

Package ‘scviR’

November 14, 2025

Date 2025-05-02

Title experimental interface from R to scvi-tools

Version 1.11.0

Description This package defines interfaces from R to scvi-tools. A vignette works through the totalVI tutorial for analyzing CITE-seq data. Another vignette compares outputs of Chapter 12 of the OSCA book with analogous outputs based on totalVI quantifications. Future work will address other components of scvi-tools, with a focus on building understanding of probabilistic methods based on variational autoencoders.

License Artistic-2.0

Encoding UTF-8

Depends R (>= 4.3), basilisk, shiny, SingleCellExperiment

Imports reticulate, BiocFileCache, utils, pheatmap, SummarizedExperiment, S4Vectors, limma, scatter, stats, MatrixGenerics

Suggests knitr, testthat, reshape2, ggplot2, rhdf5, BiocStyle

VignetteBuilder knitr

biocViews Infrastructure, SingleCell, DataImport

RoxygenNote 7.3.3

URL <https://github.com/vjcitn/scviR>

BugReports <https://github.com/vjcitn/scviR/issues>

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

git_branch devel

git_last_commit 4329942

git_last_commit_date 2025-10-29

Repository Bioconductor 3.23

Date/Publication 2025-11-13

Author Vincent Carey [aut, cre] (ORCID:
<<https://orcid.org/0000-0003-4046-0063>>)

Maintainer Vincent Carey <stvjc@channing.harvard.edu>

Contents

adtProfiles	2
anndataR	3
bsklenv	4
cacheCiteseq5k10kPbmcs	4
cacheCiteseq5k10kTutvae	5
cacheCiteseqHDPdata	6
cacheCiteseqHDPmodel	6
clusters.adt	7
clusters.rna	7
exploreSubcl	7
getCh12AllSce	8
getCh12Sce	9
getCiteseq5k10kPbmcs	10
getCiteseqTutvae	10
getPro5k10kAdata	11
getSubclLM	11
getSubclusteringFeatures	12
getTotalVI5k10kAdata	13
getTotalVINormalized5k10k	13
MuDataR	14
muonR	14
pyHelp2	15
scanpyHelper	15
scanpyR	16
scviHelper	16
scviR	17
Index	18

adtProfiles	<i>produce a heatmap from a specialized CITE-seq SingleCellExperiment</i>
-------------	---

Description

produce a heatmap from a specialized CITE-seq SingleCellExperiment

Usage

adtProfiles(x, lb = -3, ub = 3, do_z = FALSE)

Arguments

x	SingleCellExperiment instance that has an 'se.averaged' component in its meta-data
lb	numeric(1) lower bound on 'breaks' sequence for ComplexHeatmap::pheatmap, defaults to -3
ub	numeric(1) upper bound on 'breaks' sequence for ComplexHeatmap::pheatmap, defaults to 3
do_z	logical(1) if TRUE, divide the residuals by their standard deviation across clusters, defaults to false

Value

ComplexHeatmap::pheatmap instance
side effect of pheatmap::pheatmap call

Note

See the OSCA book ch12.5.2 for the application.

Examples

```
ch12sce <- getCh12Sce()
adtProfiles(ch12sce)
adtProfiles(ch12sce, do_z = TRUE)
```

anndataR

basic interface to anndata

Description

basic interface to anndata

Usage

```
anndataR()
```

Value

basiliskRun result with import from reticulate, typically a Module

Examples

```
ad <- anndataR()
ad
ad$read_h5ad
```

bsk1env	<i>python declarations</i>
---------	----------------------------

Description

python declarations

Usage

bsk1env

Format

An object of class BasiliskEnvironment of length 1.

cacheCiteSeq5k10kPbmcs

grab scvi-tools-processed PBMC CITE-seq data in anndata format (gzipped) from Open Storage Network

Description

grab scvi-tools-processed PBMC CITE-seq data in anndata format (gzipped) from Open Storage Network

Usage

cacheCiteSeq5k10kPbmcs()

Value

invisibly, the path to the .h5ad file

Note

Original h5ad files obtained using scvi-tools 0.18.0 `scvi.data.pbmcs_10x_cite_seq`, then processed according to steps in the scviR vignette, which follow the [scvi-tools tutorial](<https://colab.research.google.com/github/scverse/tutorials/blob/0.18.0/totalVI.ipynb>) by Gayoso et al.

