

Package ‘SCArray’

October 12, 2021

Type Package

Title Large-scale single-cell RNA-seq data manipulation with GDS files

Version 1.0.0

Date 2021-04-30

Depends R (>= 3.5.0), gdsfmt (>= 1.27.4), methods, DelayedArray

Imports BiocGenerics, S4Vectors, IRanges, utils, SummarizedExperiment, SingleCellExperiment

Suggests Matrix, DelayedMatrixStats, scater, uwot, RUnit, knitr, markdown, rmarkdown

Description Provides large-scale single-cell RNA-seq data manipulation using Genomic Data Structure (GDS) files. It combines dense and sparse matrices stored in GDS files and the Bioconductor infrastructure framework (SingleCellExperiment and DelayedArray) to provide out-of-memory data storage and large-scale manipulation using the R programming language.

License GPL-3

VignetteBuilder knitr

ByteCompile TRUE

URL <https://github.com/AbbVie-ComputationalGenomics/SCArray>

biocViews Infrastructure, DataRepresentation, DataImport, SingleCell, RNASeq

git_url <https://git.bioconductor.org/packages/SCArray>

git_branch RELEASE_3_13

git_last_commit 0faf080

git_last_commit_date 2021-05-19

Date/Publication 2021-10-12

Author Xiuwen Zheng [aut, cre] (<<https://orcid.org/0000-0002-1390-0708>>)

Maintainer Xiuwen Zheng <xiuwen.zheng@abbvie.com>

R topics documented:

SCArray-package	2
scArray	2
scClose	3
scConvGDS	4
scExperiment	5
scOpen	6
Index	8

SCArray-package	<i>Large-scale single-cell RNA-seq data manipulation with GDS files</i>
-----------------	---

Description

The package combines dense/sparse matrices stored in GDS files and the Bioconductor infrastructure framework to provide out-of-memory data storage and manipulation using the R programming language.

Details

Package: SCArray
Type: Package
License: GPL version 3

Author(s)

Xiuwen Zheng <xiuwen.zheng@abbvie.com>

Examples

```
# a GDS file for SingleCellExperiment
fn <- system.file("extdata", "LaMannoBrainData.gds", package="SCArray")

sce <- scExperiment(fn)
sce
```

scArray	<i>Get an DelayedArray instance</i>
---------	-------------------------------------

Description

Gets an DelayedArray instance from a single-cell GDS file.

Usage

```
scArray(gdsfile, varname)
```

Arguments

gdsfile	character for a file name, or a single-cell GDS object with class <code>SCArrayFileClass</code>
varname	character for the node name in the GDS file

Value

Return an object of class `DelayedArray`.

Author(s)

Xiuwen Zheng

See Also

[scOpen](#), [scExperiment](#)

Examples

```
# a GDS file for SingleCellExperiment
fn <- system.file("extdata", "LaMannoBrainData.gds", package="SCArray")

cnt <- scArray(fn, "counts")
cnt
```

scClose	<i>Close the Single-cell GDS File</i>
---------	---------------------------------------

Description

Closes a single-cell GDS file which is open.

Usage

```
scClose(gdsfile)
```

Arguments

gdsfile	a single-cell GDS object with class <code>SCArrayFileClass</code>
---------	---

Value

None.

Author(s)

Xiuwen Zheng

See Also

[scOpen](#)

Examples

```
# a GDS file for SingleCellExperiment
fn <- system.file("extdata", "LaMannoBrainData.gds", package="SCArray")

# open the GDS file
(f <- scOpen(fn))

# read
cell.id <- read.gdsn(index.gdsn(f, "feature.id"))
samp.id <- read.gdsn(index.gdsn(f, "sample.id"))

scClose(f)
```

scConvGDS

Create a GDS file

Description

Creates a single-cell GDS file from an R object.

