Package 'scATAC.Explorer'

April 6, 2023

Title A Collection of Single-cell ATAC Sequencing Datasets and
Corresponding Metadata

Version 1.4.0

Description

This package provides a tool to search and download a collection of publicly available single cell
ATAC-seq datasets and their metadata. scATAC-Explorer aims to act as a single
point of entry for users looking to study single cell ATAC-seq data.
Users can quickly search available datasets using the metadata table and download
datasets of interest for immediate analysis within R.

License Artistic-2.0

Encoding UTF-8

LazyData FALSE

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

VignetteBuilder knitr

Suggests BiocStyle, knitr, rmarkdown, testthat (>= 3.0.0)

Imports methods, Matrix

Depends R (>= 4.1), SingleCellExperiment, BiocFileCache, data.table,
utils, S4Vectors

biocViews SingleCellData, SequencingData, ExpressionData, GEO, Tissue,
Genome, PackageTypeData

BugReports <https://github.com/shooshtarilab/scATACseq/issues>

Config/testthat/edition 3

git_url https://git.bioconductor.org/packages/scATAC.Explorer

git_branch RELEASE_3_16

git_last_commit c77d6b6

git_last_commit_date 2022-11-01

Date/Publication 2023-04-06

Author Arrian Gibson-Khademi [aut, cre],
     Erik Christensen [aut],
     Parisa Shooshtari [aut]

Maintainer Arrian Gibson-Khademi <agibsonk@uwo.ca>

R topics documented:

---

queryATAC   *A function to query scATAC-seq datasets available in this package*

Description

This function allows you to search and subset included scATAC-seq datasets. A named list of
scATAC-seq_data objects matching the provided options will be returned. Some included datasets
are represented using multiple matrices. Each matrix will be a seperate named object within the
list. The returned list is named by matrix allow easy identification of data. If queryATAC is called
without any options it will retrieve all available datasets in sparse matrix format. This should only
be done on machines with a large amount of ram (>64gb) because some datasets are quite large. In
most cases it is recommended to instead filter databases with some criteria.

Usage

```
queryATAC(
  accession = NULL,
  author = NULL,
  journal = NULL,
  year = NULL,
  pmid = NULL,
  sequence_tech = NULL,
  score_type = NULL,
  has_cluster_annotation = NULL,
  has_cell_type_annotation = NULL,
  organism = NULL,
  genome_build = NULL,
  broad_cell_category = NULL,
  tissue_cell_type = NULL,
  disease = NULL,
  metadata_only = FALSE,
  sparse = TRUE
)
```

Arguments

`accession`	Search by geo accession number. Good for returning individual datasets
`author`	Search by the author who published the dataset
`journal`	Search by the journal the dataset was published in.
`year`	Search by exact year or year ranges with '<', '>', or '-'. For example, you can return datasets newer than 2013 with '>2013'
`pmid`	Search by Pubmed ID associated with the study. Good for returning individual datasets
`sequence_tech`	Search by sequencing technology used to sample the cells.
`score_type`	Search by type of score (TPM, FPKM, raw count)
`has_cluster_annotation`	
	Return only those datasets that have clustering results available, or only those without (TRUE/FALSE)
`has_cell_type_annotation`	
	Return only those datasets that have cell-type annotations available, or only those without annotations (TRUE/FALSE)
`organism`	Search by source organism used in the study, for example human or mouse.
`genome_build`	Return datasets built only using specified genome build (ex. hg19)
`broad_cell_category`	
	Return datasets based on broad cell categories (ex. Hematopoetic cells). To view all cell categories available, explore the metadata table
`tissue_cell_type`	
	Return datasets based on tissue or cell types sampled (ex. PBMCs, Bone marrow, Oligodendrocytes)
`disease`	Return datasets based on sampled disease (ex. carcinoma, leukemia, diabetes)
`metadata_only`	Return rows of metadata instead of actual datasets. Useful for exploring what data is available without actually downloading data. Defaults to FALSE
`sparse`	Return expression as a sparse matrix. Reccomended to use sparse format, as dense formats tend to be excessively large.

Value

A list containing a table of metadata or one or more SingleCellExperiment objects

Examples

```
## Retrieve the metadata table to see what data is available
res <- queryATAC(metadata_only = TRUE)

## Retrieve a filtered metadata table that only shows datasets with
## cell type annotations and clustering annotations
res <- queryATAC(has_cluster_annotation = TRUE,
                 has_cell_type_annotation = TRUE,
                 metadata_only = TRUE)

## Retrieve a single dataset identified from the table
res <- queryATAC(accession = "GSE89362")
```

saveATAC   *A function to save a scATAC-seq dataset stored in a SingleCellExperiment*

Description

This function allows you to save the counts, peaks, cell ID's/barcodes, and any cell clustering data to disk in csv format. It takes two options: an object to save and a directory to save in. Multiple files will be created in the provided output directory, one for each type of data available in the scATAC_data object (counts, cell ID/Barcode, peak regions, cell type/cluster annotations).

Usage

```
saveATAC(object, outdir)
```

Arguments

object   The SingleCellExperiment object to be written to disk, this should be an individual dataset returned by queryATAC.

outdir   The directory to save the data in, the directory should not exist yet.

Value

Nothing

Examples

```
# Retrieve a previously identified dataset (see queryATAC) and save it to disk
res <- queryATAC(accession = 'GSE89362')[[1]]

saveATAC(res, output_directory_name)
```

Index