It may be advantageous to set `'options(timeout=3600)'` or to allow an even greater time for internet downloads, if working at a relatively slow network connection.

Examples

```
h5path <- cacheCiteSeq5k10kPbmcs()
cmeta <- rhdf5::h5ls(h5path)
dim(cmeta)
head(cmeta, 17)
```

`cacheCiteseq5k10kTutvae`

Deprecated: grab scvi-tools VAE instance built on the PBMC datasets following the tutorial

Description

Deprecated: grab scvi-tools VAE instance built on the PBMC datasets following the tutorial

Usage

```
cacheCiteseq5k10kTutvae()
```

Value

invisibly, the path to the .zip file holding the fitted VAE and associated data

Note

the serialized model is obsolete

VAE construction followed tutorial at <https://docs.scvi-tools.org/en/stable/tutorials/notebooks/totalVI.html>.

It may be advantageous to set `'options(timeout=3600)'` or to allow an even greater time for internet downloads, if working at a relatively slow network connection.

Examples

```
## Not run:
zpath <- cacheCiteseq5k10kTutvae()
td <- tempdir()
utils::unzip(zpath, exdir = td)
vaedir <- paste0(td, "/vae2_ov")
scvi <- scviR()
adm <- anndataR()
hpath <- cacheCiteseq5k10kPbmcs()
adata <- adm$read_h5ad(hpath)
mod <- scvi$model$`_totalvi`$TOTALVI$load(vaedir, adata) #, use_gpu = FALSE)
mod

## End(Not run)
```

cacheCiteSeqHDPdata	<i>retrieve and cache a 349-protein CITE-seq dataset as employed in scvi-tools tutorial</i>
---------------------	---

Description

retrieve and cache a 349-protein CITE-seq dataset as employed in scvi-tools tutorial

Usage

```
cacheCiteSeqHDPdata()
```

cacheCiteSeqHDPmodel	<i>grab scvi-tools muon-oriented VAE instance built on the PBMC datasets following the tutorial</i>
----------------------	---

Description

grab scvi-tools muon-oriented VAE instance built on the PBMC datasets following the tutorial

Usage

```
cacheCiteSeqHDPmodel()
```

Value

invisibly, the path to the .zip file holding the weights in pt format for the fitted VAE

Note

VAE construction followed tutorial at '<https://docs.scvi-tools.org/en/stable/tutorials/notebooks/totalVI.html>'.

We are using the scvi tutorial read early may 2025. The notebook uses "h5 format of single-cell multiomic data generated by Proteintech Genomics ... The data is from human resting PBMCs stained with the MultiPro® Human Discovery Panel (HDP) followed by processing using 10x Genomics Flex chemistry with Feature Barcoding Technology."

It may be advantageous to set 'options(timeout=3600)' or to allow an even greater time for internet downloads, if working at a relatively slow network connection.

Examples

```
zpath <- cacheCiteSeqHDPmodel()
td <- tempdir()
utils::unzip(zpath, exdir = td)
vaedir <- paste0(td, "/vae3_pt")
dir(vaedir)
```

clusters.adt	<i>ADT-based cluster labels for 7472 cells in OSCA chapter 12 analysis</i>
--------------	--

Description

ADT-based cluster labels for 7472 cells in OSCA chapter 12 analysis

Usage

clusters.adt

Format

factor

clusters.rna	<i>mRNA-based cluster labels for 7472 cells in OSCA chapter 12 analysis</i>
--------------	---

Description

mRNA-based cluster labels for 7472 cells in OSCA chapter 12 analysis

Usage

clusters.rna

Format

factor

exploreSubcl	<i>app to explore diversity in RNA-subclusters within ADT clusters</i>
--------------	--

Description

app to explore diversity in RNA-subclusters within ADT clusters

Usage

exploreSubcl(sce, inlist, adtcls)

Arguments

sce	a SingleCellExperiment with altExp with ADT quantification
inlist	list of SingleCellExperiments (SCEs) formed by <code>scrn::quickSubCluster</code>
adtcls	vector of ADT cluster assignments

Value

shinyApp instance

Note

TSNE should already be available in ‘altExp(sce)’; follow OSCA book 12.5.2. If using example, set ‘ask=FALSE’.

