Package ‘SPACO’

April 1, 2026

Title Spatial Component Analysis for Spatial Sequencing Data

Version 1.0.1

Description Spatial components offer tools for dimension reduction and spatially variable gene detection for high dimensional spatial transcriptomics data. Construction of a projection onto low-dimensional feature space of spatially dependent metagenes offers pre-processing to clustering, testing for spatial variability and denoising of spatial expression patterns. For more details, see Koehler et al. (2026) <doi:10.1093/bioinformatics/btag052>.

Encoding UTF-8

RoxygenNote 7.3.3

LinkingTo Rcpp, RcppEigen

Depends R (>= 4.2.3)

Imports ggplot2, Seurat (>= 5.3.0), tibble, ggforce, methods, rARPACK, tidyr, mgcv, scales, Matrix (>= 1.5)

License MIT + file LICENSE

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation yes

Author David Köhler [aut, cre] (ORCID: <`https://orcid.org/0009-0006-0027-4046`>)

Maintainer David Köhler <`koehler@imbie.uni-bonn.de`>

Repository CRAN

Date/Publication 2026-03-31 22:40:38 UTC

Contents

`create_SpaCoObject_from_KNN` *create_SpaCoObject_from_KNN*

Description

create_SpaCoObject_from_KNN

Usage

```
create_SpaCoObject_from_KNN(SeuratObject, n = 10)
```

Arguments

`SeuratObject`	Seurat object to export kNN-graph from.
`n`	Number of neighbors to consider.

Value

Returns a SPaCoObject with the SCT data and the kNN-graph as neighborhood matrix.

`denoised_projection_plot` *Plot denoised gene expression*

Description

Plot denoised gene expression

Usage

```
denoised_projection_plot(SpaCoObject, feature = NULL)
```

Arguments

`SpaCoObject`	SpacoObject with computed projections
`feature`	Gene for which to plot denoised projection

Value

returns a ggplot object with the denoised gene expression.

`denoise_profiles` *computes smoothed gene profiles of genes present in the data.*

Description

computes smoothed gene profiles of genes present in the data.

Usage

```
denoise_profiles(SpaCoObject)
```

Arguments

`SpaCoObject`	Spaco object to compute profiles of.

Value

smoothed gene profiles in the SpaCoObject.

`eigenMapMatMult` *Multiply two matrices using Eigen library*

Description

This function multiplies two matrices using the Eigen library, which provides fast linear algebra operations.

Usage

```
eigenMapMatMult(A, B)
```

Arguments

`A`	a matrix
`B`	a matrix

Value

the product of A and B

`feature_plot` *Plot gene expression*

Description

Plot gene expression

Usage

```
feature_plot(SpaCoObject, feature)
```

Arguments

`SpaCoObject`	SpacoObject with computed projections
`feature`	Gene to plot

Value

returns a ggplot object with gene expression.

`read_10x_for_spaco` *Read in 10x Visium spatial transcriptomics data*

Description

Read in 10x Visium spatial transcriptomics data

Usage

```
read_10x_for_spaco(
  data_dir,
  slice,
  filename,
  variable_features_n = variable_features_n,
  spatial_file = spatial_file,
  vars_to_regress = NULL
)
```

Arguments

`data_dir`	Directory containing the H5 file specified by file name and the image data in a sub directory called spatial
`slice`	Name for the stored image of the tissue slice
`filename`	Filename of data to be read

`variable_features_n`	Number of most variable features to keep.
`spatial_file`	Name of csv file from which to read tissue positions.
`vars_to_regress`	Names of features to be regressed against using PercentageFeatureSet

Value

Retuns a ready to run SPaCoObject.

`set_projection` *Set projections of Spaco Object*

Description

Set projections of Spaco Object

Usage

```
set_projection(SpaCoObject, data, neighborhood = SpaCoObject@neighbours)
```

Arguments

`SpaCoObject`	Fitted object of class `SpaCoObject` as returned from [SpaCo] on which to update the projection slot with new data.
`data`	gene expression data matrix; p genes as columns, n loci as rows
`neighborhood`	neighborhood matrix of the new data. Must be of dimension n times n, where n is the number of columns of `data`

Value

Returns the SpaCoObject with the updated data, neighborhood matrix, and projection matrix.

`seurat_to_spaco` *Wrapper to transform existing Seurat object into an SpaCoObject.*

Description

Wrapper to transform existing Seurat object into an SpaCoObject.

