

Package: OmniCorr (via r-universe)

September 24, 2026

Title Multi-omics Integration and Visualization Framework

Version 0.99.0

Description OmniCorr is an R package for correlation-based integration and visualization of multi-omics datasets, including transcriptomics, metatranscriptomics, metagenomics, metaproteomics, and associated sample metadata. The package computes pairwise correlations between independently processed omics layers and provides flexible tools for organizing and visualizing cross-omics associations using aligned heatmaps and optional network-style visualizations. OmniCorr is designed to complement existing preprocessing and network inference workflows rather than replace them.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

Depends R (>= 4.5.0)

Imports ComplexHeatmap, circlize, RColorBrewer, WGCNA, grid

Suggests cowplot, knitr, pheatmap, rmarkdown, testthat

VignetteBuilder knitr

biocViews DimensionReduction, FeatureExtraction, Metabolomics, Metagenomics, Microbiome, NetworkInference, Preprocessing, Proteomics, Transcriptomics, Software

URL <https://github.com/veronicaquarato/OmniCorr>

BugReports <https://github.com/veronicaquarato/OmniCorr/issues>

Config/testthat/edition 3

Config/roxygen2/version 8.1.0

Config/pak/sysreqs cmake make libicu-dev libpng-dev libuv1-dev perl

Repository <https://biocstaging.r-universe.dev>

Date/Publication 2026-09-24 14:39:43 UTC

RemoteUrl <https://github.com/BiocStaging/OmniCorr>

RemoteRef HEAD
RemoteSha 8085328a362eeecb10d901d0cdd02f1f31752cc

Contents

calculate_correlations	2
CheckSampleOrder	3
metadata	4
Metagenomics	5
Metatranscriptomics	5
run_omnicorr	6
Transcriptomics	8
Index	9

calculate_correlations
<i>Calculate correlation between two data frames</i>

Description

This function calculates the correlation matrix between the columns of two data frames.

Usage

```
calculate_correlations(  
  df1,  
  df2,  
  method = "pearson",  
  adjust_method = "fdr",  
  use = "all.obs",  
  show_significance = "stars"  
)
```

Arguments

df1	A data frame or matrix containing the first set of variables to be correlated. Rows are samples, columns are variables (e.g., genes, proteins, metabolites).
df2	A data frame or matrix containing the second set of variables to be correlated. Rows are samples, columns are variables.
method	The correlation method to be used. Default is "pearson". Other options include "spearman" and "kendall".
adjust_method	Method for adjusting the p-values for multiple testing. Default is "fdr". If "none", no adjustment is applied. Supports all methods from p.adjust.methods.
use	Handling of missing data. One of "all.obs" or "pairwise.complete.obs".

show_significance

How to represent significance in the output. One of "stars", "p_value", or "correlation".

Details

Pearson and Spearman correlations are computed using WGCNA::corAndPvalue for efficiency. Kendall correlation uses cor.test in a loop and may be slow for large datasets.

Value

A list containing:

correlation Matrix of correlation coefficients (rounded to 3 decimals).

p_value Matrix of raw p-values.

p_value_adj Matrix of adjusted p-values.

signif_matrix Matrix of significance stars, p-values, or correlations.

Examples

```
data(Transcriptomics)
data(Metagenomics)

result_list <- calculate_correlations(
  df1 = Transcriptomics,
  df2 = Metagenomics,
  show_significance = "p_value"
)
```

CheckSampleOrder	<i>Check and align sample order between two datasets</i>
------------------	--

Description

Ensures that two data frames or matrices contain identical samples and reorders the second dataset to match the first.

Usage

```
CheckSampleOrder(df1, df2)
```

Arguments

df1 A data frame or matrix with samples in rows (reference order)

df2 A data frame or matrix with samples in rows

Value

A list with df1 and df2 reordered to have identical rownames and order

Examples

```
df1 <- matrix(1:6, nrow = 3,
              dimnames = list(c("S1", "S2", "S3"), c("A", "B")))
df2 <- matrix(7:12, nrow = 3,
              dimnames = list(c("S3", "S1", "S2"), c("C", "D")))
result <- CheckSampleOrder(df1, df2)
result$df2
```

metadata

Metadata information

Description

This data frame contains information about the samples used in the study.

Usage

```
metadata
```

Format

A data frame with 14 variables:

BW Body weight of the sample in grams.

Length Length of the sample in centimeters.

Sex Sex of the sample.

Gut_struc Structural condition of the gut of the sample.

Gut_light Light coloration of the gut of the sample.

