Package 'wbCorr'

October 31, 2026

Title Bivariate Within- and Between-Cluster Correlations

Version 0.4.0

Description Separates supplied variables into within- and between-cluster components and calculates bivariate correlations for each level separately. For Pearson correlations, the centered-score decomposition corresponds to commonly used between- and within-cluster correlations reviewed by Tu et al. (2025) <doi:10.1002/sim.10326>. The package's descriptive Spearman option is distinct from the clustered rank parameters introduced in that paper. The package is also motivated by the distinction between within- and between-person variation described by Curran and Bauer (2011) <doi:10.1146/annurev.psych.093008.100356> and by Hamaker (2024) <doi:10.1080/00273171.2022.2155930>. The package is intended for longitudinal or otherwise clustered data where researchers need transparent correlation matrices before fitting more complex multilevel models.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 3.5.0)

Suggests httr, psych, spelling, testthat (>= 3.0.0)

Imports methods, writexl

Config/testthat/edition 3

Language en-US

URL https://pascal-kueng.github.io/wbCorr/, https://github.com/Pascal-Kueng/wbCorr

BugReports https://github.com/Pascal-Kueng/wbCorr/issues

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Pascal Küng [aut, cre, cph] (ORCID: <https://orcid.org/0000-0001-7346-9414>)

Maintainer Pascal Küng <pascal.kueng@psychologie.uzh.ch>

Repository CRAN

Date/Publication 2026-10-31 15:10:02 UTC

Contents

get_ICC 2
get_matrix 3
get_table 4
plot,wbCorr-method 5
print,wbCorr-method 7
show,wbCorr-method 7
simdat_intensive_longitudinal 8
to_excel 9
update_wbCorr 10
wbCorr 11
wbCorr-class 13

Index **14**

`get_ICC` *Return all ICCs for the original variables.*

Description

You can use get_ICC() or get_ICCs() interchangeably.

Usage

```
get_ICC(object)

get_ICCs(object)

get_icc(object)
```

Arguments

`object`	A wbCorr object, created by the wbCorr() function.

Value

A dataframe with ICCs for all variables. The ICC(1,1) of the one-way random-effects model is estimated from ANOVA mean squares, using all available observations of each variable. Sample estimates can be negative.

See Also

`wbCorr`

Examples

```
# importing our simulated example dataset with pre-specified within- and between- correlations
data("simdat_intensive_longitudinal")

# create object:
correlations <- wbCorr(data = simdat_intensive_longitudinal,
                       cluster = 'participantID')

# returns the ICCs:
ICCs <- get_ICC(correlations)
print(ICCs)
```

`get_matrix` *Return matrices for within- and/or between-cluster correlations.*

Description

You can use summary(), get_matrices(), or get_matrix() interchangeably. Merged matrices include the ICC on the diagonal. For more detailed statistics, use get_table().

Usage

```
get_matrix(object, which = c("within", "between", "merge"), ...)

get_matrices(object, which = c("within", "between", "merge"), ...)

## S3 method for class 'wbCorr'
summary(object, ...)

## S4 method for signature 'wbCorr'
summary(object, which = c("within", "between", "merge"), ...)
```

Arguments

`object`	A wbCorr object, created by the wbCorr() function.
`which`	A string or a character vector indicating which summaries to return. Options are 'within' or 'w', 'between' or 'b', and various merge options like 'merge', 'm', 'merge_wb', 'wb', 'merge_bw', 'bw'. Default is c('within', 'between', 'merge').
`...`	Additional arguments passed to the base summary method

Value

A list containing the selected matrices of within- and/or between-cluster correlations, and ICCs on the diagonals for merged matrices.

See Also

get_tables, wbCorr

Examples

```
# importing our simulated example dataset with pre-specified within- and between- correlations
data("simdat_intensive_longitudinal")

# create object:
correlations <- wbCorr(data = simdat_intensive_longitudinal,
                       cluster = 'participantID')

# returns a correlation matrix with stars for p-values:
matrices <- summary(correlations) # the get_matrix() and get_matrices() functions are equivalent
print(matrices)

# Access specific matrices by:
# Option 1:
matrices$within
# Option 2:
within_matrix <- summary(correlations, which = 'w') # or use 'within'
merged_within_between <- summary(correlations, which = 'wb')
print(within_matrix) # could be saved to an excel or csv file (e.g., write.csv)
```

---

get_table *Retrieve full tables for both within- and/or between-cluster correlations for a wbCorr object.*

Description

This function has an alias get_tables() which can be used interchangeably. For correlations matrices, see the summary() function.

