Package 'celldex'

September 3, 2024

Title Index of Reference Cell Type Datasets

Version 1.15.0

Date 2024-04-25

Description Provides a collection of reference expression datasets with curated cell type labels, for use in procedures like automated annotation of single-cell data or deconvolution of bulk RNA-seq.

License GPL-3

Depends SummarizedExperiment

Imports utils, methods, Matrix, ExperimentHub, AnnotationHub, AnnotationDbi, S4Vectors, DelayedArray, DelayedMatrixStats, gypsum, alabaster.base, alabaster.matrix, alabaster.se, DBI, RSQLite, jsonlite

Suggests testthat, knitr, rmarkdown, BiocStyle, DT, jsonvalidate, BiocManager, ensembldb

biocViews ExperimentHub, ExperimentData, ExpressionData, SequencingData, RNASeqData

VignetteBuilder knitr

Encoding UTF-8

URL https://github.com/LTLA/celldex

BugReports https://support.bioconductor.org/

RoxygenNote 7.3.1

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

git_branch devel

git_last_commit 41cd4a6

git_last_commit_date 2024-04-30

Repository Bioconductor 3.20

Date/Publication 2024-09-03

Author Dvir Aran [aut],
Aaron Lun [aut, cre, cph],
Daniel Bunis [aut],
Jared Andrews [aut],
Friederike Dündar [aut]

Maintainer Aaron Lun <infinite.monkeys.with.keyboards@gmail.com>

Contents

`BlueprintEncodeData` *Obtain human bulk RNA-seq data from Blueprint and ENCODE*

Description

Download and cache the normalized expression values of 259 RNA-seq samples of pure stroma and immune cells as generated and supplied by Blueprint and ENCODE.

Usage

```
BlueprintEncodeData(
  rm.NA = c("rows", "cols", "both", "none"),
  ensembl = FALSE,
  cell.ont = c("all", "nonna", "none"),
  legacy = FALSE
)
```

Arguments

rm.NA	String specifying how missing values should be handled. "rows" will remove genes with at least one missing value, "cols" will remove samples with at least one missing value, "both" will remove any gene or sample with at least one missing value, and "none" will not perform any removal.
ensembl	Logical scalar indicating whether to convert row names to Ensembl IDs. Genes without a mapping to a non-duplicated Ensembl ID are discarded.
cell.ont	String specifying whether Cell Ontology terms should be included in the colData. If "nonna", all samples without a valid term are discarded; if "all", all samples are returned with (possibly NA) terms; if "none", terms are not added.
legacy	Logical scalar indicating whether to pull data from ExperimentHub. By default, we use data from the gypsum backend.

Details

This function provides normalized expression values for 259 bulk RNA-seq samples generated by Blueprint and ENCODE from pure populations of stroma and immune cells (Martens and Stunnenberg, 2013; The ENCODE Consortium, 2012). The samples were processed and normalized as described in Aran, Looney and Liu et al. (2019), i.e., the raw RNA-seq counts were downloaded from Blueprint and ENCODE in 2016 and normalized via edgeR (TPMs).

Blueprint Epigenomics contains 144 RNA-seq pure immune samples annotated to 28 cell types. ENCODE contains 115 RNA-seq pure stroma and immune samples annotated to 17 cell types. All together, this reference contains 259 samples with 43 cell types ("label.fine"), manually aggregated into 24 broad classes ("label.main"). The fine labels have also been mapped to the Cell Ontology ("label.ont", if cell.ont is not "none"), which can be used for further programmatic queries.

Value

A SummarizedExperiment object with a "logcounts" assay containing the log-normalized expression values, along with cell type labels in the colData.

Author(s)

Friederike Dündar

References

The ENCODE Project Consortium (2012). An integrated encyclopedia of DNA elements in the human genome. *Nature* 489, pages 57–74.

Martens JHA and Stunnenberg HG (2013). BLUEPRINT: mapping human blood cell epigenomes. *Haematologica* 98, 1487–1489.

Aran D, Looney AP, Liu L et al. (2019). Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat. Immunol.* 20, 163–172.

Examples

```
ref.se <- BlueprintEncodeData(rm.NA = "rows")
```

`DatabaseImmuneCellExpressionData` *Obtain human bulk RNA-seq data from DICE*

Description

Download and cache the normalized expression values of 1561 bulk RNA-seq samples of sorted cell populations from the Database of Immune Cell Expression (DICE).

