Package 'mlstats'

July 11, 2026

Title Multilevel Descriptive Statistics and Data Preparation

Version 0.1.0

Description Provides tools for multilevel descriptive statistics and data preparation. Computes within-group and between-group correlations (via variance decomposition or two-level structural equation modeling), intraclass correlation coefficients (ICCs), and descriptive statistics for nested data (e.g., repeated measurements per person), supporting both frequentist (via 'lme4' or 'lavaan') and Bayesian (via 'brms') estimation. Results are formatted according to APA standards and can be exported as tables using 'gt' or 'tinytable'. Also includes functions for decomposing variables into within-group and between-group components for use in Random Effects Within-Between (REWB) models.

License MIT + file LICENSE

URL https://felixdidi.github.io/mlstats/, https://github.com/felixdidi/mlstats

BugReports https://github.com/felixdidi/mlstats/issues

Encoding UTF-8

LazyData true

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Suggests brms, gt, knitr, lavaan, lmerTest, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

Imports cli, dplyr, lme4, pillar, rlang, scales, stringr, tibble, tinytable, vctrs

NeedsCompilation no

Author Felix Dietrich [aut, cre, cph]

Maintainer Felix Dietrich <mail@felix-dietrich.de>

Repository CRAN

Date/Publication 2026-07-11 09:00:02 UTC

Contents

`decompose_within_between` *Decompose Variables into Within-Group and Between-Group Components*

Description

This function performs a multilevel decomposition of variables by computing:

- Grand mean centered scores (deviations from overall mean)
- Between-group scores (group means)
- Within-group scores (deviations from group means)

Usage

```
decompose_within_between(
  data,
  group,
  vars,
  components = c("gmc", "between", "within"),
  gmc_pattern = "{col}_grand_mean_centered",
  between_pattern = "{col}_between_{group}",
  within_pattern = "{col}_within_{group}"
)
```

Arguments

`data`	A data frame containing the variables to decompose.
`group`	A character string specifying the name of the grouping variable.
`vars`	A character vector specifying the names of variables to decompose.
`components`	A character vector specifying which components to compute. Any subset of `c("gmc", "between", "within")` (default: all three). `"gmc"` = grand mean centering, `"between"` = group means, `"within"` = within-group deviations. If `"within"` is requested without `"between"`, the between component is computed internally as an intermediate step and not included in the output.
`gmc_pattern`	A glue-style naming pattern for grand-mean-centered columns. Use `{col}` for the variable name. Default: `"{col}_grand_mean_centered"`.

`between_pattern`
: A glue-style naming pattern for between-group (group mean) columns. Use `{col}` for the variable name and `{group}` for the grouping variable name. Default: `"{col}_between_{group}"`.

`within_pattern`
: A glue-style naming pattern for within-group deviation columns. Use `{col}` for the variable name and `{group}` for the grouping variable name. Default: `"{col}_within_{group}"`.

Details

This decomposition is commonly used in multilevel modeling to separate within-group and between-group variance components (Enders & Tofighi, 2007). The decomposed variables are particularly useful for Random Effects Within-Between (REWB) models (Bell et al., 2019), which allow the estimation of distinct within-group and between-group effects.

The function performs three centering operations:

1. Grand mean centering: Each value is expressed as a deviation from the overall sample mean. This centers the entire distribution at zero.

2. Between-group component: For each observation, this equals the mean of their group. These values are constant within groups and vary between groups. In REWB models, this represents the between-group effect of the predictor.

3. Within-group component: Each value is expressed as a deviation from their group mean. This removes all between-group variance and represents the within-group effect of the predictor in REWB models.

Value

A data frame containing:

- All original variables from `data`
- Grand mean centered versions (named by `gmc_pattern`), if `"gmc"` in `components`
- Between-group means (named by `between_pattern`), if `"between"` in `components`
- Within-group deviations (named by `within_pattern`), if `"within"` in `components`

References

Bell, A., Fairbrother, M., & Jones, K. (2019). Fixed and random effects models: making an informed choice. *Quality & Quantity*, 53(2), 1051-1074.

