Package ‘CARDspa’

April 27, 2025

Title Spatially Informed Cell Type Deconvolution for Spatial Transcriptomics

Version 1.0.0

Date 2025-2-6

Description CARD is a reference-based deconvolution method that estimates cell type composition in spatial transcriptomics based on cell type specific expression information obtained from a reference scRNA-seq data. A key feature of CARD is its ability to accommodate spatial correlation in the cell type composition across tissue locations, enabling accurate and spatially informed cell type deconvolution as well as refined spatial map construction. CARD relies on an efficient optimization algorithm for constrained maximum likelihood estimation and is scalable to spatial transcriptomics with tens of thousands of spatial locations and tens of thousands of genes.

License GPL-3 + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Depends R (>= 4.3.0)

Imports Rcpp (>= 1.0.7),RcppArmadillo, SummarizedExperiment, methods, MCMCpack, fields, wrMisc, concaveman, sp, dplyr, sf, Matrix, RANN, ggplot2, reshape2, RColorBrewer, S4Vectors, scatterpie, grDevices,ggcorrplot, stats, nnls, BiocParallel, RcppML, NMF, spatstat.random, gtools, SingleCellExperiment, SpatialExperiment

LazyData false

biocViews Spatial, SingleCell, Transcriptomics, Visualization

LinkingTo Rcpp, RcppArmadillo

Suggests knitr, rmarkdown, testthat, BiocStyle

VignetteBuilder knitr

URL `https://github.com/YMa-lab/CARDspa`

BugReports https://github.com/YMa-lab/CARDspa/issues

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

git_branch RELEASE_3_21

git_last_commit 1aa63af

git_last_commit_date 2025-04-15

Repository Bioconductor 3.21

Date/Publication 2025-04-27

Author Ying Ma [aut],
Jing Fu [cre]

Maintainer Jing Fu <`jing_fu@brown.edu`>

Contents

`assign_sc_cords` *The function to assign the spatial location information for each single cell*

Description

The function to assign the spatial location information for each single cell

Usage

```
assign_sc_cords(mappint_spot_cell_cor, cords_new, numcell, sc_eset, ct_varname)
```

Arguments

`mappint_spot_cell_cor`	a mapped correlation matrix indicating the relashionship between each measured spatial location and the single cell in the scRNAseq reference
`cords_new`	output from the function get_high_res_cords
`numcell`	a numeric value indicating the number of single cells in each measured location, we suggest 20 for ST technology, 7 for 10x Viisum and 2 for Slide-seq
`sc_eset`	a single cell experiment object stored in CARD object
`ct_varname`	character, the name of the column in metaData that specifies the cell type annotation information, stroed in CARD object

Value

Return the assigned spatial location information for the mapped single cell

---

`CARD-class` *Each CARD object has a number of slots which store information. Key slots to access are listed below.*

Description

Each CARD object has a number of slots which store information. Key slots to access are listed below.

Value

Return an object of CARD class

Slots

- `sc_eset` The filtered scRNA-seq data along with meta data stored in the format of SingleCellExperiment.
- `spatial_countMat` The filtered spatial count data.
- `spatial_location` The weights for combining p-values from multiple kernels.
- `Proportion_CARD` The estimated cell type proportion by CARD with each row is a spatial location and each column is a cell type.
- `project` The name of the project, default is deconvolution.
- `info_parameters` The paramters that are used in model fitting.
- `algorithm_matrix` The intermediate matrices that are used in the model fitting step.
- `refined_prop` The refined cell type proportion matrix estimated by CARD for the newly grided spatial locations. The number of initial grids are defined by the user.
- `refined_expression` The refined predicted expression matrix (normalized) estimated by CARD for the newly grided spatial locations. The number of initial grids are defined by the user.

`CARDfree` *SpatialDeconv function based on Conditional Autoregressive model*

Description

SpatialDeconv function based on Conditional Autoregressive model

Usage

```
CARDfree(
  XinputIn,
  UIn,
  WIn,
  phiIn,
  max_iterIn,
  epsilonIn,
  initV,
  initb,
  initSigma_e2,
  initLambda
)
```

Arguments

`XinputIn`	The input of normalized spatial data
`UIn`	The input of cell type specific basis matrix B
`WIn`	The constructed W weight matrix from Gaussian kernel
`phiIn`	The phi value

`max_iterIn`	Maximum iterations
`epsilonIn`	epsilon for convergence
`initV`	Initial matrix of cell type compositions V
`initb`	Initial vector of cell type specific intercept
`initSigma_e2`	Initial value of residual variance
`initLambda`	Initial vector of cell type sepcific scalar.