Anus_bleed Presence or absence of anus bleeding in the sample.

Gut_bleed Presence or absence of gut bleeding in the sample.

Fat_Score Score of fat deposition in the sample.

Heart_Sco Score of heart condition in the sample.

softliver Presence or absence of soft liver in the sample.

patchy_liver Presence or absence of patchy liver in the sample.

Liver_Sco Score of liver condition in the sample.

HSI Hepatosomatic index of the sample.

CSI Cardiosomatic index of the sample.

Value

A data frame containing sample metadata.

Source

Information was collected from the study samples.

Examples

```
# View the first few rows of the metadata
head(metadata)
```

Metagenomics	<i>Example Metagenomics dataset</i>
--------------	-------------------------------------

Description

A processed metagenomics feature-abundance matrix used as an example input for OmniCorr.

Usage

```
data(Metagenomics)
```

Format

A numeric matrix with samples as rows and features as columns.

Source

Derived from published cow rumen multi-omics analysis.

Metatranscriptomics	<i>Example Metatranscriptomics dataset</i>
---------------------	--

Description

A processed metatranscriptomics feature-abundance matrix used as an example input for OmniCorr.

Usage

```
data(Metatranscriptomics)
```

Format

A numeric matrix with samples as rows and features as columns.

Source

Derived from published cow rumen multi-omics analysis.

run_omnicorr

Run OmniCorr Multi-Omics Integration

Description

Integrates multiple omics layers by computing correlations between a reference omics dataset and multiple comparison layers and visualizing the results as aligned multi-panel heatmaps using **ComplexHeatmap**.

The reference layer is displayed as an expression heatmap with optional clustering of features and samples. Correlation heatmaps for comparison layers and metadata are aligned to the same reference feature order to ensure consistent interpretation across panels.

Usage

```
run_omnicorr(
  reference_layer,
  comparison_layers,
  metadata = NULL,
  reference_name = "Reference Layer",
  comparison_names = NULL,
  metadata_name = "Metadata",
  method = "pearson",
  adjust_method = "fdr",
  use = "all.obs",
  show_significance = "stars",
  cluster_reference_rows = TRUE,
  cluster_reference_columns = TRUE,
  show_row_names = TRUE,
  show_column_names = TRUE,
  font_size = 8,
  star_font_size = 8,
  verbose = TRUE
)
```

Arguments

reference_layer	Numeric matrix/data.frame (samples x features).
comparison_layers	Named list of numeric matrices/data.frames.
metadata	Optional numeric matrix/data.frame (samples x variables).
reference_name	Title for the reference heatmap.

comparison_names Optional titles for comparison heatmaps.
 metadata_name Title for metadata heatmap.
 method Correlation method: "pearson", "spearman", "kendall".
 adjust_method P-value correction method (stats::p.adjust.methods).
 use Missing value handling strategy ("all.obs" or "pairwise.complete.obs").
 show_significance "stars", "p_value", or "correlation".
 cluster_reference_rows Cluster reference features.
 cluster_reference_columns Cluster reference samples.
 show_row_names Logical; display row labels.
 show_column_names Logical; display column labels.
 font_size Font size for row/column labels.
 star_font_size Font size for cell annotations.
 verbose Print progress messages.

Value

Invisibly returns a `ComplexHeatmap::HeatmapList`.

Examples

```

reference <- matrix(
  rnorm(20),
  nrow = 5,
  dimnames = list(paste0("S", 1:5), paste0("Gene", 1:4))
)
comparison <- matrix(
  rnorm(15),
  nrow = 5,
  dimnames = list(paste0("S", 1:5), paste0("Taxon", 1:3))
)
result <- run_omnicorr(
  reference_layer = reference,
  comparison_layers = list(Microbiome = comparison)
)

```

Transcriptomics	<i>Example Transcriptomics dataset</i>
-----------------	--

Description

A processed transcriptomics expression matrix used as an example input for OmniCorr. Rows correspond to samples and columns to transcript-level features.

Usage

```
data(Transcriptomics)
```

Format

A numeric matrix with samples as rows and features as columns.

Source

Derived from published cow rumen multi-omics analysis.

Index

* **datasets**

Metagenomics, [5](#)

Metatranscriptomics, [5](#)

Transcriptomics, [8](#)

calculate_correlations, [2](#)

CheckSampleOrder, [3](#)

metadata, [4](#)

Metagenomics, [5](#)

Metatranscriptomics, [5](#)

run_omnicorr, [6](#)

Transcriptomics, [8](#)