Usage

```
seurat_to_spaco(SeuratObject, assay = "SCT", n_image = 1, layer = "scale.data")
```

Arguments

`SeuratObject`	Seurat object to export
`assay`	Assay to export from the Seurat object. Default is SCT assay.
`n_image`	Number of the image to export from Seurat object. Only relevant if Seurat object contains multiple images. Default is 1.
`layer`	Which layer to export data from. Default is scale.data.

Value

Returns a SpaCoObject with all slots filled necessary to perform the spatial component analysis.

`SpaCo` *SpaCo*

Description

SpaCo

Usage

```
SpaCo(
  SpaCoObject,
  PC_criterion = "percent",
  PC_value = 0.95,
  set_nspacs = NULL,
  nSim = 1000,
  nSpacQuantile = 0.05,
  reducedSpots = FALSE,
  nReduce = 1000
)
```

Arguments

`SpaCoObject`	Object of class `SpaCoObject` as generated from [SpaCoObject] on which to perform SCA.
`PC_criterion`	criterion on which to select number of principal components for initial covariance matrix reconstruction; either "number" to select a number of PCs or "percent" to select number of PCs to explain specified amount of data variance
`PC_value`	Value to specify number of PCs or desired level of explained variance, see "PC_criterion"
`set_nspacs`	Boolean if number of relevant spacs is to be computed. Increases run time significantly
`nSim`	Number of simulations for computation of spac number
`nSpacQuantile`	Quantile to use as cutoff for spac number
`reducedSpots`	Should algorithm be run on a subset of spots? Default = FALSE
`nReduce`	Number of spots to sample to run algorithm on subset if `reducedSpots==TRUE`. Ignored if `reducedSpots==FALSE`

Value

Returns a SpaCoObject filled with the result of the spatial component analysis.

`SpaCoObject` *Create a constructor function that creates an object of class SpaCoObject*

Description

Create a constructor function that creates an object of class SpaCoObject

Usage

```
SpaCoObject(neighbours, data, coordinates)
```

Arguments

`neighbours`	Binary matrix with weights describing if cells are to be considered neighbours or not depending on the defined distance.
`data`	Matrix with normalized and scaled gene counts. Rows as cells and genes as columns
`coordinates`	Matri with the cell coordinates on the slides. Rows and Columns in the 10x Visium case.

Value

Returns a SpaCoObject with the given slots filled

`Spaco_plot` *Plot SPaCo meta genes.*

Description

Plot SPaCo meta genes.

Usage

```
Spaco_plot(SpaCoObject, spac = 1)
```

Arguments

`SpaCoObject`	SpacoObject with computed projections
`spac`	component to plot

Value

returns a ggplot object with the meta gene expression.

`spacs_to_seurat` *transfer computed spatial components to existing Seurat object.*

Description

transfer computed spatial components to existing Seurat object.

Usage

```
spacs_to_seurat(SpaCoObject, SeuratObject, nSpacs = SpaCoObject@nSpacs)
```

Arguments

`SpaCoObject`	SpaCoObject to export spatial components from.
`SeuratObject`	Seurat object to add spatial components to.
`nSpacs`	Number of Spacs which are to be projected on for dimension reduction

Value

Returns a Seurat Object with the spatial components projections in the dimensional reduction slot.

`subset_non_neighbour_cells` *Filtering function to remove cells without neighbours in defined distance from existing Seurat object to be conformable with existing SpaCoObject.*

Description

Filtering function to remove cells without neighbours in defined distance from existing Seurat object to be conformable with existing SpaCoObject.

Usage

```
subset_non_neighbour_cells(SpaCoObject, SeuratObject)
```

Arguments

`SpaCoObject`	SpaCoObject to integrate into Seurat object.
`SeuratObject`	Seurat object to be filtered.

Value

Returns a Seurat object with cells filtered to match SpaCoObject.

`SVGTest` *Compute the spatial variable genes of a SpaCoObject after runinng runSCA*

Description

Compute the spatial variable genes of a SpaCoObject after runinng runSCA

Usage

```
SVGTest(SpaCoObject, adjustMethod = "holm")
```

Arguments

`SpaCoObject`	SpaCoObject to compute spatially variable genes of.
`adjustMethod`	method for p-value adjustment. See p.adjust function.

Value

returns a data frame of spatial variable genes and their p-Values.

Index