Usage

```
get_table(object, which = c("within", "between"))

get_tables(object, which = c("within", "between"))
```

Arguments

object	A wbCorr object, created by the wbCorr() function.
which	A character vector indicating which correlation table to return. Options are 'within' or 'w', and 'between' or 'b'.

Value

A list containing the selected tables of within- and/or between-cluster correlations.

See Also

summary, wbCorr

Examples

```
# importing our simulated example dataset with pre-specified within- and between- correlations
data("simdat_intensive_longitudinal")

# create object:
correlations <- wbCorr(data = simdat_intensive_longitudinal,
                       cluster = 'participantID')

# returns a list with full detailed tables of the correlations:
tables <- get_table(correlations) # the get_tables() function is equivalent
print(tables)

# Access specific tables by:
# Option 1:
tables$between
# Option 2:
within_table <- get_tables(correlations, which = 'w') # or use 'within' or 'between'
print(within_table) # within_table could be saved to an excel or csv file (e.g., write.csv)
```

---

plot,wbCorr-method *Plot within- and between associations*

Description

Plots the centered variables of the provided dataframe against each other. Choose whether to plot the between-centered variables (representing the between-cluster correlations by plotting cluster means) or the within-centered variables (representing the within-cluster correlations by plotting deviations from person-means). Panels below the diagonal show scatterplots with a regression line. Panels above the diagonal show the correlation from the wbCorr object with its significance stars.

Usage

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

## S4 method for signature 'wbCorr'
plot(
  x,
  y,
  which = NULL,
  plot_NA = TRUE,
  standardize = TRUE,
  outlier_detection = "zscore",
```

```
  outlier_threshold = "recommended",
  type = "p",
  pch = 20,
  dot_lwd = 2,
  reg_lwd = 2,
  ...
)
```

Arguments

x	A wbCorr object to be plotted.
y	Choose which correlations to plot ('within' / 'w' or 'between' / 'b'); can be used as a positional argument.
...	further options to be passed to the base plot (pairs) function.
which	Can be used as an alternative to 'y' (e.g., which = 'w'). It has the same functionality as 'y', but takes precedence if both are specified.
plot_NA	Boolean. Whether variables that have no variation on the selected level should be plotted or not.
standardize	Boolean. Whether the dataset should be standardized before plotting.
outlier_detection	If FALSE, outliers will not be marked in red. Otherwise you may provide the method. Choose from: 'zscore', 'mad', or 'tukey'.
outlier_threshold	If 'recommended', the threshold for 'zscore' and 'mad' will be set to 3, and for 'tukey' to 1.5. You can provide and other numeric here.
type	points, lines, etc. see ?base::plot for available types).
pch	Graphical parameter. Select which type of points should be plotted.
dot_lwd	Graphical parameter. Set size of the points.
reg_lwd	Graphical parameter. Set thickness of the regression line.

Value

Called for the side effect of drawing a pairs plot; returns NULL invisibly.

See Also

wbCorr

print,wbCorr-method *Print Method for the wbCorr Class*

Description

Prints a summary of the wbCorr object.

Usage

```
## S4 method for signature 'wbCorr'
print(x, ...)
```

Arguments

x	A wbCorr object.
...	Additional arguments, currently unused.

Value

Invisibly returns the supplied wbCorr object. Called for the side effect of printing a compact summary of the within-cluster table, between-cluster table, and ICC table.

See Also

wbCorr

Examples

```
# Example
data("simdat_intensive_longitudinal")
correlations <- wbCorr(simdat_intensive_longitudinal,
                       cluster = 'participantID',
                       confidence_level = 0.95,
                       method = 'spearman')
print(correlations)
```

show,wbCorr-method *Show Method for the wbCorr Class*

Description

Shows a summary of the wbCorr object, equivalent to the print method.

Usage

```
## S4 method for signature 'wbCorr'
show(object)
```

Arguments

`object`	A `wbCorr` object.

Value

Invisibly returns the supplied `wbCorr` object. Called for the side effect of showing the same compact summary as `print()`.

See Also

`wbCorr`, `print.wbCorr`

Examples

