Package ‘DR.SC’

February 6, 2022

Type Package

Title Joint Dimension Reduction and Spatial Clustering

Version 2.5

Date 2022-02-06

Author Wei Liu [aut, cre],
      Xingjie Shi [aut],
      Jin Liu [aut]

Maintainer Wei Liu <wei.liu@duke-nus.edu.sg>

Description Joint dimension reduction and spatial clustering is conducted for
Single-cell RNA sequencing and spatial transcriptomics data, and more details can be referred to
Wei Liu, Xu Liao, Xiang Zhou, Shi-jie Shi and Jin Liu. (2021) <doi:10.1101/2021.12.25.474153>. It is not only computationally effi-
cient and scalable to the sample size increment, but also is capable of choosing the smooth-
ness parameter and the number of clusters as well.

License GPL-3

Depends parallel, spatstat.geom, R (>= 4.0.0)

Imports CompQuadForm, cowplot, ggplot2, GiRaF, MASS, Matrix, mclust,
      methods, purrr, S4Vectors, RColorBrewer, Rcpp (>= 1.0.5),
      Seurat, stats

Suggests knitr, rmarkdown

LinkingTo Rcpp, RcppArmadillo

VignetteBuilder knitr

Encoding UTF-8

NeedsCompilation yes

RoxygenNote 7.1.1

Repository CRAN

Date/Publication 2022-02-06 03:20:02 UTC
Joint dimension reduction and spatial clustering for scRNA-seq and spatial transcriptomics data

Usage

```r
### S3 method for class "Seurat"
DR.SC(seu, q=15, K=NULL, platform= "Visium",
       nfeatures=2000, K_set = seq(2, 10), variable.type="HVGs", ...)
```

Arguments

- **seu**: an object of class "Seurat". The details of this object are given under 'Details'.
- **q**: a positive integer, specify the number of latent features to be extracted, default as 15.
- **K**: a positive integer, specify the number of clusters, default as NULL. When K=NULL, it is automatically selected by MBIC criteria.
- **platform**: a string, specify the platform of the provided data, default as "Visium". There are more platforms to be chosen, including ("Visium", "ST", "seqfish", "merfish", "slide-seqv2", "seqscope", "hdst") and "scRNAseq", where the first group means there are spatial coordinates information in the metadata of seu, named...
"row" and "col" and a Hidden Markov random field is used to model the spatial coordinates, and the last one "scRNAseq" means there is no spatial information in object seu and a multinomial model is used to model the unobserved class labels. The platform helps to calculate the adjacency matrix.

nfeatures a positive integer, means how many highly variable or spatially variable genes used for DR-SC model, default as 2000. If there are less than 2000 features in seu, then all features are used for DR-SC model.

K_set an optional vector of positive integer, means the candidates of number of clusters used for MBIC.

variable.type an optional string, specify whether use highly variable genes (HVGs) or spatially variable genes (SVGs) to fit DR-SC model, default as "HVGs".

... Other arguments to pass into DR.SC_fit function.

Details

seu is an object named Seurat, thich can easily created by R package Seurat. If the data is collected by the spatial transcriptomics technologies such as 10X Visium, ST, seqFISH, MERFISH and Slide-seq, then there are spatial coordinates information in the metadata of seu, named "row" and "col". DR-SC model uses a Hidden Markov random field to model the spatial coordinates. If the data is collected by the single cell RNA sequencing techonologies which means there is no spatial information in object seu then a multinomial model is used to model the unobserved class labels.

Moreover, if variable.type=="HVGs", FindVariableFeatures function in Seurat is used to find the highly variable genes automatically; if variable.type=="SVGs", FindSVGs function in DR.SC is used to find the spatially variable genes automatically.

Value

DR.SC returns a revised Seurat object. There are two revisions in the seu. 1. the metadata is added a new column named spatial.drsc.cluster that represents the clustering results from DR-SC model, and the Idents(seu) is assigned with spatial.drsc.cluster. 2. a DimReduc object named dr-sc is added in the slot reductions, which represents the features extracted by DR-SC model.

