Package ‘TMExplorer’

December 19, 2024

Title A Collection of Tumour Microenvironment Single-cell RNA Sequencing Datasets and Corresponding Metadata

Version 1.17.0

Description This package provides a tool to search and download a collection of tumour microenvironment single-cell RNA sequencing datasets and their metadata. TMExplorer aims to act as a single point of entry for users looking to study the tumour microenvironment at the single cell level. Users can quickly search available datasets using the metadata table and then download the ones they are interested in for analysis.

License Artistic-2.0

Encoding UTF-8

LazyData FALSE

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

VignetteBuilder knitr

Suggests BiocStyle, knitr, rmarkdown

Imports methods, Matrix

Depends R (>= 4.1), SingleCellExperiment, BiocFileCache

biocViews CancerData, SingleCellData, RNASeqData, SequencingData, ExpressionData, GEO, PackageTypeData

BugReports `https://github.com/shooshtarilab/TMExplorer/issues`

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

git_branch devel

git_last_commit 9a6c437

git_last_commit_date 2024-10-29

Repository Bioconductor 3.21

Date/Publication 2024-12-19

Author Erik Christensen [aut, cre],
Alaine Naidas [aut],
David Chen [aut],
Parisa Shooshtari [aut]

Maintainer Erik Christensen <echris3@uwo.ca>

Contents

queryTME *A function to query TME datasets available in this package*

Description

This function allows you to search and subset included TME datasets. A list of tme_data objects matching the provided options will be returned, if queryTME is called without any options it will retrieve all available datasets. 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

```
queryTME(
  geo_accession = NULL,
  score_type = NULL,
  has_signatures = NULL,
  has_truth = NULL,
  tumour_type = NULL,
  author = NULL,
  journal = NULL,
  year = NULL,
  pmid = NULL,
  sequence_tech = NULL,
  organism = NULL,
  metadata_only = FALSE,
  sparse = FALSE
)
```

Arguments

`geo_accession`	Search by geo accession number. Good for returning individual datasets
`score_type`	Search by type of score (TPM, FPKM, raw count)
`has_signatures`	Return only those datasets that have cell-type gene signatures available, or only those without (TRUE/FALSE)
`has_truth`	Return only those datasets that have cell-type annotations available, or only those without annotations

`tumour_type` Search by type of tumour represented by the dataset

`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.

`organism` Search by source organism used in the study, for example human or mouse.

`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. Uses less memory but is less convenient to view, recommended only if encounter memory issues with dense data. Defaults to FALSE.

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 <- queryTME(metadata_only = TRUE)

## Retrieve a filtered metadata table that only shows datasets with
## cell type annotations and cell type gene signatures
res <- queryTME(has_truth = TRUE, has_signatures = TRUE, metadata_only = TRUE)

## Retrieve a single dataset identified from the table
res <- queryTME(geo_accession = "GSE72056")
```

`saveTME` *A function to save a TME dataset*

Description

This function allows you to save the expression, labels, and cell types 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 tme_data object (expression, gene signatures, cell type annotations).

Usage

```
saveTME(object, outdir)
```

Arguments

`object` The tme_data object to be written to disk, this should be an individual dataset returned by queryTME.

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

Value

Nothing

Examples

```
# Retrieve a previously identified dataset (see queryTME) and save it to disk
res <- queryTME(geo_accession = 'GSE72056')[[1]]

saveTME(res, output_directory_name)
```

Index