Enders, C. K., & Tofighi, D. (2007). Centering predictor variables in cross-sectional multilevel models: A new look at an old issue. *Psychological Methods*, 12(2), 121-138.

See Also

`within_between_correlations`, which uses this function internally to perform the within/between decomposition.

Examples

```
data("media_diary")

# Decompose all three components (default)
result <- decompose_within_between(
  data = media_diary,
  group = "person",
  vars = c("stress", "screen_time")
)

# Only between and within (no grand mean centering)
result_wb <- decompose_within_between(
  data = media_diary,
  group = "person",
  vars = c("stress", "screen_time"),
  components = c("between", "within")
)

# Custom column naming: flat suffixes without the group name
result_flat <- decompose_within_between(
  data = media_diary,
  group = "person",
  vars = c("stress", "screen_time"),
  components = c("between", "within"),
  between_pattern = "{col}_between",
  within_pattern = "{col}_within"
)
```

`media_diary` *Simulated daily diary study: entertainment media use and wellbeing*

Description

A simulated daily diary dataset for illustrating multilevel descriptive statistics with **mlstats**. The data mimics a study in which 100 participants completed brief daily surveys for 14 consecutive days, reporting their wellbeing, perceived stress, entertainment media use, and enjoyment on that media. Trait self-control was measured once at the beginning of the study.

The dataset is designed to illustrate the difference between within-person and between-person correlations, including a case where the two go in opposite directions (**screen_time** × **wellbeing**):

- *Within persons*: on days when someone watches more entertainment media than usual, they report slightly better wellbeing — consistent with short-term escapism or mood repair through media use.
- *Between persons*: people who watch more entertainment media on average tend to report lower average wellbeing — chronic heavy media use is associated with lower wellbeing, partly because it reflects lower trait self-control.

The pooled (naive) correlation between **screen_time** and **wellbeing** is near zero, masking both of these real effects.

Usage

```
media_diary
```

Format

A data frame with 1,400 rows and 6 columns:

person Integer person identifier (1–100).

self_control Trait self-control, measured once at study entry (1–7 scale, higher = more self-control). Constant within persons; ICC approximately 1.

wellbeing Daily positive wellbeing (1–7 scale, higher = better).

screen_time Minutes of entertainment media consumed that day (e.g., television, streaming services; non-negative integer).

stress Daily perceived stress (1–7 scale, higher = more stressed).

enjoyment How much the person enjoyed the media they watched that day (1–7 scale, higher = more enjoyment).

Source

Simulated data. Generated by `data-raw/media_diary.R` using a fixed random seed (`set.seed(42)`) for reproducibility. See that script for full simulation details including the intended within- and between-person correlation structure.

Examples

```
data("media_diary")

# Quick look at the structure
str(media_diary)

# Number of persons and observations
length(unique(media_diary$person))  # 100 persons
nrow(media_diary)                   # 1,400 diary entries
```

---

mldesc *Compute Multilevel Descriptive Statistics*

Description

Creates a publication-ready descriptive statistics table for multilevel data (e.g., repeated measurements per person, or students nested within schools). For each variable, the table reports basic descriptives, the proportion of variance that lies between groups (the intraclass correlation, ICC), and how each pair of variables relates both within and between groups (see `within_between_correlations` and `vignette("correlation-methods")` for the statistical background on the latter).