Value

A list

`CARDfree-class` *Each CARDfree object has a number of slots which store information. Key slots to access are listed below.*

Description

Each CARDfree object has a number of slots which store information. Key slots to access are listed below.

Value

Return an object of CARDfree class

Slots

`spatial_countMat` The filtered spatial count data.

`spatial_location` The weights for combining p-values from multiple kernels.

`Proportion_CARD` The estimated cell type proportion by CARD with each row is a spatial location and each column is a cell type.

`estimated_refMatrix` The estimated reference matrix by CARDfree with each row represents a gene and each column represents a cell type cluster.

`project` The name of the project, default is deconvolution.

`markerList` The nlist of cell type specific markers, with each element represents the vector of cell type specific markers

`info_parameters` The paramters that are used in model fitting.

`algorithm_matrix` The intermediate matrices that are used in the model fitting step.

`refined_prop` The refined cell type proportion matrix estimated by CARD for the newly grided spatial locations. The number of initial grids are defined by the user.

`refined_expression` The refined predicted expression matrix (normalized) estimated by CARD for the newly grided spatial locations. The number of initial grids are defined by the user.

`CARDref` *SpatialDeconv function based on Conditional Autoregressive model*

Description

SpatialDeconv function based on Conditional Autoregressive model

Usage

```
CARDref(
  XinputIn,
  UIn,
  WIn,
  phiIn,
  max_iterIn,
  epsilonIn,
  initV,
  initb,
  initSigma_e2,
  initLambda
)
```

Arguments

`XinputIn`	The input of normalized spatial data
`UIn`	The input of cell type specific basis matrix B
`WIn`	The constructed W weight matrix from Gaussian kernel
`phiIn`	The phi value
`max_iterIn`	Maximum iterations
`epsilonIn`	epsilon for convergence
`initV`	Initial matrix of cell type compositions V
`initb`	Initial vector of cell type specific intercept
`initSigma_e2`	Initial value of residual variance
`initLambda`	Initial vector of cell type sepcific scalar.

Value

A list

`CARD_deconvolution` *Spatially Informed Cell Type Deconvolution for Spatial Transcriptomics by CARD*

Description

Spatially Informed Cell Type Deconvolution for Spatial Transcriptomics by CARD

Usage

```
CARD_deconvolution(
  sc_count,
  sc_meta,
  spatial_count,
  spatial_location,
  ct_varname,
  ct_select,
  sample_varname,
  mincountgene = 100,
  mincountspot = 5,
  sce = NULL,
  spe = NULL
)
```

Arguments

`sc_count`	Raw scRNA-seq count data, each column is a cell and each row is a gene.
`sc_meta`	data frame, with each row representing the cell type and/or sample information of a specific cell. The row names of this data frame should match exactly with the column names of the sc_count data
`spatial_count`	Raw spatial resolved transcriptomics data, each column is a spatial location, and each row is a gene.
`spatial_location`	data frame, with two columns representing the x and y coordinates of the spatial location. The rownames of this data frame should match eaxctly with the columns of the spatial_count.
`ct_varname`	character, the name of the column in metaData that specifies the cell type annotation information
`ct_select`	vector of cell type names that you are interested in to deconvolute, default as NULL. If NULL, then use all cell types provided by single cell dataset;
`sample_varname`	character,the name of the column in metaData that specifies the sample information. If NULL, we just use the whole as one sample.
`mincountgene`	Minimum counts for each gene
`mincountspot`	Minimum counts for each spatial location

`sce`	a `SingleCellExperiment` object containing scRNA-seq count data in the `counts` assay, and cell types and sample information in the colData.
`spe`	a `SpatialExperiment` object containing spatial data in the `counts` assay, and spatial coordinates in the spatialCoords.

Value

Returns a SpatialExperiment object with estimated cell type proportion stored in object$Proportion_CARD.