Note

nothing

Author(s)

Wei Liu

References

None

See Also

None
**Examples**

```r
## we generate the spatial transcriptomics data with lattice neighborhood, i.e. ST platform.
seu <- gendata_RNAExp(height=10, width=10,p=50, K=4,platform="ST")
library(Seurat)
seu <- NormalizeData(seu, verbose=FALSE)
# choose 100 highly variable features
# seu <- FindVariableFeatures(seu, nfeatures = 100)
# maxIter = 2 is only used for illustration, and user can use default.
# seu1 <- DR.SC(seu, K=4, platform = 'ST', maxIter=2,verbose=FALSE)

# choose spatially variable features (SVGs)
seu <- FindSVGs(seu, nfeatures = 40, verbose=FALSE)
# use SVGs to fit DR.SC model
# maxIter = 2 is only used for illustration, and user can use default.
seu1 <- DR.SC(seu, K=4, platform = 'ST', maxIter=2,variable.type = 'SVGs',verbose=TRUE)
```

---

**DR.SC_fit**

*Joint dimension reduction and spatial clustering*

**Description**

Joint dimension reduction and spatial clustering for scRNA-seq and spatial transcriptomics data

**Usage**

```r
DR.SC_fit(X, Adj_sp=NULL, q=15, K= NULL,error.heter= TRUE, K_set = seq(2, 10),
beta_grid=seq(0.5, 5, by=0.5),maxIter=25, epsLogLik=1e-5, verbose=FALSE,
maxIter.ICM=6,pen.const=1,wpca.int=FALSE, parallel='parallel', num_core=5)
```

**Arguments**

- **X**: a sparse matrix with class dgCMatrix or matrix, specify the log-normalization gene expression matrix used for DR-SC model.
- **Adj_sp**: an optional sparse matrix with class dgCMatrix, specify the adjoint matrix used for DR-SC model. We provide this interface for those users who would like to define the adjoint matrix by their own.
- **q**: a positive integer, specify the number of latent features to be extracted, default as 15.
- **K**: a positive integer, specify the number of clusters, default as NULL. When K=NULL, it is automatically selected by MBIC criteria.
- **K_set**: a vector of positive integer, means the candidates of number of clusters used for MBIC.
- **error.heter**: an optional logical value, whether use the heterogenous error for DR-SC model, default as TRUE. If error.heter=FALSE, then the homogenous error is used for model.
DR.SC_fit

betagrid an optional vector of positive value, the candidate set of the smoothing parameter to be searched by the grid-search optimization approach.

maxIter an optional positive value, represents the maximum iterations of EM.

epsLogLik an optional positive value, tolerance value of relative variation rate of the observed pseudo log-loglikelihood value, default as \('1e-5\)'.

verbose an optional logical value, whether output the information of the ICM-EM algorithm.

maxIter_ICM an optional positive value, represents the maximum iterations of ICM.

pen.const an optional positive value, the adjusted constant used in the MBIC criteria. It usually takes value between 0.1 to 1.

wpca.int an optional logical value, means whether use the weighted PCA to obtain the initial values of loadings and other parameters, default as FALSE which means the conventional PCA is used.

parallel a optional string, specify the parallel way to choose the number of clusters by MBIC. We provide two methods: 1. parallel="parallel" uses parallel R package to conduct the parallel schema; 2. parallel=NULL doesn't use parallel computation.

num_core an optional positive integer, means the cores used in parallel computation.

Details

Nothing

Value

DR.SC_fit returns a list with the following components:

cluster inferred class labels

hZ extracted latent features.

beta estimated smoothing parameter

Mu mean vectors of mixtures components.

Sigma covariance matrix of mixtures components.

