levels = NULL, negative = c("error", "zero"))
```

Arguments

<code>gstudy</code>	A fitted G-study from any supported estimator.
<code>design_levels</code>	Optional named integer vector of proposed facet counts. Use singular names, for example <code>c(item = 10, rater = 3)</code> .
<code>negative</code>	Either "error" (default) or "zero". The latter explicitly truncates negative component estimates for planning and records their names in the <code>adjusted_components</code> attribute. It does not change the fitted object.

Value

A two-row data frame with `decision`, `error_variance` and `sem`, on the mean-score scale. SEM is the square root of the corresponding D-study error variance, not the sampling standard error of a reliability coefficient.

<code>simulate_gstudy</code>	<i>Simulate a Balanced Gaussian Crossed G-study</i>
------------------------------	---

Description

Simulate a Balanced Gaussian Crossed G-study

Usage

```
simulate_gstudy(
  n_persons,
  design_levels,
  variance_components,
  mean = 0,
  seed = NULL
)
```

Arguments

<code>n_persons</code>	Number of persons, an integer at least two.
<code>design_levels</code>	Named integer counts of at least two for each facet.
<code>variance_components</code>	Named nonnegative variances. Names use person, facet names and colon-separated interactions in design order. The highest interaction is named residual. Omitted components are zero.
<code>mean</code>	Grand mean of the simulated scores.
<code>seed</code>	Optional integer seed, restoring the caller's RNG state afterward.

Value

A long-format data frame with `person`, `facet` columns and `score`. Independent Gaussian effects are shared by observations with the same factor combination. The highest interaction and cell error are confounded. Scores are continuous and unbounded; no ordinal or binary model is implied.

Examples

```
d <- simulate_gstudy(20, c(item = 4),  
                     c(person = 2, item = 0.2, residual = 1), seed = 42)  
gstudy_crossed(d, "person", "item", "score")
```

`variance_proportions_table`*Variance Component Proportions from a G-study*

Description

Returns the estimated variance components together with their proportion of the sum of estimated variance components.

Usage

```
variance_proportions_table(x, ...)
```

Arguments

<code>x</code>	A "gstudy_gtheoryr" object.
<code>...</code>	Unused.

Value

A data frame.

Examples

```
scores <- read.csv(  
  system.file("extdata", "crossed_scores_rater_occasion.csv", package = "gtheoryr"),  
  stringsAsFactors = FALSE  
)  
  
gs <- gstudy_pxiro(  
  scores,  
  person = "person",  
  item = "item",  
  rater = "rater",  
  occasion = "occasion",  
  score = "score"  
)  
  
variance_proportions_table(gs)
```

Index

`anova_table`, [3](#)

`bootstrap_gstudy`, [4](#)

`check_gstudy_design`, [5](#)

`dstudy_crossed` (`gstudy_crossed`), [11](#)

`dstudy_grid`, [6](#)

`dstudy_nested_ip`, [6](#)

`dstudy_pxi`, [7](#)

`dstudy_pxif`, [8](#)

`dstudy_pxio` (`dstudy_pxir`), [9](#)

`dstudy_pxir`, [9](#)

`dstudy_pxiro` (`gstudy_pxiro`), [16](#)

`dstudy_sensitivity`, [10](#)

`error_budget`, [11](#)

`gstudy_crossed`, [11](#)

`gstudy_nested_ip`, [12](#)

`gstudy_pxi`, [13](#)

`gstudy_pxif`, [14](#)

`gstudy_pxio` (`gstudy_pxir`), [15](#)

`gstudy_pxir`, [15](#)

`gstudy_pxiro`, [16](#)

`gtheoryr` (`gtheoryr-package`), [2](#)

`gtheoryr-package`, [2](#)

`mean_squares_table` (`anova_table`), [3](#)

`optimize_dstudy`, [17](#)

`score_interval`, [18](#)

`sem_gtheory`, [19](#)

`simulate_gstudy`, [20](#)

`variance_components_table`
 (`anova_table`), [3](#)

`variance_proportions_table`, [21](#)