Examples

```
library(RcppML)
library(NMF)
library(RcppArmadillo)
data(spatial_count)
data(spatial_location)
data(sc_count)
data(sc_meta)
CARD_obj <- CARD_deconvolution(
    sc_count = sc_count,
    sc_meta = sc_meta,
    spatial_count = spatial_count,
    spatial_location = spatial_location,
    ct_varname = "cellType",
    ct_select = unique(sc_meta$cellType),
    sample_varname = "sampleInfo",
    mincountgene = 100,
    mincountspot = 5
)
```

`CARD_imputation` *Construct an enhanced spatial expression map on the unmeasured tissue locations*

Description

Construct an enhanced spatial expression map on the unmeasured tissue locations

Usage

```
CARD_imputation(CARD_object, num_grids, ineibor = 10, exclude = NULL)
```

Arguments

`CARD_object`	SpatialExperiment Object created by CARD_deconvolution with estimated cell type compositions on the original spatial resolved transcriptomics data.
`num_grids`	Initial number of newly grided spatial locations. The final number of newly grided spatial locations will be lower than this value since the newly grided locations outside the shape of the tissue will be filtered

`ineibor`	Numeric, number of neighbors used in the imputation on newly grided spatial locations, default is 10.
`exclude`	Vector, the rownames of spatial location data on the original resolution that you want to exclude. This is to avoid the weird detection of the shape.

Value

Return a SpatialExperiment object with the refined cell type compositions estimated for newly grided spots and the refined predicted gene expression (normalized).

Examples

```
data(spatial_count)
data(spatial_location)
data(sc_count)
data(sc_meta)
CARD_obj <- CARD_deconvolution(
    sc_count = sc_count,
    sc_meta = sc_meta,
    spatial_count = spatial_count,
    spatial_location = spatial_location,
    ct_varname = "cellType",
    ct_select = unique(sc_meta$cellType),
    sample_varname = "sampleInfo",
    mincountgene = 100,
    mincountspot = 5
)
CARD_obj <- CARD_imputation(
    CARD_obj,
    num_grids = 200,
    ineibor = 10,
    exclude = NULL
)
```

`CARD_refFree` *Extension of CARD into a reference-free version of deconvolution: CARDfree.*

Description

Extension of CARD into a reference-free version of deconvolution: CARDfree.

Usage

```
CARD_refFree(
  markerlist,
  spatial_count,
  spatial_location,
```

```
  mincountgene = 100,
  mincountspot = 5,
  spe = NULL
)
```

Arguments

`markerlist`	a list of marker genes, with each element of the list being the vector of cell type specific marker genes
`spatial_count`	Raw spatial resolved transcriptomics data, each column is a spatial location, and each row is a gene.
`spatial_location`	data frame, with two columns representing the x and y coordinates of the spatial location. The rownames of this data frame should match eaxctly with the columns of the spatial_count.
`mincountgene`	Minimum counts for each gene
`mincountspot`	Minimum counts for each spatial location
`spe`	a `SpatialExperiment` object containing spatial data in the `counts` assay, and spatial coordinates in the spatialCoords.

Value

Returns a SpatialExperiment object with estimated cell type proportion stored in object$Proportion_CARD. Because this is a reference-free version, the columns of estimated proportion is not cell type but cell type cluster

Examples

```
library(RcppML)
library(NMF)
library(RcppArmadillo)
data(markerList)
data(spatial_count)
data(spatial_location)
CARDfree_obj <- CARD_refFree(
markerlist = markerList[8:16],
spatial_count = spatial_count[1:2500, ],
spatial_location = spatial_location,
mincountgene = 100,
mincountspot = 5
)
```

`CARD_scmapping` *Extension of CARD into performing single cell Mapping from non-single cell spatial transcriptomics dataset.*

Description

Extension of CARD into performing single cell Mapping from non-single cell spatial transcriptomics dataset.

Usage

```
CARD_scmapping(CARD_object, shapeSpot = "Square", numcell, ncore = 10)
```

Arguments

`CARD_object`	CARD object create by the CARD_deconvolution function.
`shapeSpot`	a character indicating whether the sampled spatial coordinates for single cells locating in a Square-like region or a Circle-like region. The center of this region is the measured spatial location in the non-single cell resolution spatial transcriptomics data. The default is 'Square', the other shape is 'Circle'
`numcell`	a numeric value indicating the number of single cells in each measured location, we suggest 20 for ST technology, 7 for 10x Viisum and 2 for Slide-seq
`ncore`	a numeric value indicating the number of cores used to accelerating the procedure

Value

Returns a SingleCellExperiment SCE object with the mapped expression at single cell resolution and the spatial location information of each single cell