W estimated loading matrix

Lam_vec estimated variance of errors in probabilistic PCA model

loglik pseudo observed log-likelihood

Note

nothing

Author(s)

Wei Liu
## Examples

```r
## we generate the spatial transcriptomics data with lattice neighborhood, i.e. ST platform.
seu <- gendata_RNAExp(height=10, width=10, p=50, K=4)
library(Seurat)
seu <- NormalizeData(seu, verbose=FALSE)
# choose 40 highly variable features using FindVariableFeatures in Seurat
# seu <- FindVariableFeatures(seu, nfeatures = 40)
# or choose 40 spatially variable features using FindSVGs in DR.SC
seu <- FindSVGs(seu, nfeatures = 40, verbose=FALSE)
# users define the adjacency matrix
Adj_sp <- getAdj(seu, platform = 'ST')
var.features <- seu$assays$RNA@var.features
X <- Matrix::t(seu$"RNA"@data[var.features,])
# maxIter = 2 is only used for illustration, and user can use default.
drscList <- DR.SC_fit(X, Adj_sp=Adj_sp, K=4, maxIter=2, verbose=TRUE)
```

---

### drscPlot

**tNSE or UMAP plot visualization**

**Description**

Intuitive way of visualizing how cell types changes across the spatial locations.

**Usage**

```
drscPlot(seu, dims=1:5, visu.method='tSNE',...)
```

**Arguments**

- **seu**: an object of class "Seurat" obtained by DR.SC.
- **dims**: a positive integer to specify the number of latent features for visualization.
- **visu.method**: a string including 'tSNE' or "UMAP".
- **...**: Other arguments passing to DimPlot function.

**Details**

Nothing
FindSVGs

Value

return a ggplot2 object.

Note

nothing

Author(s)

Wei Liu

References

None

See Also

None

Examples

```r
## we generate the spatial transcriptomics data with lattice neighborhood, i.e. ST platform.
seu <- gendata_RNAExp(height=10, width=10,p=50, K=4)
library(Seurat)
seu <- NormalizeData(seu)
# choose spatially variable features
seu <- FindSVGs(seu)

# use SVGs to fit DR.SC model
# maxIter = 2 is only used for illustration, and user can use default.
seu1 <- DR.SC(seu, K=4,platform = 'ST', variable.type = 'SVGs',maxIter = 2,verbose=FALSE)
drscPlot(seu1)
```

FindSVGs

\textit{Find spatially variable genes}

Description

Identifies features that have spatially variation along spots.

Usage

\texttt{FindSVGs(seu, nfeatures=2000, covariates=NULL, num_core=1, verbose=TRUE)}
Arguments

seu an object of class "Seurat".

nfeatures a positive integer, means how many spatially variable genes to be chosen. If there are less than 2000 features in seu, then all features are identified.

covariates a covariate matrix named control variable matrix whose number of rows is equal to the number of columns of seu.

num_core an optional positive integer, specify the cores used for identifying the SVGs in parallel.

verbose an optional logical value, whether output the related information.

Details

Nothing

Value

return a revised Seurat object by adding three columns named "is.SVGs", "order.SVGs" and "adjusted.pval.SVGs" in the meta.features of default Assay.

Note

nothing

References


See Also
topSVGs

Examples

seu <- gendata_RNAExp(height=20, width=20,p=200, K=4)

seu <- FindSVGs(seu, nfeatures=100)
topSVGs(seu)
**Description**

Generate simulated spatial transcriptomics data or scRNAseq data.

**Usage**

```r
gedata_RNAExp(height=30, width=30, platform="ST", p =100, q=10, K=7, G=4,sigma2=1, tau=8, seed=1, view=FALSE)
```

**Arguments**

- `height`, `width`: Height and width of lattice grids for generating spatial coordinates. \( n=\text{height} \times \text{width} \) cells for scRNAseq data
- `platform`: set the platform for the simulated data, only support 'ST' and 'scRNAseq'.
- `p`: number of genes to generate.
- `q`: number of true latent features to generate gene expression
- `K`: number of clusters (cell types).
- `seed`: random seed for generate data
- `G`: the number of neighbors. The latter must be one of \( G = 4 \) or \( G = 8 \), which respectively correspond to a first order and a second order dependency structure. By default, \( G = 4 \).
- `sigma2`: Variance of error term in probabilistic PCA model.
- `tau`: a positive factor of mixture mean values.
- `view`: Logical value indicating whether the draw should be printed. Do not display the optional borders.

