# Package: anglemania (via r-universe)

October 29, 2026

**URL** https://github.com/BIMSBbioinfo/anglemania/

**Title** Feature Extraction for scRNA-seq Dataset Integration

**Version** 1.2.0

**Description** anglemania extracts genes from multi-batch scRNA-seq experiments for downstream dataset integration. It shows improvement over the conventional usage of highly-variable genes for many integration tasks. We leverage gene-gene correlations that are stable across batches to identify biologically informative genes which are less affected by batch effects. Currently, its main use is for single-cell RNA-seq dataset integration, but it can be applied for other multi-batch downstream analyses such as NMF.

**License** GPL (>= 3)

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**LinkingTo** Rcpp, rmio, bigstatsr

**Depends** R (>= 4.5.0)

**Imports** bigparallelr, bigstatsr, checkmate, digest, dplyr, Matrix, pbapply, S4Vectors, SingleCellExperiment, stats, SummarizedExperiment, tidyr, withr

**Suggests** batchelor, BiocStyle, bluster, knitr, magick, matrixStats, patchwork, RcppArmadillo, rmarkdown, scater, scran, Seurat, splatter, testthat (>= 3.0.0), UpSetR

**VignetteBuilder** knitr

**biocViews** SingleCell, BatchEffect, MultipleComparison, FeatureExtraction

**BiocType** Software

**Config/testthat/edition** 3

**BugReports** https://github.com/BIMSBbioinfo/anglemania/issues

**Config/pak/sysreqs** libicu-dev zlib1g-dev

**Repository** Bioconductor 3.23

**Date/Publication** 2026-04-28 13:05:32 UTC

**RemoteUrl** https://github.com/bioc/anglemania

**RemoteRef** RELEASE_3_23

**RemoteSha** 95dff7c002922f892a8daba6fde40e3a0f323352

# Contents

anglemania . . . . . 2
extract_angles . . . . . 5
factorise . . . . . 7
permute_nonzero . . . . . 8
sce_example . . . . . 9

**Index** **10**

---

`anglemania` *anglemania*

---

## Description

`anglemania` computes critical angles between genes across all samples provided in an SingleCellExperiment object. It calculates angles, transforms them to z-scores, computes statistical measures, and selects the top genes based on mean and standard deviation of z-scores. These genes are biologically informative and invariant to batch effects.

This function adds a unique batch identifier to the metadata of a `SingleCellExperiment` object by combining specified dataset and batch keys. This is useful for distinguishing samples during integration or analysis.

## Usage

```
anglemania(
  sce,
  batch_key,
  dataset_key = NA_character_,
  max_n_genes = 2000,
  min_cells_per_gene = 1,
  min_samples_per_gene = 2,
  allow_missing_features = FALSE,
  method = "cosine",
  permute_row_or_column = "column",
  permutation_function = "sample",
  prefilter_threshold = 0.5,
  normalization_method = "divide_by_total_counts",
```

```
  verbose = TRUE
)

check_params(
  sce,
  batch_key,
  dataset_key,
  max_n_genes,
  method,
  min_cells_per_gene,
  min_samples_per_gene,
  allow_missing_features,
  permute_row_or_column,
  permutation_function,
  prefilter_threshold,
  normalization_method,
  verbose
)

add_unique_batch_key(sce, dataset_key = NA_character_, batch_key)

get_intersect_genes(
  matrix_list,
  allow_missing_features = FALSE,
  min_samples_per_gene = 1,
  verbose = TRUE
)
```

## Arguments

| | |
|---|---|
| sce | A SingleCellExperiment object. |
| batch_key | A character string specifying the column name in the metadata that identifies the batch. |
| dataset_key | A character string specifying the column name in the metadata that identifies the dataset. If NA, only the batch_key is used. |
| max_n_genes | Integer specifying the maximum number of genes to select. |
| min_cells_per_gene | Integer specifying the minimum number of cells per gene. Default is 1. |
| min_samples_per_gene | Integer indicating the minimum number of samples per gene. |
| allow_missing_features | Logical indicating whether to allow missing features. |
| method | Character string specifying the method to use for calculating the relationship between gene pairs. Default is "cosine". Other options include "spearman" |
| permute_row_or_column | Character "row" or "column", whether permutations should be executed row-wise or column wise. Default is "column" |

permutation_function
: Character "sample" or "permute_nonzero". If sample,then sample is used for constructing background distributions. If permute_nonzero, then only non-zero values are permuted. Default is "sample"

prefilter_threshold
: Numeric value specifying the threshold prefiltering genes. Speeds up gene selection.

normalization_method
: Character "divide_by_total_counts" or "scale_by_total_counts". Default is "divide_by_total_counts"

verbose
: Logical indicating whether to print messages.

matrix_list
: A list of bigstatsr::FBM objects.