Usage

```
scConvGDS(obj, outfn, save.sp=TRUE, type=c("float32", "float64", "int32"),
  compress="LZMA_RA", clean=TRUE, verbose=TRUE)
```

Arguments

obj	a dense/sparse matrix, DelayedMatrix, SummarizedExperiment or SingleCellExperiment
outfn	the output file name in GDS format
save.sp	if TRUE, save it to a sparse matrix in GDS; otherwise, store dense matrix
type	numeric data type in the output file
compress	the compression method, see add.gdsn
clean	TRUE
verbose	if TRUE, show information

Value

Return the path of the output file.

Author(s)

Xiuwen Zheng

See Also[scOpen](#), [scClose](#)**Examples**

```
# load a SingleCellExperiment object
fn <- system.file("extdata", "LaMannoBrainSub.rds", package="SCArray")
sce <- readRDS(fn)

scConvGDS(sce, "test.gds")

# remove the temporary output file
unlink("test.gds")
```

scExperiment*Get a SummarizedExperiment*

Description

Gets an instance of `SingleCellExperiment` or `SummarizedExperiment`.

Usage

```
scExperiment(gdsfile, sce=TRUE, use.names=TRUE, load.row=TRUE, load.col=TRUE)
```

Arguments

<code>gdsfile</code>	character for a file name, or a single-cell GDS object with class <code>SCArrayFileClass</code>
<code>sce</code>	if <code>TRUE</code> , return an instance of <code>SingleCellExperiment</code> , otherwise an instance of <code>SummarizedExperiment</code>
<code>use.names</code>	if <code>TRUE</code> , load dimnames from 'feature.id' and 'sample.id'
<code>load.row</code>	<code>TRUE</code> for loading <code>rowData</code> from the gds node "feature.data" in <code>gdsfile</code>
<code>load.col</code>	<code>TRUE</code> for loading <code>colData</code> from the gds node "sample.data" in <code>gdsfile</code>

Value

Return an instance of [SingleCellExperiment](#) or [SummarizedExperiment](#).

Author(s)

Xiuwen Zheng

See Also

[scOpen](#), [scClose](#)

Examples

```
# a GDS file for SingleCellExperiment
fn <- system.file("extdata", "LaMannoBrainData.gds", package="SCArray")

sce <- scExperiment(fn)
sce
```

scOpen	<i>Open a Single-cell GDS File</i>
--------	------------------------------------

Description

Opens a single-cell GDS file.

Usage

```
scOpen(gdsfn, readonly=TRUE, allow.duplicate=TRUE)
```

Arguments

gdsfn	the input file name
readonly	whether read-only or not
allow.duplicate	if TRUE, it is allowed to open a GDS file with read-only mode when it has been opened in the same R session

Value

Return an object of class `SCArrayFileClass` inherited from [gds.class](#).

Author(s)

Xiuwen Zheng

See Also

[scClose](#), [scArray](#)

Examples

```
# a GDS file for SingleCellExperiment
fn <- system.file("extdata", "LaMannoBrainData.gds", package="SCArray")

# open the GDS file
(f <- scOpen(fn))

# read a GDS file
cell.id <- read.gdsn(index.gdsn(f, "feature.id"))
samp.id <- read.gdsn(index.gdsn(f, "sample.id"))

# get a DelayedArray object
(cnt <- scArray(f, "counts"))

scClose(f)
```

Index

* **GDS**

- scArray, [2](#)
- SCArray-package, [2](#)
- scClose, [3](#)
- scConvGDS, [4](#)
- scExperiment, [5](#)
- scOpen, [6](#)

* **SingleCell**

- scArray, [2](#)
- SCArray-package, [2](#)
- scClose, [3](#)
- scConvGDS, [4](#)
- scExperiment, [5](#)
- scOpen, [6](#)

add.gdsn, [4](#)

DelayedArray, [3](#)

gds.class, [6](#)

- scArray, [2](#), [6](#)
- SCArray-package, [2](#)
- scClose, [3](#), [5](#), [6](#)
- scConvGDS, [4](#)
- scExperiment, [3](#), [5](#)
- scOpen, [3–6](#), [6](#)
- SingleCellExperiment, [5](#)
- SummarizedExperiment, [5](#)