Usage

```
DatabaseImmuneCellExpressionData(
  ensembl = FALSE,
  cell.ont = c("all", "nonna", "none"),
  legacy = FALSE
)
```

Arguments

`ensembl`	Logical scalar indicating whether to convert row names to Ensembl IDs. Genes without a mapping to a non-duplicated Ensembl ID are discarded.
`cell.ont`	String specifying whether Cell Ontology terms should be included in the `colData`. If `"nonna"`, all samples without a valid term are discarded; if `"all"`, all samples are returned with (possibly `NA`) terms; if `"none"`, terms are not added.
`legacy`	Logical scalar indicating whether to pull data from ExperimentHub. By default, we use data from the gypsum backend.

Details

This function provides normalized expression values of 1561 bulk RNA-seq samples generated by DICE from pure populations of human immune cells.

TPM normalized values for each cell type were downloaded from https://dice-database.org/downloads. Genes with no reads across samples were removed, and values were log2 normalized after a pseudocount of 1 was added.

The dataset contains 1561 human RNA-seq samples annotated to 5 main cell types (`"label.main"`):

- B cells
- Monocytes
- NK cells
- T cells, CD8+
- T cells, CD4+

Samples were additionally annotated to 15 fine cell types (`"label.fine"`):

- B cells, naive
- Monocytes, CD14+
- Monocytes, CD16+
- NK cells
- T cells, memory TREG
- T cells, CD4+, naive
- T cells, CD4+, naive, stimulated
- T cells, CD4+, naive Treg
- T cells, CD4+, Th1
- T cells, CD4+, Th1_17
- T cells, CD4+, Th2
- T cells, CD8+, naïve
- T cells, CD8+, naïve, stimulated
- T cells, CD4+, TFH
- T cells, CD4+, Th17

The subtypes have also been mapped to the Cell Ontology (`"label.ont"`, if `cell.ont` is not `"none"`), which can be used for further programmatic queries.

Value

A SummarizedExperiment object with a `"logcounts"` assay containing the log-normalized expression values, along with cell type labels in the `colData`.

Author(s)

Jared Andrews

References

Schmiedel B et al. (2018). Impact of Genetic Polymorphisms on Human Immune Cell Gene Expression. *Cell* 175, 1701-1715.

Examples

```
ref.se <- DatabaseImmuneCellExpressionData()
```

`fetchReference` *Fetch a reference dataset*

Description

Fetch a reference dataset (or its metadata) from the gypsum backend.

Usage

```
fetchReference(
  name,
  version,
  path = NA,
  package = "celldex",
  cache = cacheDirectory(),
  overwrite = FALSE,
  realize.assays = FALSE,
  ...
)

fetchMetadata(
  name,
  version,
  path = NA,
  package = "celldex",
  cache = cacheDirectory(),
  overwrite = FALSE
)
```

Arguments

`name`	String containing the name of the reference dataset.
`version`	String containing the version of the dataset.
`path`	String containing the path to a subdataset, if `name` contains multiple reference datasets. Defaults to `NA` if no subdatasets are present.
`package`	String containing the name of the package.
`cache, overwrite`	Arguments to pass to `saveVersion` or `saveFile`.
`realize.assays`	Logical scalar indicating whether to realize assays into memory. Dense and sparse ReloadedArray objects are converted into ordinary arrays and dgCMatrix objects, respectively.
`...`	Further arguments to pass to `readObject`.

Value

`fetchReference` returns the dataset as a SummarizedExperiment. This is guaranteed to have a `"logcounts"` assay with log-normalized expression values, along with at least one character vector of labels in the column data.

`fetchMetadata` returns a named list of metadata for the specified dataset.

Author(s)

Aaron Lun

See Also

https://github.com/ArtifactDB/bioconductor-metadata-index, on the expected schema for the metadata.

`saveReference` and `uploadDirectory`, to save and upload a dataset.

`listReferences` and `listVersions`, to get possible values for `name` and `version`.

Examples

```
fetchReference("immgen", "2024-02-26")
str(fetchMetadata("immgen", "2024-02-26"))
```

HumanPrimaryCellAtlasData

Obtain the HPCA data

Description

Download and cache the normalized expression values of the data stored in the Human Primary Cell Atlas. The data will be downloaded from ExperimentHub, returning a SummarizedExperiment object for further use.