## Details

This function performs the following steps:

1. Computes angles between genes for each batch in the SingleCellExperiment using the specified method, via factorise.
2. Transforms the angles to z-scores.
3. Computes statistical measures (mean z-score, signal-to-noise ratio) across batches using get_list_stats.
4. Selects the top n genes based on mean and standard deviation of z-scores using select_genes.

The computed statistics and selected genes are added to the SingleCellExperiment object, which is returned.

## Value

An updated SingleCellExperiment object with computed statistics and selected genes. The results are stored in the metadata of the SingleCellExperiment object.

A list of validated parameters

A SingleCellExperiment object with an additional metadata column containing the unique batch key.

A character vector of intersected genes.

## Functions

- check_params(): Check Parameters provided to the anglemania function
- add_unique_batch_key(): Temporarily add a unique batch key to the dataset
- get_intersect_genes(): Extract the intersected genes from a list of matrices (count matrices from different batches/datasets). It also allows for missing features in individual matrices, so that a feature does not have to be present in every single batch.

## See Also

get_list_stats, select_genes, factorise, big_apply, https://arxiv.org/abs/1306.0256

## Examples

```
# Set seed (optional)
set.seed(1)
sce <- sce_example()
sce <- anglemania(
  sce,
  batch_key = "batch",
  method = "cosine"
)

# Access the selected genes
selected_genes <- get_anglemania_genes(sce)
selected_genes[1:10]
sce <- sce_example()
params <- check_params(
  sce,
  batch_key = "batch",
  dataset_key = "dataset",
  max_n_genes = 2000,
  method = "cosine",
  min_cells_per_gene = 1,
  min_samples_per_gene = 2,
  allow_missing_features = FALSE,
  permute_row_or_column = "column",
  permutation_function = "sample",
  prefilter_threshold = 0.5,
  normalization_method = "divide_by_total_counts",
  verbose = TRUE
)
sce <- sce_example()
head(SummarizedExperiment::colData(sce))
sce <- add_unique_batch_key(
    sce,
    batch_key = "batch",
    dataset_key = "dataset"
)
head(SummarizedExperiment::colData(sce))
library(SingleCellExperiment)
sce <- sce_example()
barcodes_by_batch <- split(colnames(sce), colData(sce)$batch)
matrix_list <- lapply(barcodes_by_batch, function(barcodes) {
    SingleCellExperiment::counts(sce)[, barcodes]
})
intersect_genes <- get_intersect_genes(matrix_list)
head(intersect_genes)
```

---

`extract_angles` *Calculate cosine angle between genes*

---

## Description

Constructs a matrix of gene-gene relationships based on distance metrics.

## Usage

```
extract_angles(x_mat, method = "cosine")
```

## Arguments

| | |
|---|---|
| x_mat | An FBM object containing raw gene expression data, where rows correspond to genes and columns to samples. The data will be normalized and scaled within the function. |
| method | A character string specifying the method to compute the gene-gene relationships. Options are:<br>• "cosine" (default): Computes the cosine angle between genes.<br>• "spearman": Computes the Spearman rank correlation coefficient by rank-transforming the data before computing the correlation. |

## Details

The function returns the gene-gene angle matrix as an FBM object.

## Value

An FBM object containing the gene-gene correlation matrix. The matrix is square with dimensions equal to the number of genes and contains the pairwise correlations between genes. The diagonal elements are set to NA.

## See Also

big_apply, big_cor, FBM, factorise

## Examples

```
mat <- matrix(
 c(
     5, 3, 0, 0,
     0, 0, 0, 3,
     2, 1, 3, 4,
     0, 0, 1, 0,
     1, 2, 1, 2,
     3, 4, 3, 4
   ),
   nrow = 6, # 6 genes
   ncol = 4, # 4 cells
   byrow = TRUE
)

mat <- bigstatsr::FBM(nrow = nrow(mat), ncol = ncol(mat), init = mat)
```

```
angle_mat <- extract_angles(mat)
angle_mat[]
```

---

`factorise` *Factorize Angle Matrices into Z-Scores*

---

## Description

`factorise` computes the angle matrix of the input gene expression matrix using the specified method, performs permutation to create a null distribution, and transforms the correlations into z-scores. This function is optimized for large datasets using the **bigstatsr** package.