Examples

```
library(SingleCellExperiment)
data(spatial_count)
data(spatial_location)
data(sc_count)
data(sc_meta)
CARD_obj <- CARD_deconvolution(
    sc_count = sc_count,
    sc_meta = sc_meta,
    spatial_count = spatial_count,
    spatial_location = spatial_location,
    ct_varname = "cellType",
    ct_select = unique(sc_meta$cellType),
    sample_varname = "sampleInfo",
    mincountgene = 100,
    mincountspot = 5
)
```

```
scMapping <- CARD_scmapping(
CARD_obj,
shapeSpot = "Square",
numcell = 20,
ncore = 2)
print(scMapping)
```

`CARD_visualize_Cor` *Visualize the cell type proportion correlation*

Description

Visualize the cell type proportion correlation

Usage

```
CARD_visualize_Cor(proportion, colors = colors)
```

Arguments

`proportion`	Data frame, cell type proportion estimated by CARD in either original resolution or enhanced resolution.
`colors`	Vector of color names that you want to use, if NULL, we will use the default color scale c("#91a28c","white","#8f2c37")

Value

Returns a ggcorrplot figure.

Examples

```
library(ggplot2)
data(spatial_count)
data(spatial_location)
data(sc_count)
data(sc_meta)
CARD_obj <- CARD_deconvolution(
    sc_count = sc_count,
    sc_meta = sc_meta,
    spatial_count = spatial_count,
    spatial_location = spatial_location,
    ct_varname = "cellType",
    ct_select = unique(sc_meta$cellType),
    sample_varname = "sampleInfo",
    mincountgene = 100,
    mincountspot = 5
)
CARD_visualize_Cor(CARD_obj$Proportion_CARD, colors = NULL)
```

`CARD_visualize_gene` *Visualize the spatial distribution of cell type proportion*

Description

Visualize the spatial distribution of cell type proportion

Usage

```
CARD_visualize_gene(
  spatial_expression,
  spatial_location,
  gene_visualize,
  colors = colors,
  NumCols
)
```

Arguments

`spatial_expression`	Data frame, spatial gene expression in either original resolution or enhanced resolution.
`spatial_location`	Data frame, spatial location information.
`gene_visualize`	Vector of selected gene names that are interested to visualize
`colors`	Vector of color names that you want to use, if NULL, we will use the default color scale in virdis palette
`NumCols`	Numeric, number of columns in the figure panel, it depends on the number of cell types you want to visualize.

Value

Returns a ggplot2 figure.

Examples

```
library(ggplot2)
library(SummarizedExperiment)
library(SpatialExperiment)
data(spatial_count)
data(spatial_location)
data(sc_count)
data(sc_meta)
CARD_obj <- CARD_deconvolution(
    sc_count = sc_count,
    sc_meta = sc_meta,
    spatial_count = spatial_count,
```

```
    spatial_location = spatial_location,
    ct_varname = "cellType",
    ct_select = unique(sc_meta$cellType),
    sample_varname = "sampleInfo",
    mincountgene = 100,
    mincountspot = 5
)
CARD_visualize_gene(
    spatial_expression = assays(CARD_obj)$spatial_countMat,
    spatial_location = spatialCoords(CARD_obj),
    gene_visualize = c("A4GNT", "AAMDC", "CD248"),
    colors = NULL,
    NumCols = 3
)
```

`CARD_visualize_pie` *Visualize the spatial distribution of cell type proportion in a geom scatterpie plot*

Description

Visualize the spatial distribution of cell type proportion in a geom scatterpie plot

Usage

```
CARD_visualize_pie(proportion, spatial_location, colors = NULL, radius = NULL)
```

Arguments

`proportion`	Data frame, cell type proportion estimated by CARD in either original resolution or enhanced resolution.
`spatial_location`	Data frame, spatial location information.
`colors`	Vector of color names that you want to use, if NULL, we will use the color palette "Spectral" from RColorBrewer package.
`radius`	Numeric value about the radius of each pie chart, if NULL, we will calculate it inside the function.

Value

Returns a ggplot2 figure.