Usage

```
HumanPrimaryCellAtlasData(
  ensembl = FALSE,
  cell.ont = c("all", "nonna", "none"),
  legacy = FALSE
)
```

Arguments

`ensembl`	Logical scalar indicating whether to convert row names to Ensembl IDs. Genes without a mapping to a non-duplicated Ensembl ID are discarded.
`cell.ont`	String specifying whether Cell Ontology terms should be included in the `colData`. If `"nonna"`, all samples without a valid term are discarded; if `"all"`, all samples are returned with (possibly `NA`) terms; if `"none"`, terms are not added.
`legacy`	Logical scalar indicating whether to pull data from ExperimentHub. By default, we use data from the gypsum backend.

Details

This function provides normalized expression values for 713 microarray samples from the Human Primary Cell Atlas (HPCA) (Mabbott et al., 2013). These 713 samples were processed and normalized as described in Aran, Looney and Liu et al. (2019).

Each sample has been assigned to one of 37 main cell types (`"label.main"`) and 157 subtypes (`"label.fine"`). The subtypes have also been mapped to the Cell Ontology (`"label.ont"`, if `cell.ont` is not `"none"`), which can be used for further programmatic queries.

Value

A SummarizedExperiment object with a `"logcounts"` assay containing the log-normalized expression values, along with cell type labels in the `colData`.

Author(s)

Friederike Dündar

References

Mabbott NA et al. (2013). An expression atlas of human primary cells: inference of gene function from coexpression networks. *BMC Genomics* 14, Article 632.

Aran D, Looney AP, Liu L et al. (2019). Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat. Immunol.* 20, 163–172.

Examples

```
ref.se <- HumanPrimaryCellAtlasData()
```

`ImmGenData` *Obtain mouse bulk expression data from the Immunologic Genome Project*

Description

Download and cache the normalized expression values of 830 microarray samples of pure mouse immune cells, generated by the Immunologic Genome Project (ImmGen).

Usage

```
ImmGenData(
  ensembl = FALSE,
  cell.ont = c("all", "nonna", "none"),
  legacy = FALSE
)
```

Arguments

`ensembl`	Logical scalar indicating whether to convert row names to Ensembl IDs. Genes without a mapping to a non-duplicated Ensembl ID are discarded.
`cell.ont`	String specifying whether Cell Ontology terms should be included in the `colData`. If `"nonna"`, all samples without a valid term are discarded; if `"all"`, all samples are returned with (possibly `NA`) terms; if `"none"`, terms are not added.
`legacy`	Logical scalar indicating whether to pull data from ExperimentHub. By default, we use data from the gypsum backend.

Details

This function provides normalized expression values of 830 microarray samples generated by ImmGen from pure populations of murine immune cells (<http://www.immgen.org/>). The samples were processed and normalized as described in Aran, Looney and Liu et al. (2019), i.e., CEL files from the Gene Expression Omnibus (GEO; GSE15907 and GSE37448), were downloaded, processed, and normalized using the robust multi-array average (RMA) procedure on probe-level data.

This dataset consists of 20 broad cell types (`"label.main"`) and 253 finely resolved cell subtypes (`"label.fine"`). The subtypes have also been mapped to the Cell Ontology (`"label.ont"`, if `cell.ont` is not `"none"`), which can be used for further programmatic queries.

Value

A SummarizedExperiment object with a `"logcounts"` assay containing the log-normalized expression values, along with cell type labels in the `colData`.

Author(s)

Friederike Dündar

References

Heng TS, Painter MW, Immunological Genome Project Consortium (2008). The Immunological Genome Project: networks of gene expression in immune cells. *Nat. Immunol.* 9, 1091-1094.

Aran D, Looney AP, Liu L et al. (2019). Reference-based analysis of lung single-cell sequencing reveals a transitional profibrotic macrophage. *Nat. Immunol.* 20, 163–172.

Examples

```
ref.se <- ImmGenData()
```

---

listReferences *List available references*

Description

List the available reference datasets and the associated versions in **celldex**.

Usage

```
listReferences()

listVersions(name)

fetchLatestVersion(name)
```

Arguments

name	String containing the name of the reference dataset.

Value

For `listReferences`, a character vector containing the names of the available references.

For `listVersions`, a character vector containing the names of the available versions of the `name` reference.

For `fetchLatestVersion`, a string containing the name of the latest version.

Author(s)

Aaron Lun

Examples

```
listReferences()
listVersions("immgen")
fetchLatestVersion("immgen")
```

MonacoImmuneData *Obtain bulk RNA-seq data of sorted human immune cells*

Description

Download and cache the normalized expression values of 114 bulk RNA-seq samples of sorted immune cell populations that can be found in GSE107011.