Examples

```
if (interactive()) {
  sce <- getCh12Sce()
  all.sce <- getCh12AllSce()
  data(clusters.adt)
  runApp(exploreSubcl(sce, all.sce, clusters.adt)) # trips up interactive pkgdown?
}
```

getCh12AllSce	<i>get list of cluster-specific SCE for 10k PBMC annotated as in OSCA book chapter 12</i>
---------------	---

Description

get list of cluster-specific SCE for 10k PBMC annotated as in OSCA book chapter 12

Usage

```
getCh12AllSce()
```

Value

SimpleList of SingleCellExperiment instances

Note

This is a list of SingleCellExperiment instances with data on a total of 7472 cells from a 10x CITE-seq experiment. An altExp component in each list element includes antibody-derived tag (ADT) counts on 17 proteins. The data are acquired and processed as described in ch 12 of the OSCA book, circa February 2023. List elements correspond to mRNA-based sub-clusters of ADT-based clusters.

Examples

```
ch12_allsce <- getCh12AllSce()
vapply(ch12_allsce, ncol, numeric(1))
```

getCh12Sce	<i>get SCE for 10k PBMC annotated as in OSCA book chapter 12</i>
------------	--

Description

get SCE for 10k PBMC annotated as in OSCA book chapter 12

Usage

```
getCh12Sce(clear_cache = FALSE)
```

Arguments

`clear_cache` logical(1) will delete relevant entries in available cache before continuing, defaults to FALSE

Value

SingleCellExperiment instance

Note

This is a SingleCellExperiment instance with data on 7472 cells from a 10x CITE-seq experiment. An altExp component includes antibody-derived tag (ADT) counts on 17 proteins. The data are acquired and processed as described in ch 12 of the OSCA book, circa February 2023. A metadata element (se.averaged) includes the result of averaging protein abundance estimates within ADT-based clusters, as is done to give rise to Figure 12.8 of the OSCA book.

Examples

```
ch12sce <- getCh12Sce()
ch12sce
```

`getCiteSeq5k10kPbmcs` *helper to get the processed anndata for CITE-seq PBMCs from scvi-tools tutorial*

Description

helper to get the processed anndata for CITE-seq PBMCs from scvi-tools tutorial

Usage

```
getCiteSeq5k10kPbmcs()
```

Value

python reference to anndata

Note

It may be advantageous to set 'options(timeout=3600)' or to allow an even greater time for internet downloads, if working at a relatively slow network connection.

Examples

```
getCiteSeq5k10kPbmcs()
```

`getCiteSeqTutvae` *helper to get the tutorial VAE for PBMCs from scvi-tools tutorial*

Description

helper to get the tutorial VAE for PBMCs from scvi-tools tutorial

Usage

```
getCiteSeqTutvae(use_gpu = FALSE)
```

Arguments

`use_gpu` logical(1), defaulting to FALSE, passed to TOTALVI.load

Value

python reference to anndata

Note

March 2024 use_gpu ignored

Examples

```
## Not run:
getCiteseqTutvae()

## End(Not run)
```

getPro5k10kAdata	<i>get an anndata reference to 5k10k protein after totalVI from tutorial</i>
------------------	--

Description

get an anndata reference to 5k10k protein after totalVI from tutorial

Usage

```
getPro5k10kAdata()
```

Value

python reference to anndata

Note

It may be advantageous to set ‘options(timeout=3600)’ or to allow an even greater time for internet downloads, if working at a relatively slow network connection.

Examples

```
getPro5k10kAdata()
```

getSubclLM	<i>get lmFit for heterogeneity across subclusters</i>
------------	---

Description

get lmFit for heterogeneity across subclusters

Usage

```
getSubclLM(inlist, clname)
```

Arguments

inlist	list of SingleCellExperiments (SCEs) formed by <code>scrn::quickSubCluster</code>
clname	character(1) name of cluster SCE to assess

Value

limma::lmFit output

Note

It is assumed that 'logcounts' is an assay element, and that 'subcluster' is a colData element of each SCE in inlist

Examples

```
all.sce <- getCh12AllSce()
lm3 <- getSubclLM(all.sce, "3")
names(lm3)
```

getSubclusteringFeatures

get lmFit F-stat based collection of n genes most varying in mean across subclusters