**Details**

Nothing

**Value**

return a "Seurat" object. If `platform="ST"`, then the metadata of this Seurat object will include two columns with names "row" and "col" which are the spatial coordinates; If `platform="scRNAseq"`, then the metadata of this Seurat object will not have them.

**Note**

nothing

**Author(s)**

Wei Liu
getAdj

Calculate the adjacency matrix given the spatial coordinates

Description

Calculate the adjacency matrix for the spatial transcriptomics data measured on 10X Visium or other platforms as a Seurat object.

Usage

```r
getAdj(obj, platform = c("Visium", "ST", "seqfish", "merfish", "slide-seqv2", "seqscope", "HDST"))
```

Arguments

- **obj**: an object with class "Seurat", there are spatial coordinates information in the metadata of obj, named "row" and "col", where first column is x-axis coordinate, the second column is y-axis coordinate. `getAdj_manual` and `getAdj_auto` supports multi-dimensional spatial coordinates with a matrix as input.
- **platform**: a string, specify the platform of the provided data, default as "Visium". There are many platforms to be supported, including ("Visium", "ST", "SeqFISH", "merFISH", "slide-seqv2", "seqscope", "HDST"), which means there are spatial coordinates information in the metadata of seu, named "row" and "col". The platform helps to calculate the adjacency matrix by defining the neighborhoods.
**getAdj_auto**

**Details**

For lattice grids, i.e., two-dimensional coordinates, the interior spot has four neighbors (left, right, up and down), the boundary spot has three neighbors, and the spot in the corner has two neighbors. For hexagon grids, such as spatial coordinate in 10X Visium platform, the interior spot has six neighbors. More flexible definition can be used if there are some additional information. And then use `getAdj_manual` function to evaluate the adjacency matrix as a input for `DR.SC_fit` to run DR-SC model.

**Value**

Return a `dgCMatrix` object recording the information of neighborhoods about each spot.

**Note**

nothing

**Author(s)**

Wei Liu

**References**

None

**See Also**

`getAdj_auto, getAdj_manual`.

**Examples**

```r
## S3 method for class "Seurat"
seu <- gendata_RNAExp(height=20, width=20, p=200, K=4)
Adj_sp <- getAdj(seu, platform = 'ST')
```

### Description

an efficient function to find the radius by bi-section method then find neighborhoods based on the matrix of position and the chosen radius, which ensures that each spot has about 4–6 neighborhoods in the sense of median and shows it works well.

**Usage**

`getAdj_auto(pos)`
getAdj_manual

Arguments

pos is a n-by-2 matrix of position.

Value

A sparse adjoint matrix containing the neighbourhood.

See Also

getAdj, getAdj_manual.

getAdj_manual Calculate adjacency matrix by user-specified radius

Description

an efficient function to find the neighborhood based on the matrix of position and a pre-defined radius.

Usage

getAdj_manual(pos, radius)

Arguments

pos is a n-by-d matrix of position, where n is the number of spots, and d is the dimension of coordinates.

radius is a threshold of Euclidean distance to decide whether a spot is a neighborhood of another spot. For example, if the Euclidean distance between spot A and B is less than radius, then A is taken as the neighbourhood of B.

Value

A sparse matrix containing the neighbourhood

See Also

getAdj_auto, getAdj.
getneighborhood_fast

Description
an efficient function to find the neighborhood based on the matrix of position and a pre-defined cutoff

Usage
getneighborhood_fast(x, radius)

Arguments
x
is a n-by-2 matrix of position.

radius
is a threshold of Euclidean distance to decide whether a spot is an neighborhood of another spot. For example, if the Euclidean distance between spot A and B is less than cutoff, then A is taken as the neighbourhood of B.

Value
A sparse matrix containing the neighbourhood

HCC1
A liver dataset on Visium platform

Description
A liver dataset measured on the Visium platform, which includes 2983 spots and 2000 genes.