Usage

```
MonacoImmuneData(
  ensembl = FALSE,
  cell.ont = c("all", "nonna", "none"),
  legacy = FALSE
)
```

Arguments

ensembl	Logical scalar indicating whether to convert row names to Ensembl IDs. Genes without a mapping to a non-duplicated Ensembl ID are discarded.
cell.ont	String specifying whether Cell Ontology terms should be included in the colData. If "nonna", all samples without a valid term are discarded; if "all", all samples are returned with (possibly NA) terms; if "none", terms are not added.
legacy	Logical scalar indicating whether to pull data from ExperimentHub. By default, we use data from the gypsum backend.

Details

The dataset contains 114 human RNA-seq samples annotated to 10 main cell types ("label.main"):

- CD8+ T cells
- T cells
- CD4+ T cells
- Progenitors
- B cells
- Monocytes
- NK cells
- Dendritic cells
- Neutrophils
- Basophils

Samples were additionally annotated to 29 fine cell types ("label.fine"):

- Naive CD8 T cells

- Central memory CD8 T cells
- Effector memory CD8 T cells
- Terminal effector CD8 T cells
- MAIT cells
- Vd2 gd T cells
- Non-Vd2 gd T cells
- Follicular helper T cells
- T regulatory cells
- Th1 cells
- Th1/Th17 cells
- Th17 cells
- Th2 cells
- Naive CD4 T cells
- Terminal effector CD4 T cells
- Progenitor cells
- Naive B cells
- Non-switched memory B cells
- Exhausted B cells
- Switched memory B cells
- Plasmablasts
- Classical monocytes
- Intermediate monocytes
- Non classical monocytes
- Natural killer cells
- Plasmacytoid dendritic cells
- Myeloid dendritic cells
- Low-density neutrophils
- Low-density basophils

The subtypes have also been mapped to the Cell Ontology (`"label.ont"`, if `cell.ont` is not `"none"`), which can be used for further programmatic queries.

Value

A SummarizedExperiment object with a `"logcounts"` assay containing the log-normalized expression values, along with cell type labels in the `colData`.

Author(s)

Jared Andrews

References

Monaco G et al. (2019). RNA-Seq Signatures Normalized by mRNA Abundance Allow Absolute Deconvolution of Human Immune Cell Types *Cell Rep.* 26, 1627-1640.

Examples

```
ref.se <- MonacoImmuneData()
```

`MouseRNAseqData` *Obtain mouse bulk expression data of sorted cell populations (RNA-seq)*

Description

Download and cache the normalized expression values of 358 bulk RNA-seq samples of sorted cell populations that can be found at GEO.

Usage

```
MouseRNAseqData(
  ensembl = FALSE,
  cell.ont = c("all", "nonna", "none"),
  legacy = FALSE
)
```

Arguments

`ensembl`	Logical scalar indicating whether to convert row names to Ensembl IDs. Genes without a mapping to a non-duplicated Ensembl ID are discarded.
`cell.ont`	String specifying whether Cell Ontology terms should be included in the `colData`. If `"nonna"`, all samples without a valid term are discarded; if `"all"`, all samples are returned with (possibly `NA`) terms; if `"none"`, terms are not added.
`legacy`	Logical scalar indicating whether to pull data from ExperimentHub. By default, we use data from the gypsum backend.

Details

This dataset was contributed by the Benayoun Lab that identified, downloaded and processed data sets on GEO that corresponded to sorted cell types (Benayoun et al., 2019).

The dataset contains 358 mouse RNA-seq samples annotated to 18 main cell types (`"label.main"`):

- Adipocytes
- Astrocytes
- B cells
- Cardiomyocytes

- Dendritic cells
- Endothelial cells
- Epithelial cells
- Erythrocytes
- Fibroblasts
- Granulocytes
- Hepatocytes
- Macrophages
- Microglia
- Monocytes
- Neurons
- NK cells
- Oligodendrocytes
- T cells

These are split further into 28 subtypes (`"label.fine"`). The subtypes have also been mapped to the Cell Ontology (`"label.ont"`, if `cell.ont` is not `"none"`), which can be used for further programmatic queries.

Value

A SummarizedExperiment object with a `"logcounts"` assay containing the log-normalized expression values, along with cell type labels in the `colData`.

Author(s)

Friederike Dündar

References

Benayoun B et al. (2019). Remodeling of epigenome and transcriptome landscapes with aging in mice reveals widespread induction of inflammatory responses. *Genome Res.* 29, 697-709.