Usage

```
mldesc(
  data,
  group,
  vars,
  method = c("decomposition", "sem", "bayes"),
  weight = TRUE,
  flip = FALSE,
  significance = c("basic", "detailed"),
  ci = 0.9,
  folder = NULL,
  remove_leading_zero = TRUE
)
```

Arguments

data	A data frame containing the variables to analyze.
group	A character string specifying the name of the grouping variable.
vars	A character vector specifying the names of variables to describe.
method	Character string specifying the estimation method for correlations and the ICC: "decomposition" (default), "sem", or "bayes". See within_between_correlations for details on the correlation methods. With method = "bayes", the ICC is also estimated with a Bayesian intercept-only model (via brms::brm) instead of lme4::lmer, reporting the posterior median.
weight	Logical. If TRUE (default), the mean and SD are calculated across all observations (so larger groups contribute more), and the between-group correlation gives more weight to larger groups. If FALSE, every group counts equally: the mean and SD are calculated on group means, and the between-group correlation is unweighted. For correlations, this is only used when method = "decomposition" or method = "bayes".
flip	Logical. If TRUE, between-group correlations are shown in the upper triangle and within-group correlations in the lower triangle. Default is FALSE.
significance	Character string specifying the significance marking style. Either "basic" (default) or "detailed". If "basic", correlations with p < .05 are marked with a star. If "detailed", correlations are marked with 1-3 stars for p < .05, p < .01, or p < .001, respectively. Ignored (with a message) when method = "bayes", which always marks correlations whose credible interval excludes zero with a single star.
ci	Numeric value strictly between 0 and 1 specifying the credible interval width used for the within-group and between-group correlations when method = "bayes". Default is 0.9 (90% CI). The ICC always reports the posterior median only and is not affected by this argument. Ignored (with a message) for other methods.
folder	Character string specifying the directory path where brms models should be saved. Required when method = "bayes"; ignored (with a message) otherwise. Default is NULL.

`remove_leading_zero`
: Logical. If TRUE (default), removes leading zeros from decimal values in correlation and ICC columns according to APA standards.

Details

The function combines three types of information:

Descriptive statistics: Basic summary statistics for each variable. When `weight = TRUE` (default), statistics are calculated across all observations. When `weight = FALSE`, the mean is the mean of group means, and the SD is the standard deviation of group means, representing between-group variability.

Correlations: Within-group correlations (upper triangle) and between-group correlations (lower triangle), computed using `within_between_correlations`. See that function's documentation and the package vignette for how each method estimates these correlations and tests them for significance.

ICC: The intraclass correlation coefficient, computed from an unconditional (intercept-only) multilevel model using `lme4::lmer` (or `brms::brm` when `method = "bayes"`). The ICC represents the proportion of variance in each variable that lies between groups, with values close to 1 indicating a variable that barely varies within groups (e.g., a stable trait), and values close to 0 indicating a variable that barely varies between groups (e.g., a fast-changing state).

The ICC is always computed from a linear (Gaussian) model, regardless of a variable's measurement scale. For binary, ordinal, or count variables this yields a linear-probability-style ICC rather than a latent-scale ICC from a generalized linear mixed model. A warning is emitted if any `vars` look binary, ordinal, or count-like (few, whole-number values).

With `method = "bayes"`, the function fits one `brms` model per variable for the ICCs, plus all the models described in `within_between_correlations` for the correlations — for p variables, p ICC fits in addition to the within/between-group correlation fits. This can take a long time for larger numbers of variables; see `vignette("correlation-methods")` for details.

Value

A tibble of class `mlstats_desc_tibble` containing:

- `variable`: Variable name
- `n_obs`: Number of observations
- `m`: Mean
- `sd`: Standard deviation
- `range`: Range from minimum to maximum
- One column per variable in `vars` containing correlations
- `icc`: Intraclass correlation coefficient

The tibble can be returned as a gt object using `print(result, format = "gt")` and as a tinytable object using `print(result, format = "tt")`.

References

Bürkner, P.-C. (2017). brms: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software, 80*(1), 1–28. doi:10.18637/jss.v080.i01

Pedhazur, E. J. (1997). *Multiple regression in behavioral research: Explanation and prediction*. Harcourt Brace.