Description

get lmFit F-stat based collection of n genes most varying in mean across subclusters

Usage

```
getSubclusteringFeatures(inlist, cname, n = 20)
```

Arguments

inlist	list of SingleCellExperiments (SCEs) formed by <code>scrn::quickSubCluster</code>
cname	character(1) name of cluster SCE to assess
n	numeric(1) number to preserve

Value

list with two elements, `feat` = rowData corresponding to variable genes, `stats` = topTable result

Note

Symbol will be taken from `feat` and placed in `stats` component if available

Examples

```
all.sce <- getCh12AllSce()
scl <- getSubclusteringFeatures(all.sce, "3", 10)
names(scl)
```

getTotalVI5k10kAdata *get anndata reference to full totalVI processing of 5k10k data*

Description

get anndata reference to full totalVI processing of 5k10k data

Usage

```
getTotalVI5k10kAdata()
```

Value

python reference to anndata

Examples

```
full <- getTotalVI5k10kAdata()
full
```

getTotalVINormalized5k10k
get matrices of normalized quantifications from full totalVI 5k10k from tutorial

Description

get matrices of normalized quantifications from full totalVI 5k10k from tutorial

Usage

```
getTotalVINormalized5k10k()
```

Value

list of matrices

Examples

```
nm1ist <- getTotalVINormalized5k10k()
vapply(nm1ist, dim, numeric(2))
```

MuDataR	<i>basic interface to MuData</i>
---------	----------------------------------

Description

basic interface to MuData

Usage

```
MuDataR()
```

Value

basiliskRun result with import from reticulate, typically a Module

Examples

```
md <- MuDataR()
md
head(names(md))
```

muonR	<i>basic interface to muon</i>
-------	--------------------------------

Description

basic interface to muon

Usage

```
muonR()
```

Value

basiliskRun result with import from reticulate, typically a Module

Examples

```
md <- muonR()
md
head(names(md))
```

pyHelp2	<i>helper to get text from python help utility – may need handling through basilisk</i>
---------	---

Description

helper to get text from python help utility – may need handling through basilisk

Usage

```
pyHelp2(object)
```

Arguments

object a reference to a python module typically with class 'python.builtin.module'

Value

character vector of lines from python help result

scanpyHelper	<i>shiny app that helps access documentation on python-accessible components</i>
--------------	--

Description

shiny app that helps access documentation on python-accessible components

Usage

```
scanpyHelper()
```

Value

shinyApp instance

`scanpyR`*basic interface*

Description

basic interface

Usage

```
scanpyR()
```

Value

basiliskRun result with import from reticulate, typically a Module

Examples

```
sc <- scanpyR()
sc
sc$pp
```

`scviHelper`*shiny app that helps access documentation on python-accessible components*

Description

shiny app that helps access documentation on python-accessible components

Usage

```
scviHelper()
```

Value

shinyApp instance

`scviR`

*basic interface***Description**

basic interface

Usage

```
scviR()
```

Value

basiliskRun result with import from reticulate, typically a Module

Examples

```
scvi <- scviR()  
scvi  
scvi$model
```

Index

* datasets

- bsklenv, [4](#)
- clusters.adt, [7](#)
- clusters.rna, [7](#)

adtProfiles, [2](#)

anndataR, [3](#)

bsklenv, [4](#)

cacheCiteSeq5k10kPbmcs, [4](#)

cacheCiteSeq5k10kTutvae, [5](#)

cacheCiteSeqHDPdata, [6](#)

cacheCiteSeqHDPmodel, [6](#)

clusters.adt, [7](#)

clusters.rna, [7](#)

exploreSubcl, [7](#)

getCh12AllSce, [8](#)

getCh12Sce, [9](#)

getCiteSeq5k10kPbmcs, [10](#)

getCiteSeqTutvae, [10](#)

getPro5k10kAdata, [11](#)

getSubclLM, [11](#)

getSubclusteringFeatures, [12](#)

getTotalVI5k10kAdata, [13](#)

getTotalVINormalized5k10k, [13](#)

MuDataR, [14](#)

muonR, [14](#)

pyHelp2, [15](#)

scanpyHelper, [15](#)

scanpyR, [16](#)

scviHelper, [16](#)

scviR, [17](#)