Examples

```
library(ggplot2)
library(SpatialExperiment)
data(spatial_count)
data(spatial_location)
data(sc_count)
data(sc_meta)
CARD_obj <- CARD_deconvolution(
    sc_count = sc_count,
    sc_meta = sc_meta,
    spatial_count = spatial_count,
    spatial_location = spatial_location,
    ct_varname = "cellType",
    ct_select = unique(sc_meta$cellType),
    sample_varname = "sampleInfo",
    mincountgene = 100,
    mincountspot = 5
)
colors <- c(
    "#FFD92F", "#4DAF4A", "#FCCDE5", "#D9D9D9", "#377EB8", "#7FC97F",
    "#BEAED4", "#FDC086", "#FFFF99", "#386CB0", "#F0027F", "#BF5B17",
    "#666666", "#1B9E77", "#D95F02", "#7570B3", "#E7298A", "#66A61E",
    "#E6AB02", "#A6761D"
)
CARD_visualize_pie(
    proportion = CARD_obj$Proportion_CARD,
    spatial_location = spatialCoords(CARD_obj),
    colors = colors,
    radius = 0.52
)
```

`CARD_visualize_prop` *Visualize the spatial distribution of cell type proportion*

Description

Visualize the spatial distribution of cell type proportion

Usage

```
CARD_visualize_prop(
  proportion,
  spatial_location,
  ct_visualize = ct_visualize,
  colors = c("lightblue", "lightyellow", "red"),
  NumCols,
  pointSize = 3
)
```

Arguments

`proportion`	Data frame, cell type proportion estimated by CARD in either original resolution or enhanced resolution.
`spatial_location`	Data frame, spatial location information.
`ct_visualize`	Vector of selected cell type names that are interested to visualize
`colors`	Vector of color names that you want to use, if NULL, we will use the default color scale c("lightblue","lightyellow","red")
`NumCols`	Numeric, number of columns in the figure panel, it depends on the number of cell types you want to visualize.
`pointSize`	Size of each point used for plotting

Value

Returns a ggplot2 figure.

Examples

```
library(ggplot2)
library(SpatialExperiment)
data(spatial_count)
data(spatial_location)
data(sc_count)
data(sc_meta)
CARD_obj <- CARD_deconvolution(
    sc_count = sc_count,
    sc_meta = sc_meta,
    spatial_count = spatial_count,
    spatial_location = spatial_location,
    ct_varname = "cellType",
    ct_select = unique(sc_meta$cellType),
    sample_varname = "sampleInfo",
    mincountgene = 100,
    mincountspot = 5
)
ct_visualize <- c(
    "Acinar_cells", "Cancer_clone_A", "Cancer_clone_B",
    "Ductal_terminal_ductal_like", "Ductal_CRISP3_high-centroacinar_like",
    "Ductal_MHC_Class_II", "Ductal_APOL1_high-hypoxic", "Fibroblasts"
)
CARD_visualize_prop(
    proportion = CARD_obj$Proportion_CARD,
    spatial_location = spatialCoords(CARD_obj),
    ct_visualize = ct_visualize,
    colors = c("lightblue", "lightyellow", "red"),
    NumCols = 4,
    pointSize = 3.0
)
```

`CARD_visualize_prop_2CT` *Visualize the spatial distribution of two cell type proportions on the same plot*

Description

Visualize the spatial distribution of two cell type proportions on the same plot

Usage

```
CARD_visualize_prop_2CT(
  proportion,
  spatial_location,
  ct2_visualize = ct2_visualize,
  colors = NULL
)
```

Arguments

`proportion`	Data frame, cell type proportion estimated by CARD in either original resolution or enhanced resolution.
`spatial_location`	Data frame, spatial location information.
`ct2_visualize`	Vector of selected two cell type names that are interested to visualize, here we only focus on two cell types
`colors`	list of color names that you want to use for each cell type, if NULL, we will use the default color scale list list(c("lightblue","lightyellow","red"),c("lightblue","lightyellow","black"))

Value

Returns a ggplot2 figure.

Examples

```
library(ggplot2)
library(SpatialExperiment)
data(spatial_count)
data(spatial_location)
data(sc_count)
data(sc_meta)
CARD_obj <- CARD_deconvolution(
    sc_count = sc_count,
    sc_meta = sc_meta,
    spatial_count = spatial_count,
    spatial_location = spatial_location,
    ct_varname = "cellType",
    ct_select = unique(sc_meta$cellType),
```

```
    sample_varname = "sampleInfo",
    mincountgene = 100,
    mincountspot = 5
)
CARD_visualize_prop_2CT(
    proportion = CARD_obj$Proportion_CARD,
    spatial_location = spatialCoords(CARD_obj),
    ct2_visualize = c("Cancer_clone_A", "Cancer_clone_B"),
    colors = list(c("lightblue", "lightyellow", "red"), c(
        "lightblue", "lightyellow",
        "black"
    ))
)
```