Note
nothing

Author(s)
Wei Liu

References
None

Examples
data("HCC1")
**mbicPlot**  
*MBIC plot visualization*

**Description**
Intuitive way of visualizing how modified BIC values changes across different number of clusters

**Usage**
```
mbicPlot(seu)
```

**Arguments**
- **seu**: an object of class "Seurat" revised by `DR.SC` with argument `K=NULL`.

**Details**
Nothing

**Value**
return a ggplot2 object.

**Note**
nothing

**Author(s)**
Wei Liu

**References**
None

**See Also**
None

**Examples**
```r
## we generate the spatial transcriptomics data with lattice neighborhood, i.e. ST platform.
seu <- gendata_RNAExp(height=20, width=20,p=100, K=4)
library(Seurat)
seu <- NormalizeData(seu)
# choose spatially variable features
seu <- FindSVGs(seu)
## Just for illustrating the usage of mbicPlot
```
seu@tools$icMat <- data.frame(K=2:5, mbic=c(105, 101, 99, 108))
mbicPlot(seu)

---

### read10XVisium

**Read the spatial transcriptomics data measured on 10X Visium platform**

**Description**

Read the spatial transcriptomics data measured on 10X Visium platform as a Seurat object, where the spatial coordinates are saved in the metadata, named "row" and "col".

**Usage**

```r
read10XVisium(dirname)
```

**Arguments**

- `dirname` A string, the directory of Visium datasets.

**Details**

Nothing

**Value**

Return a Seurat object.

**Note**

Nothing

**Author(s)**

Wei Liu

**References**

None

**See Also**

None
## Description

Read the single cell RNA sequencing data measured on scRNA sequencing platform as a Seurat object.

## Usage

`readscRNAseq(mtx, cells, features, ...)`

## Arguments

- `mtx`  
  a string, name or remote URL of the mtx file
- `cells`  
  a string, Name or remote URL of the cells/barcodes file
- `features`  
  a string, Name or remote URL of the features/genes file
- `...`  
  the arguments passing to `ReadMtx`

## Details

Nothing

## Value

return a Seurat object including expression matrix.

## Note

nothing

## Author(s)

Wei Liu

## References

None
RunWPCA

See Also
None

Examples

```r
## Not run:
### set the file directory, then read it.
seu <- readscRNAseq(mtx="GSM3755564_16_Liver_Treg_matrix.mtx.gz",
                    features="GSM3755564_16_Liver_Treg_genes.tsv.gz",
                    cells="GSM3755564_16_Liver_Treg_barcodes.tsv.gz")

seu

## End(Not run)
```

RunWPCA

Run Weighted Principal Component Analysis

Description
Run a weighted PCA dimensionality reduction

Usage

```r
RunWPCA(object, q=15)

### S3 method for class "Seurat"
## RunWPCA(object, q=15)

### S3 method for class "matrix"
## RunWPCA(object, q=15)

### S3 method for class "dgCMatrix"
## RunWPCA(object, q=15)
```

Arguments

- `object` an object named "Seurat", "maxtrix" or "dgCMatrix". The object of class "Seurat" must include slot "scale.data".
- `q` an optional positive integer, specify the number of features to be extracted.

Details
Nothing

Value
For Seurat object, return a Seurat object. For object "matrix" and "dgCMatrix", return a object "matrix" with rownames same as the colnames of X, and colnames "WPCA1" to "WPCAz".
spatialPlotClusters

Note
nothing

Author(s)
Wei Liu

References

See Also
None

Examples
## Not run:
library(Seurat)
seu <- gendata_RNAExp(height=20, width=20,p=100, K=4)
## log-normalization
seu <- NormalizeData(seu)
##
seu <- FindVariableFeatures(seu, nfeatures=80)
## Scale
seu <- ScaleData(seu)
## Run WPCA
seu <- RunWPCA(seu)
##
seu <- RunTSNE(seu, reduction='wpca')
## Find SVGs
seu <- FindSVGs(seu, nfeatures=80)
(genes <- topSVGs(seu, ntop=10))
Idents(seu) <- factor(paste0("cluster", seu$true_clusters), levels=paste0("cluster",1:4))
RidgePlot(seu, features = genes[1:2], ncol = 2)
FeaturePlot(seu, features = genes[1:2], reduction = 'tsne',ncol=2)
## End(Not run)

spatialPlotClusters  Spatial coordinates plot visualization

Description
Intuitive way of visualizing how cell types changes across the spatial locations.
Usage

spatialPlotClusters(seu)

Arguments

seu  
an object of class "Seurat" obtained by DR.SC.

