Package 'OmniCorr'

November 25, 2026

Title Multi-Omics Integration and Visualization Framework

Version 1.0.0

Description Provides tools 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 (>= 4.6.0)

Imports ComplexHeatmap, SummarizedExperiment, circlize, RColorBrewer, WGCNA, grid

Suggests cowplot, knitr, pheatmap, rmarkdown, testthat

VignetteBuilder knitr

biocViews DimensionReduction, FeatureExtraction, Metabolomics, Metagenomics, Microbiome, Network, 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 8.1.1

git_url https://git.bioconductor.org/packages/OmniCorr

git_branch RELEASE_3_24

git_last_commit 8db8489

git_last_commit_date 2026-11-23

Repository Bioconductor 3.24

Date/Publication 2026-11-24

Author Shashank Gupta [aut] (ORCID: <https://orcid.org/0000-0001-8839-7003>),
Veronica Quarato [aut, cre] (ORCID:
<https://orcid.org/0009-0008-4049-9746>),
Wanxin Lai [aut],
Research Council of Norway [fnd] (Project 300846),
European Union [fnd] (Horizon 2020 ERA-Net Cofund BlueBio (311913);
MSCA DN HoloGen (101169005)),
Novo Nordisk Foundation [fnd] (Project 0054575-SuPAcow)

Maintainer Veronica Quarato <vequa7472@nmbu.no>

Contents

OmniCorr-package 2
calculate_correlations 3
check_sample_order 4
metadata 5
Metagenomics 6
Metatranscriptomics 6
run_omnicorr 7
Transcriptomics 9

Index **10**

`OmniCorr-package` *OmniCorr: Multi-omics Integration and Visualization Framework*

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.

Author(s)

Maintainer: Veronica Quarato <vequa7472@nmbu.no> (ORCID)

Authors:

- Veronica Quarato <vequa7472@nmbu.no> (ORCID)
- Shashank Gupta <bioinformaticssrm2011@gmail.com> (ORCID)
- Wanxin Lai

Other contributors:

- Research Council of Norway (Project 300846) [funder]
- European Union (Horizon 2020 ERA-Net Cofund BlueBio (311913); MSCA DN HoloGen (101169005)) [funder]
- Novo Nordisk Foundation (Project 0054575-SuPAcow) [funder]

See Also

Useful links:

- https://github.com/veronicaquarato/OmniCorr
- Report bugs at https://github.com/veronicaquarato/OmniCorr/issues

---

`calculate_correlations` *Calculate correlations between two omics datasets*

Description

This function calculates correlations between variables from two datasets provided as data frames, matrices, or SummarizedExperiment objects.

Usage

```
calculate_correlations(
  df1,
  df2,
  method = c("pearson", "spearman", "kendall"),
  adjust_method = c("fdr", setdiff(stats::p.adjust.methods, "fdr")),
  use = c("all.obs", "pairwise.complete.obs"),
  show_significance = c("stars", "p_value", "correlation")
)
```

Arguments

`df1`	A data frame, matrix, or SummarizedExperiment containing the first set of variables. For matrices and data frames, rows are samples and columns are variables. For SummarizedExperiment objects, the first assay is extracted and transposed automatically.
`df2`	A data frame, matrix, or SummarizedExperiment containing the second set of variables, following the same conventions as df1.
`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. Default is "all.obs". One of "all.obs" or "pairwise.complete.obs".
`show_significance`	How to represent significance in the output. Default is "stars". One of "stars", "p_value", or "correlation".

Details

SummarizedExperiment inputs are supported by extracting and transposing their first assay. Assays are converted to ordinary in-memory matrices before correlation analysis, including delayed or disk-backed assays.

This function is primarily intended for correlations between module eigengenes or other reduced-dimensional omics features. Large feature matrices may require substantial memory.

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"
)
```

---

check_sample_order *Check and align sample order between two datasets*

Description

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

Usage

```
check_sample_order(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 <- check_sample_order(df1, df2)
result$df2
```

---

metadata *Metadata information*

Description

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

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.

Details

Sample metadata from the salmon multi-omics study. The dataset contains 36 samples and 14 variables. Rows represent samples, identified by sample IDs.

Value

A data frame containing 36 samples (rows) and 14 metadata variables (columns).

Source

Information was collected from the study samples.

Examples

```
data("metadata", package = "OmniCorr")
head(metadata)
```

`Metagenomics` *Example Metagenomics dataset*

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` *Salmon Metatranscriptomics Dataset*

Description

A processed metatranscriptomics dataset containing WGCNA module eigengenes derived from salmon metatranscriptomic data.

Usage