```
# Example using the iris dataset
cors <- wbCorr(iris, iris$Species, between_weighting = 'cluster_size')
show(cors)
```

---

simdat_intensive_longitudinal

Simulated Intensive Longitudinal Dataset

Description

A simulated intensive longitudinal dataset to test the package capabilities. This dataset contains 5,000 observations from 100 participants measured on 50 days, plus three variables (var1, var2, and var3) that are correlated at both the within- and between-person levels.

Format

A data frame with 5,000 rows and the following five columns:

participantID Identifier for each participant (integer)

day Day variable varying only within-person (integer)

var1 Variable 1 (numerical)

var2 Variable 2 (numerical)

var3 Variable 3 (numerical)

Details

The within-person correlations are all positive:

- var1 & var2: 0.1
- var1 & var3: 0.3
- var2 & var3: 0.8

The between-person correlations are all negative:

- var1 & var2: -0.5
- var1 & var3: -0.4
- var2 & var3: -0.2

Time trends (within):

- var1 & time: 0.0
- var2 & time: 0.0
- var3 & time: 0.4

Source

A simulated dataset by P. Küng

---

to_excel *Saves the passed summary or table to excel*

Description

Use `to_excel(get_matrix(wbCorrObject))` or `to_excel(get_table(wbCorrObject))` to save the provided table/matrix to an excel file.

Usage

```
to_excel(SummaryObject, path = file.path(getwd(), "wbCorr.xlsx"))
```

Arguments

`SummaryObject`	A summary or matrix object, such as those returned by `get_matrix()` or `get_table()`, or a single table.
`path`	Specify the filename and a path. If no path is provided, the file will be saved to the current working directory.

Value

Writes an Excel file (.xlsx) to disk and invisibly returns its path.

See Also

get_tables, wbCorr, get_matrix

Examples

```
# Importing our simulated example dataset with pre-specified within- and between- correlations
data("simdat_intensive_longitudinal")

# Create object:
correlations <- wbCorr(data = simdat_intensive_longitudinal,
                       cluster = 'participantID')

# Returns a correlation matrix with stars for p-values:
matrices <- get_matrix(correlations) # summary(correlations) works too.

to_excel(matrices, path = tempfile(fileext = ".xlsx"))
```

---

update_wbCorr *Check for updates of wbCorr*

Description

This function checks if there is a newer version on GitHub by comparing the version numbers in the local and remote DESCRIPTION files. It only runs when called explicitly by the user and does not install updates.

Usage

```
update_wbCorr(ask = FALSE)
```

Arguments

ask	Deprecated and ignored.

Value

An integer: 1 if there's a newer version available, 0 if the current version is the latest, or NULL if there was an error accessing the remote DESCRIPTION file.

wbCorr *wbCorr*

Description

The wbCorr function creates a wbCorr object containing within- and between-cluster correlations, p-values, and confidence intervals for a given dataset and clustering variable. The object can be plotted.

Usage

```
wbCorr(
  data,
  cluster,
  confidence_level = 0.95,
  method = "pearson",
  nboot = 1000,
  inference = c("analytic", "none", "cluster_bootstrap"),
  between_weighting = c("equal_clusters", "cluster_size")
)