Details

Nothing

Value

return a ggplot2 object.

Note

nothing

Author(s)

Wei Liu

References

None

See Also

None

Examples

## we generate the spatial transcriptomics data with lattice neighborhood, i.e. ST platform.
seu <- gendata_RNAExp(height=10, width=10,p=50, K=4)
library(Seurat)
seu <- NormalizeData(seu)
# choose spatially variable features using Seurat
seu <- FindSVGs(seu)
# use SVGs to fit DR.SC model
# maxIter = 2 is only used for illustration, and user can use default.
seu1 <- DR.SC(seu, K=4,platform = 'ST', variable.type = 'SVGs',maxIter=2,verbose=FALSE)
spatialPlotClusters(seu1)
sp_means_Rcpp

*Description*

Calculate column-wise or row-wise mean

*Usage*

```r
sp_means_Rcpp(sp_data, rowMeans = FALSE)
```

*Arguments*

- `sp_data` A sparse matrix
- `rowMeans` A boolean value, whether to calculate row-wise mean

*Value*

A n x 1 or p x 1 matrix

---

sp_sums_Rcpp

*Description*

Calculate column-wise or row-wise sum

*Usage*

```r
sp_sums_Rcpp(sp_data, rowSums = FALSE)
```

*Arguments*

- `sp_data` A sparse matrix
- `rowSums` A boolean value, whether to calculate row-wise sum

*Value*

A n x 1 or p x 1 matrix
Description

Return top n spatially variable genes given a Seurat object performed by \texttt{FindSVGs}.

Usage

\texttt{topSVGs(seu, ntop=5)}

Arguments

- \texttt{seu}: an object of class "Seurat".
- \texttt{ntop}: an optional positive integer, means how many spatially variable genes to access.

Details

Nothing

Value

return a \texttt{character} vector including the names of SVGs.

Note

nothing

Author(s)

Wei Liu

References

None

See Also

\texttt{topSVGs}

Examples

seu <- gendata_RNAExp(height=20, width=20, p=200, K=4)
seu <- FindSVGs(seu, nfeatures=100, verbose=FALSE)
(genes <- topSVGs(seu, ntop=10))
Index

* **Adjacency matrix**
  getAdj, 10

* **DR-SC**
  drscPlot, 6

* **Dimension reduction**
  DR.SC, 2
  DR.SC_fit, 4
  RunWPCA, 17

* **Hexagon grids**
  getAdj, 10

* **Input of scRNAseq data**
  readscRNAseq, 16

* **Input of spatial data**
  FindSVGs, 7
  read10XVisium, 15

* **Lattice grids**
  getAdj, 10

* **Modified BIC**
  mbicPlot, 14

* **SVGs**
  topSVGs, 21

* **Spatial clustering**
  DR.SC, 2

* **Spatial plot**
  spatialPlotClusters, 18

* **UMAP**
  drscPlot, 6

* **Weighted PCA**
  RunWPCA, 17

* **spatial clustering and cell type identification**
  DR.SC_fit, 4

* **tSNE**
  drscPlot, 6

character, 21

DimPlot, 6
DR.SC, 2, 3, 6, 14, 19
DR.SC_fit, 3, 4, 11

drscPlot, 6
FindSVgs, 7, 21
FindVariableFeatures, 3
gendata.RNAExp, 9
getAdj, 10, 12
getAdj_auto, 10, 11, 12
getAdj_manual, 10–12, 12
getneighborhood_fast, 13
HCC1, 13
list, 5
mbicPlot, 14
read10XVisium, 15
ReadMtx, 16
readscRNAseq, 16
RunWPCA, 17
Seurat, 3
sp_means_Rcpp, 20
sp_sums_Rcpp, 20
spatialPlotClusters, 18
topSVGs, 8, 21, 21