Snijders, T. A. B., & Bosker, R. J. (2012). *Multilevel analysis: An introduction to basic and advanced multilevel modeling* (2nd ed.). Sage Publishers.

See Also

within_between_correlations for details on how within-group and between-group correlations are estimated and tested.

Examples

```
data("media_diary")
vars <- c("self_control", "wellbeing", "screen_time", "stress")

# Compute multilevel descriptives (default: decomposition method)
result <- mldesc(
  data = media_diary,
  group = "person",
  vars = vars
)

result

# Compute with unweighted between-group correlations
result_unweighted <- mldesc(
  data = media_diary,
  group = "person",
  vars = vars,
  weight = FALSE
)

# Use SEM-based estimation for correlations (on similarly-scaled variables;
# SEM is sensitive to large scale differences, unlike "decomposition")

result_sem <- mldesc(
  data = media_diary,
  group = "person",
  vars = c("self_control", "wellbeing", "stress"),
  method = "sem"
)


# Use detailed significance marking
result_detailed <- mldesc(
  data = media_diary,
  group = "person",
  vars = vars,
```

```
  significance = "detailed"
)

# Use Bayesian estimation for correlations and the ICC (requires brms)

result_bayes <- mldesc(
  data = media_diary,
  group = "person",
  vars = c("self_control", "wellbeing", "screen_time"),
  method = "bayes",
  folder = tempdir()
)
```

`within_between_correlations` *Compute Within-Group and Between-Group Correlations*

Description

In data with a grouping structure (e.g., repeated measurements per person, or students nested within schools), a single correlation between two variables can be misleading, because it mixes two different relationships: how the variables relate *within* each group (e.g., do a person's good days also tend to be their productive days?), and how they relate *between* groups (e.g., do people who are generally happier also tend to be generally more productive?). This function estimates both relationships separately, using one of three methods (see Details and `vignette("correlation-methods")` for the full statistical background).

Usage

```
within_between_correlations(
  data,
  group,
  vars,
  method = c("decomposition", "sem", "bayes"),
  weight = TRUE,
  flip = FALSE,
  significance = c("basic", "detailed"),
  ci = 0.9,
  folder = NULL
)
```

Arguments

`data`	A data frame containing the variables to analyze.
`group`	A character string specifying the name of the grouping variable.
`vars`	A character vector specifying the names of variables to correlate.

`method`	Character string specifying the estimation method: `"decomposition"` (default), `"sem"`, or `"bayes"`. See Details.
`weight`	Logical. Used when `method = "decomposition"` or `method = "bayes"`. If TRUE (default), the between-group correlation gives more weight to larger groups; significance/credible intervals, however, are always based on the unweighted correlation of group means. If FALSE, every group counts equally regardless of size. Ignored (with a message) when `method = "sem"`, because that method handles unequal group sizes automatically.
`flip`	Logical. If TRUE, between-group correlations are shown in the upper triangle and within-group correlations in the lower triangle. Default is FALSE.
`significance`	Character string specifying the significance marking style. Either "basic" (default) or "detailed". If "basic", correlations with $p < .05$ are marked with a star. If "detailed", correlations are marked with 1-3 stars for $p < .05$, $p < .01$, or $p < .001$, respectively. Ignored (with a message) when `method = "bayes"`, which always marks correlations whose credible interval (see `ci`) excludes zero with a single star.
`ci`	Numeric value strictly between 0 and 1 specifying the credible interval width used to decide whether a correlation is starred. Only applicable when `method = "bayes"`; default is 0.9 (90% CI). Ignored (with a message) for other methods.
`folder`	Character string specifying the directory path where `brms` models should be saved. Required when `method = "bayes"`; ignored (with a message) otherwise. Default is `NULL`.