`createCARDfreeObject` *Create the CARD object*

Description

Create the CARD object

Usage

```
createCARDfreeObject(
  markerlist,
  spatial_count,
  spatial_location,
  mincountgene = 100,
  mincountspot = 5,
  spe = NULL
)
```

Arguments

`markerlist`	a list of marker genes, with each element of the list being the vector of cell type specific marker genes
`spatial_count`	Raw spatial resolved transcriptomics data, each column is a spatial location, and each row is a gene.
`spatial_location`	data frame, with two columns representing the x and y coordinates of the spatial location. The rownames of this data frame should match eaxctly with the columns of the spatial_count.
`mincountgene`	Minimum counts for each gene
`mincountspot`	Minimum counts for each spatial location
`spe`	a `SpatialExperiment` object containing spatial data in the `counts` assay, and spatial coordinates in the spatialCoords.

Value

Returns CARDfree object with filtered spatial count and marker gene list.

`createCARDObject` *Create the CARD object*

Description

Create the CARD object

Usage

```
createCARDObject(
  sc_count,
  sc_meta,
  spatial_count,
  spatial_location,
  ct_varname,
  ct_select,
  sample_varname,
  mincountgene = 100,
  mincountspot = 5,
  sce = NULL,
  spe = NULL
)
```

Arguments

`sc_count`	Raw scRNA-seq count data, each column is a cell and each row is a gene.
`sc_meta`	data frame, with each row representing the cell type and/or sample information of a specific cell. The row names of this data frame should match exactly with the column names of the sc_count data
`spatial_count`	Raw spatial resolved transcriptomics data, each column is a spatial location, and each row is a gene.
`spatial_location`	data frame, with two columns representing the x and y coordinates of the spatial location. The rownames of this data frame should match eaxctly with the columns of the spatial_count.
`ct_varname`	character, the name of the column in metadata that specifies the cell type annotation information
`ct_select`	vector of cell type names that you are interested in to deconvolute, default as NULL. If NULL, then use all cell types provided by single cell dataset;
`sample_varname`	character,the name of the column in metadata that specifies the sample information. If NULL, we just use the whole as one sample.
`mincountgene`	Minimum counts for each gene

`mincountspot`	Minimum counts for each spatial location
`sce`	a `SingleCellExperiment` object containing scRNA-seq count data in the `counts` assay, and cell types and sample information in the colData.
`spe`	a `SpatialExperiment` object containing spatial data in the `counts` assay, and spatial coordinates in the spatialCoords.

Value

Returns CARD object with filtered spatial count and single cell RNA-seq dataset.

`create_ref` *Construct the mean gene expression basis matrix (B), this is the faster version*

Description

Construct the mean gene expression basis matrix (B), this is the faster version

Usage

```
create_ref(sc_eset, ct_select = NULL, ct_varname, sample_varname = NULL)
```

Arguments

`sc_eset`	S4 class for storing data from single-cell experiments. This format is usually created by the package SingleCellExperiment with stored counts, along with the usual metadata for genes and cells.
`ct_select`	vector of cell type names that you are interested in to deconvolute, default as NULL. If NULL, then use all cell types provided by single cell dataset;
`ct_varname`	character, the name of the column in metaData that specifies the cell type annotation information
`sample_varname`	character,the name of the column in metaData that specifies the sample information. If NULL, we just use the whole as one sample.

Value

Return a list of basis (B) matrix

`get_high_res_cords` *The function to sample the spatial location information for each single cell*

Description

The function to sample the spatial location information for each single cell

Usage

```
get_high_res_cords(cords, numcell, shape = "Square")
```

Arguments

`cords`	The spatial location information in the measure spatial locations, with the first and second columns represent the 2-D x-y coordinate system
`numcell`	a numeric value indicating the number of single cells in each measured location, we suggest 20 for ST technology, 7 for 10x Viisum and 2 for Slide-seq
`shape`	a character indicating whether the sampled spatial coordinates for single cells locating in a Square-like region or a Circle-like region. The center of this region is the measured spatial location in the non-single cell resolution spatial transcriptomics data. The default is 'Square', the other shape is 'Circle'

Value

Returns a dataframe with the sampled spatial location information for each single cell