## Usage

```
factorise(
  x_mat,
  method = "cosine",
  seed = 1,
  permute_row_or_column = "column",
  permutation_function = "sample",
  normalization_method = "divide_by_total_counts"
)
```

## Arguments

| | |
|---|---|
| `x_mat` | A `FBM` object representing the normalized and scaled gene expression matrix. |
| `method` | A character string specifying the method for calculating the relationship between gene pairs. Default is `"cosine"`. Other options include `"spearman"` |
| `seed` | An integer value for setting the seed for reproducibility during permutation. Default is 1. |
| `permute_row_or_column` | Character "row" or "column", whether permutations should be executed row-wise or column wise. Default is `"column"` |
| `permutation_function` | Character "sample" or "permute_nonzero". If sample, then sample is used for constructing background distributions. If permute_nonzero, then only non-zero values are permuted. Default is `"sample"` |
| `normalization_method` | Character "divide_by_total_counts" or "scale_by_total_counts". Default is `"divide_by_total_counts"` |

## Details

The function performs the following steps:

1. **Permutation**: The input matrix is permuted column-wise to disrupt existing angles, creating a null distribution.

2. **Angle Computation**: Computes the angle matrix for both the original and permuted matrices using `extract_angles`.
3. **Method-Specific Processing**:
   - For other methods (`"cosine"`, `"spearman"`), statistical measures are computed from the permuted data.
4. **Statistical Measures**: Calculates mean, variance, and standard deviation using `get_dstat`.
5. **Z-Score Transformation**: Transforms the original angle matrix into z-scores.

This process allows for the identification of invariant gene-gene relationships by comparing them to a null distribution derived from the permuted data.

## Value

An `FBM` object containing the z-score-transformed angle matrix.

## See Also

`extract_angles`, `get_dstat`, `big_apply`, `FBM`

## Examples

```
mat <- matrix(
 c(
     5, 3, 0, 0,
     0, 0, 0, 3,
     2, 1, 3, 4,
     0, 0, 1, 0,
     1, 2, 1, 2,
     3, 4, 3, 4
   ),
   nrow = 6, # 6 genes
   ncol = 4, # 4 cells
   byrow = TRUE
)

mat <- bigstatsr::FBM(nrow = nrow(mat), ncol = ncol(mat), init = mat)

# Run factorise with method "cosine" and a fixed seed
result_fbm <- factorise(mat, method = "cosine", seed = 1)
result_fbm[]
```

---

`permute_nonzero` *permute non-zero elements of vectors*

---

## Description

Permutes the non-zero elements of a numeric vector.

## Usage

```
permute_nonzero(v)
```

## Arguments

| | |
|---|---|
| v | A numeric vector. |

## Value

A numeric vector with non-zero elements permuted.

## Examples

```
v <- c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10) # put in some zeroes
v = c(v, 0, 0, 0)
permute_nonzero(v)
```

---

# sce_example *Generate example SingleCellExperiment object*

---

## Description

This function generates a SingleCellExperiment object with 2 batches and 2 datasets. The object contains 300 genes and 600 cells. The counts matrix is generated using the `rpois` function with different lambda values for the two batches.

## Usage

```
sce_example(seed = 42)
```

## Arguments

| | |
|---|---|
| seed | The seed for the random number generator. Because this function is only used locally, we allow to set a seed within the function. |

## Value

A SingleCellExperiment object

## Examples

```
sce <- sce_example()
sce
```

# Index

add_unique_batch_key *(*anglemania*)*, 2
anglemania, 2

big_apply, *4*, *6*, *8*
big_cor, *6*

check_params *(*anglemania*)*, 2

extract_angles, 5, *8*

factorise, *4*, *6*, 7
FBM, *6–8*

get_dstat, *8*
get_intersect_genes *(*anglemania*)*, 2
get_list_stats, *4*

permute_nonzero, 8

sce_example, 9
select_genes, *4*
SingleCellExperiment, *2*