Details

Method `"decomposition"` (the default) computes the within-group correlation by first subtracting each group's mean from every observation, then correlating the resulting deviation scores. It computes the between-group correlation by correlating the group means with one another (optionally weighted by group size; see `weight`). This approach follows Pedhazur (1997, ch. 16), and the significance tests account for the fact that subtracting group means uses up degrees of freedom, following the general testing principle in Snijders and Bosker (2012, sec. 6.1). This method is fast and easy to interpret, and works well for most data sets, but is less suited to data with very unequal group sizes.

Method `"sem"` fits a two-level structural equation model (via `lavaan::sem()`) that estimates the within-group and between-group covariance matrices simultaneously using maximum likelihood. Significance is based on the resulting z-tests. Because groups are weighted implicitly through maximum likelihood estimation rather than through the `weight` argument, this method is the more principled choice for data with very unequal group sizes or a moderate amount of missing data. It is slower than `"decomposition"` and can occasionally fail to converge for small or collinear data sets.

For `method = "sem"`, variables that never vary within a group (e.g., time-invariant traits) are modeled only at the between-group level, and variables with almost no between-group variance (intraclass correlation near zero) are modeled only at the within-group level; the corresponding cells of the unused level are reported as `NA`.

Method `"bayes"` mirrors `"decomposition"`, but estimates both correlations via Bayesian multivariate models fit with `brms::brm()` (requires the **brms** package) instead of closed-form formulas,

reporting posterior medians and credible intervals (via `ci`) in place of point estimates and p-values. It requires a `folder` argument to cache fitted models, can take considerably longer than the other two methods, and is most useful when the number of groups is small or when communicating uncertainty via credible intervals is a priority. See `vignette("correlation-methods")` for details on the number of models fit and caching behavior.

Value

A tibble containing a correlation matrix where:

- The upper triangle contains within-group correlations
- The lower triangle contains between-group correlations
- Diagonal elements are marked with "–"
- Significant correlations are marked with asterisks (see `significance` parameter, or `ci` when `method = "bayes"`)

The tibble can be returned as a gt object using `print(result, format = "gt")` and as a tinytable object using `print(result, format = "tt")`.

References

Bürkner, P.-C. (2017). brms: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software, 80*(1), 1–28. doi:10.18637/jss.v080.i01

Hox, J., Moerbeek, M., & van de Schoot, R. (2018). *Multilevel analysis: Techniques and applications* (3rd ed.). Routledge.

Pedhazur, E. J. (1997). *Multiple regression in behavioral research: Explanation and prediction*. Harcourt Brace.

Snijders, T. A. B., & Bosker, R. J. (2012). *Multilevel analysis: An introduction to basic and advanced multilevel modeling* (2nd ed.). Sage Publishers.

See Also

`mldesc`, which combines this function's output with descriptive statistics and ICCs in a single table. See `vignette("correlation-methods")` for a detailed statistical description of all three methods.

Examples

```
data("media_diary")

# Compute weighted between-group correlations (default, decomposition method)
result_weighted <- within_between_correlations(
  data = media_diary,
  group = "person",
  vars = c("wellbeing", "screen_time")
)

# Compute unweighted between-group correlations
result_unweighted <- within_between_correlations(
  data = media_diary,
```

```
  group = "person",
  vars = c("wellbeing", "screen_time"),
  weight = FALSE
)

# Use SEM-based estimation (on similarly-scaled variables; SEM is
# sensitive to large scale differences, unlike "decomposition")

result_sem <- within_between_correlations(
  data = media_diary,
  group = "person",
  vars = c("wellbeing", "stress"),
  method = "sem"
)


# Use detailed significance marking
result_detailed <- within_between_correlations(
  data = media_diary,
  group = "person",
  vars = c("wellbeing", "screen_time"),
  significance = "detailed"
)

# Use Bayesian estimation (requires the brms package)

result_bayes <- within_between_correlations(
  data = media_diary,
  group = "person",
  vars = c("wellbeing", "screen_time"),
  method = "bayes",
  folder = tempdir()
)
```

Index