---

`get_weight_for_cell` *The function to estimate the cell type composition signature for each single cell in the scRNaseq reference data*

Description

The function to estimate the cell type composition signature for each single cell in the scRNaseq reference data

Usage

```
get_weight_for_cell(sc_eset, ct_varname, ct_select, sample_varname, B)
```

Arguments

`sc_eset`	the sc_eset stored in the CARD object
`ct_varname`	character, the name of the column in metaData that specifies the cell type annotation information, stored in the CARD object
`ct_select`	vector of cell type names that you are interested in to deconvolute, default as NULL. stored in the CARD object
`sample_varname`	character,the name of the column in metaData that specifies the sample information. stored in the CARD object
`B`	reference basis matrix stored in the CARD object.

Value

Returns a matrix of the cell type composition signature for each single cell in the scRNaseq reference

---

`markerList` *marker gene list*

Description

The marker gene list is a list format with each element of the list being the cell type specific gene markers.

Usage

```
data(markerList)
```

Format

An object of class `list` of length 20.

`mvn_cv` *Imputation and Construction of High-Resolution Spatial Maps for Cell Type Composition and Gene Expression by the spatial correlation structure between original spatial locations and new grided spatial locations*

Description

Imputation and Construction of High-Resolution Spatial Maps for Cell Type Composition and Gene Expression by the spatial correlation structure between original spatial locations and new grided spatial locations

Usage

```
mvn_cv(
  vtrain,
  location_orig,
  train_ind,
  test_ind,
  B,
  xinput_norm,
  optimal_b,
  optimal_phi,
  lambda,
  ineibor
)
```

Arguments

`vtrain`	Matrix, estimated V matrix from CARD
`location_orig`	Data frame, spatial location data frame of the original spatial resolved transcriptomics dataset, stored in the spatialCoords(CARD_object)
`train_ind`	Vector, index of the original spatial locations
`test_ind`	Vector, index of the newly grided spatial locations
`B`	Matrix, used in the deconvolution as the reference basis matrix
`xinput_norm`	Matrix, used in the deconvolution as the normalized spatial count data
`optimal_b`	Vector, vector of the intercept for each cel type estimated based on the original spatial resolution
`optimal_phi`	Numeric, the optimal phi value stored in CARD_object
`lambda`	Vector, vector of cell type specific scalar in the CAR model
`ineibor`	Numeric, number of neighbors used in the imputation on newly grided spatial locations, default is 10.

Value

Return a list with the imputed Cell type composition Vtest matrix on the newly grided spatial locations and predicted normalized gene expression

---

`norm_coords_train_test` *Normalize the new spatial locations without changing the shape and relative positions*

Description

Normalize the new spatial locations without changing the shape and relative positions

Usage

```
norm_coords_train_test(location_orig, train_ind, test_ind)
```

Arguments

`location_orig`	Data frame, spatial location data frame of the original spatial resolved transcriptomics dataset, stored in the spatialCoords(CARD_object)
`train_ind`	Vector, Index of the original spatial locations
`test_ind`	Vector, Index of the newly grided spatial locations

Value

Return the normalized spatial location data frame

`sample_grid_within` *Make new spatial locations on unmeasured tissue through grids.*

Description

Make new spatial locations on unmeasured tissue through grids.

Usage

```
sample_grid_within(location, num_sample, concavity = 2)
```

Arguments

`location`	Data frame, spatial location data frame of the original spatial resolved transcriptomics dataset, stored in the spatialCoords(CARD_object)
`num_sample`	Numeric, approximate number of cells in grid within the shape of the spatial location data frame
`concavity`	Numeric, a relative measure of concavity. The default is 2.0, which can prodecure detailed enough shapes. Infinity results in a convex hull while 1 results in a more detailed shape.

Value

Return a list of data frame with newly grided points

`sc_count` *scRNA-seq count data*

Description

The scRNA-seq count data must be in the format of matrix or sparseMatrix, while each row represents a gene and each column represents a cell.

Usage

```
data(sc_count)
```

Format

An object of class `dgCMatrix` with 7000 rows and 1926 columns.

`sc_meta` *scRNAseq meta data*

Description

The scRNAseq meta data must be in the format of data frame while each row represents a cell. The rownames of the scRNAseq meta data should match exactly with the column names of the scRNAseq count data. The sc_meta data must contain the column indicating the cell type assignment for each cell (e.g., “cellType” column in the example sc_meta data). Sample/subject information should be provided, if there is only one sample, we can add a column by sc_meta$sampleInfo = "sample1".