```
data(Metatranscriptomics)
```

Format

A data.frame with 36 samples (rows) and 11 module eigengenes (columns).

Details

The original count data were variance-stabilized using DESeq2, followed by weighted gene co-expression network analysis (WGCNA). Module eigengenes were averaged at the tank level.

Source

Salmon metatranscriptomics data provided by Shashank Gupta. The input dataset is available in `raw-data/metatranscriptomics/metatranscriptomics.rds`. The reproducibility script is available in `inst/scripts/generate_salmon_metatranscriptomics.R`.

`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 = c("pearson", "spearman", "kendall"),
  adjust_method = c("fdr", setdiff(stats::p.adjust.methods, "fdr")),
  use = c("all.obs", "pairwise.complete.obs"),
  show_significance = c("stars", "p_value", "correlation"),
  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`
: A matrix, data frame, or SummarizedExperiment containing the reference omics dataset. For matrices and data frames, rows represent samples and columns represent features. For SummarizedExperiment objects, the first assay is extracted and transposed automatically.

`comparison_layers`
: A named list of matrices, data frames, or SummarizedExperiment objects containing the omics datasets to compare with the reference layer. Different input types can be combined in the same analysis. SummarizedExperiment objects are converted using their first assay.

`metadata`
: Optional numeric matrix or data frame containing sample metadata (samples in rows, variables in columns). Row names must match the reference layer sample identifiers in the same order. When provided, metadata variables are correlated with reference features and displayed in an additional heatmap. Default is NULL.

`reference_name` Title for the reference heatmap. Default is "Reference Layer".

`comparison_names`
Optional titles for comparison heatmaps. Default is NULL.

`metadata_name` Title for metadata heatmap. Default is "Metadata".

`method` Correlation method: "pearson", "spearman", or "kendall". Default is "pearson".

`adjust_method` P-value correction method (stats::p.adjust.methods). Default is "fdr".

`use` Missing value handling: "all.obs" or "pairwise.complete.obs". Default is "all.obs". If missing values are detected, the function automatically switches to "pairwise.complete.obs" with a warning.

`show_significance`
"stars", "p_value", or "correlation". Default is "stars".

`cluster_reference_rows`
Cluster reference features. Default is TRUE.

`cluster_reference_columns`
Cluster reference samples. Default is TRUE.

`show_row_names` Logical; display row labels. Default is TRUE.

`show_column_names`
Logical; display column labels. Default is TRUE.

`font_size` Font size for row/column labels. Default is 8.

`star_font_size` Font size for cell annotations. Default is 8.

`verbose` Print progress messages. Default is TRUE.

Details

All omics layers must contain the same samples in the same order. Sample identifiers are checked using row names after conversion. For SummarizedExperiment objects, these identifiers correspond to the column names of the original object.

If a SummarizedExperiment contains multiple assays, only the first assay is used. To analyze another assay, extract it explicitly using SummarizedExperiment::assay() and transpose it before passing it to run_omnicorr().

SummarizedExperiment assays are converted to ordinary in-memory matrices before analysis. Delayed or disk-backed assays are therefore fully loaded into memory. OmniCorr is primarily intended for integrating module eigengenes or other reduced-dimensional omics representations rather than large, unprocessed feature matrices.

Value

Invisibly returns a `ComplexHeatmap::HeatmapList`.

Examples

```
# Create example datasets with matching sample IDs
set.seed(123)
samples <- paste0("Sample", 1:10)

reference <- matrix(
  rnorm(10 * 5),
  nrow = 10,
  dimnames = list(samples, paste0("Ref", 1:5))
)
```

```
comparison <- matrix(
  rnorm(10 * 4),
  nrow = 10,
  dimnames = list(samples, paste0("Comp", 1:4))
)

# Generate the multi-omics heatmap
heatmap <- run_omnicorr(
  reference_layer = reference,
  comparison_layers = list(Proteomics = comparison),
  verbose = FALSE
)
```

`Transcriptomics` *Example Transcriptomics dataset*

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, 6
Metatranscriptomics, 6
Transcriptomics, 9
∗ **internal**
OmniCorr-package, 2

calculate_correlations, 3
check_sample_order, 4

metadata, 5
Metagenomics, 6
Metatranscriptomics, 6

OmniCorr *(*OmniCorr-package*)*, 2
OmniCorr-package, 2

run_omnicorr, 7

Transcriptomics, 9