wbcorr(
  data,
  cluster,
  confidence_level = 0.95,
  method = "pearson",
  nboot = 1000,
  inference = c("analytic", "none", "cluster_bootstrap"),
  between_weighting = c("equal_clusters", "cluster_size")
)
```

Arguments

data	A dataframe containing numeric variables for which correlations will be calculated.
cluster	A vector representing the clustering variable or a string with the name of the column in data that contains the clustering variable.
confidence_level	A numeric value between 0 and 1 representing the desired level of confidence for confidence intervals (default: 0.95).
method	A string indicating the correlation method to be used. Supported methods are 'pearson', 'spearman', and 'auto' (default: 'pearson'). 'pearson': p-values use t-statistics and confidence intervals use the Fisher z-transformation. 'spearman': p-values and confidence intervals use the Fisher z-transformation. 'auto': uses Spearman for pairs with an ordered factor and Pearson otherwise (reported as point-biserial or phi coefficient for binary variables).

`nboot` Specifies the amount of bootstrap samples (default: 1000).

`inference` A string specifying how p-values and confidence intervals are computed. 'analytic' (default) uses the formulas described under `method`. 'cluster_bootstrap' resamples whole clusters and computes bias-corrected (BC) bootstrap confidence intervals and p-values. Unlike the analytic tests, it accounts for dependence within clusters and does not assume normal data. Use `set.seed()` for reproducible results. 'none' returns the correlation coefficients only.

`between_weighting` A string. If 'equal_clusters' (default), every cluster mean is weighted the same in between-cluster correlations, even if some clusters contributed fewer measurement points. If 'cluster_size', cluster means are weighted by the number of observations each cluster has of both variables. These methods will be equivalent in datasets without missing data and an equal number of measurements per cluster.

Details

Calculates bivariate within- and between-cluster correlations for clustered data, such as repeated measures nested in persons, dyads, teams, or other groups. Only recommended for continuous or binary variables.

Inspired by the psych::statsBy function, wbCorr allows you to easily calculate, extract, and plot within- and between-cluster correlations for further analysis.

Each variable is split into its cluster means (between-cluster part) and the deviations from these means (within-cluster part). Cluster means use all available observations of a variable. Each correlation then uses all complete pairs of the two variables.

Within-cluster tests use N - k - 1 degrees of freedom, where N is the number of complete pairs and k the number of clusters they come from. These tests treat the deviations as independent, which is an approximation. Between-cluster tests use k - 2 degrees of freedom. A correlation is `NA` if there are too few observations, e.g. fewer than three clusters for between-cluster correlations.

Value

A wbCorr object that contains within- and between-cluster statistics. Use the get_table() function on the wbCorr object to retrieve a list of the full correlation tables. Use the summary() or get_matrix() function on the wbCorr object to retrieve various correlation matrices, including ICCs in the merged ones. Use get_ICC() to retrieve all intraclass correlations (ICC(1,1)). Finally, use to_excel() on a table or matrix (or list of matrices) to save them.

See Also

`get_table`, `summary`, `get_ICC`, `plot`, `to_excel`

Examples

```
# importing our simulated example dataset with pre-specified within- and between- correlations
data("simdat_intensive_longitudinal")

# create a wbCorr object:
```

```
correlations <- wbCorr(simdat_intensive_longitudinal,
                       'participantID')

# optionally weight between-cluster correlations by cluster size:
weighted_correlations <- wbCorr(simdat_intensive_longitudinal,
                       'participantID',
                       between_weighting = 'cluster_size')

# cluster bootstrap; use more bootstrap samples in applied work:

bootstrapped_correlations <- wbCorr(simdat_intensive_longitudinal,
                       'participantID',
                       inference = 'cluster_bootstrap',
                       nboot = 100)


# returns a list with full detailed tables of the correlations:
tables <- get_table(correlations) # the get_tables() function is equivalent
print(tables)

# returns a correlation matrix with stars for p-values:
matrices <- summary(correlations) # the get_matrix() and get_matrices() functions are equivalent
print(matrices)

# Plot the centered variables against each other
plot(correlations, 'within')
plot(correlations, which = 'b')

# Store the list of correlation matrices to excel
to_excel(matrices, path = tempfile(fileext = ".xlsx"))
```

---

wbCorr-class *wbCorr Class*

Description

A class representing within- and between-cluster correlations.

Details

The wbCorr class is used to store within- and between-cluster correlations and provides methods for printing and summarizing the correlations.

See Also

wbCorr

Index

get_ICC, 2, *12*
get_icc *(*get_ICC*)*, 2
get_ICCs *(*get_ICC*)*, 2
get_matrices *(*get_matrix*)*, 3
get_matrix, 3, *10*
get_table, 4, *12*
get_tables, *4*, *10*
get_tables *(*get_table*)*, 4

plot, *12*
plot,wbCorr-method, 5
plot.wbCorr *(*plot,wbCorr-method*)*, 5
print,wbCorr-method, 7
print.wbCorr, *8*
print.wbCorr *(*print,wbCorr-method*)*, 7

show,wbCorr-method, 7
show.wbCorr *(*show,wbCorr-method*)*, 7
simdat_intensive_longitudinal, 8
summary, *5*, *12*
summary,wbCorr-method *(*get_matrix*)*, 3
summary.wbCorr *(*get_matrix*)*, 3

to_excel, 9, *12*

update_wbCorr, 10

wbCorr, *2*, *4–8*, *10*, 11, *13*
wbcorr *(*wbCorr*)*, 11
wbCorr-class, 13