Usage

```
data(sc_meta)
```

Format

An object of class `data.frame` with 1926 rows and 3 columns.

`sc_QC` *Quality control of scRNA-seq count data*

Description

Quality control of scRNA-seq count data

Usage

```
sc_QC(
  counts_in,
  metadata,
  ct_varname,
  ct_select,
  sample_varname = NULL,
  min_cells = 0,
  min_genes = 0
)
```

Arguments

`counts_in`	Raw scRNAseq count data, each column is a cell and each row is a gene.
`metadata`	data frame, metadata with "ct_varname" specify the cell type annotation information and "sample_varname" specify the sample information
`ct_varname`	character, the name of the column in metadata that specifies the cell type annotation information
`ct_select`	vector of cell type names that you are interested in to deconvolute, default as NULL. If NULL, then use all cell types provided by single cell dataset;
`sample_varname`	character,the name of the column in metadata that specifies the sample information. If NULL, we just use the whole as one sample.
`min_cells`	numeric, we filtered out the non-expressed cells.
`min_genes`	numeric we filtered out the non-expressed genes

Value

Return the filtered scRNA-seq data and meta data stored in a S4 class (SingleCellExperiment)

`select_info` *Select Informative Genes used in the deconvolution*

Description

Select Informative Genes used in the deconvolution

Usage

```
select_info(basis, sc_eset, commongene, ct_select, ct_varname)
```

Arguments

`basis`	Reference basis matrix.
`sc_eset`	scRNAseq data along with meta data stored in the S4 class format (SingleCell-Experiment).
`commongene`	common genes between scRNAseq count data and spatial resolved transcriptomics data.
`ct_select`	vector of cell type names that you are interested in to deconvolute, default as NULL. If NULL, then use all cell types provided by single cell dataset;
`ct_varname`	character, the name of the column in metaData that specifies the cell type annotation information

Value

a vector of informative genes selected

`show,CARD-method` *Show method for the CARD class*

Description

This method provides a concise summary of an object of class `CARD`, displaying key information including the project name, the number of spots, the number of cell types, and a sample of the `Proportion_CARD` matrix.

Usage

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

Arguments

`object`	An object of class `CARD`.

Value

A concise summary of the `CARD` object is printed to the console.

`show,CARDfree-method` *Show method for the CARDfree class*

Description

This method provides a concise summary of an object of class `CARDfree`, displaying key information including the project name, the number of spots, the number of cell types, and a sample of the `Proportion_CARD` matrix.

Usage

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

Arguments

`object`	An object of class `CARDfree`.

Value

A concise summary of the `CARDfree` object is printed to the console.

`Sigma` *Calculate the variance covariance matrix used in the imputation of the new grided locations*

Description

Calculate the variance covariance matrix used in the imputation of the new grided locations

Usage

```
Sigma(location_orig, train_ind, test_ind, optimal_phi, ineibor)
```

Arguments

`location_orig`	Data frame, spatial location data frame of the original spatial resolved transcriptomics dataset, stored in the spatialCoords(CARD_object)
`train_ind`	Vector, index of the original spatial locations
`test_ind`	Vector, index of the newly grided spatial locations
`optimal_phi`	Numeric, the optimal phi value stored in CARD_object
`ineibor`	Numeric, number of neighbors used in the imputation on newly grided spatial locations, default is 10.

Value

Return a list with the imputed Cell type composition Vtest matrix on the newly grided spatial locations and predicted normalized gene expression

---

`spatial_count` *Spatial transcriptomics count data*

Description

The spatial transcriptomics count data must be in the format of matrix or sparseMatrix, while each row represents a gene and each column represents a spatial location. The column names of the spatial data can be in the “XcoordxYcoord” (i.e., 10x10) format, but you can also maintain your original spot names, for example, barcode names.

Usage

```
data(spatial_count)
```

Format

An object of class `dgCMatrix` with 11000 rows and 428 columns.

`spatial_location` *Spatial location data*

Description

The spatial location data must be in the format of data frame while each row represents a spatial location, the first column represents the x coordinate and the second column represents the y coordinate. The rownames of the spatial location data frame should match exactly with the column names of the spatial_count.

Usage

```
data(spatial_location)
```

Format

An object of class `data.frame` with 428 rows and 2 columns.

Index