Code at https://github.com/BenayounLaboratory/Mouse_Aging_Epigenomics_2018/tree/master/FigureS7_CIBERSORT/RNAseq_datasets_for_Deconvolution/2017-01-18

Examples

```
ref.se <- MouseRNAseqData()
```

NovershternHematopoieticData *Obtain bulk microarray expression for sorted hematopoietic cells*

Description

Download and cache the normalized expression values of 211 bulk human microarray samples of sorted hematopoietic cell populations that can be found in GSE24759.

Usage

```
NovershternHematopoieticData(
  ensembl = FALSE,
  cell.ont = c("all", "nonna", "none"),
  legacy = FALSE
)
```

Arguments

ensembl	Logical scalar indicating whether to convert row names to Ensembl IDs. Genes without a mapping to a non-duplicated Ensembl ID are discarded.
cell.ont	String specifying whether Cell Ontology terms should be included in the colData. If "nonna", all samples without a valid term are discarded; if "all", all samples are returned with (possibly NA) terms; if "none", terms are not added.
legacy	Logical scalar indicating whether to pull data from ExperimentHub. By default, we use data from the gypsum backend.

Details

The dataset contains 211 human microarray samples annotated to 16 main cell types ("label.main"):

- Basophils
- B cells
- CMPs
- Dendritic cells
- Eosinophils
- Erythroid cells
- GMPS
- Granulocytes
- HSCs
- Megakaryocytes
- MEPs
- Monocytes

- NK cells
- NK T cells
- CD8+ T cells
- CD4+ T cells

Samples were additionally annotated to 38 fine cell types (`"label.fine"`):

- Basophils
- Naive B cells
- Mature B cells class able to switch
- Mature B cells
- Mature B cells class switched
- Common myeloid progenitors
- Plasmacytoid Dendritic Cells
- Myeloid Dendritic Cells
- Eosinophils
- Erythroid_CD34+ CD71+ GlyA-
- Erythroid_CD34- CD71+ GlyA-
- Erythroid_CD34- CD71+ GlyA+
- Erythroid_CD34- CD71lo GlyA+
- Erythroid_CD34- CD71- GlyA+
- Granulocyte/monocyte progenitors
- Colony Forming Unit-Granulocytes
- Granulocyte (Neutrophilic Metamyelocytes)
- Granulocyte (Neutrophils)
- Hematopoietic stem cells_CD133+ CD34dim
- Hematopoietic stem cell_CD38- CD34+
- Colony Forming Unit-Megakaryocytic
- Megakaryocytes
- Megakaryocyte/erythroid progenitors
- Colony Forming Unit-Monocytes
- Monocytes
- Mature NK cells_CD56- CD16+ CD3-
- Mature NK cells_CD56+ CD16+ CD3-
- Mature NK cells_CD56- CD16- CD3-
- NK T cells
- Early B cells
- Pro B cells
- CD8+ Effector Memory RA

- Naive CD8+ T cells
- CD8+ Effector Memory
- CD8+ Central Memory
- Naive CD4+ T cells
- CD4+ Effector Memory
- CD4+ Central Memory

The subtypes have also been mapped to the Cell Ontology (`"label.ont"`, if `cell.ont` is not `"none"`), which can be used for further programmatic queries.

Value

A SummarizedExperiment object with a `"logcounts"` assay containing the log-normalized expression values, along with cell type labels in the `colData`.

Author(s)

Jared Andrews

References

Novershtern N et al. (2011). Densely interconnected transcriptional circuits control cell states in human hematopoiesis. *Cell* 144, 296-309.

Examples

```
ref.se <- NovershternHematopoieticData()
```

`reexports` *Objects exported from other packages*

Description

These objects are imported from other packages. Follow the links below to see their documentation.

gypsum `defineTextQuery`

saveReference *Save a reference dataset*

Description

Save a reference dataset to disk, usually in preparation for upload via uploadDirectory.

Usage

```
saveReference(x, labels, path, metadata)
```

Arguments

x	Matrix of log-normalized expression values. This may be sparse or dense, but should be non-integer and have no missing values. Row names should be present and unique for all rows.
labels	DataFrame of labels. Each row of labels corresponds to a column of x and contains the label(s) for that column. Each column of labels represents a different label type; typically, the column name has a label. prefix to distinguish between, e.g., label.fine, label.broad and so on. At least one column should be present.
path	String containing the path to a directory in which to save x.
metadata	Named list containing metadata for this reference dataset, see the schema returned by fetchMetadataSchema(). Note that the applications.takane property will be automatically added by this function and does not have to be supplied.

Details

The SummarizedExperiment saved to path is guaranteed to have the "logcounts" assay and at least one column in labels. This mirrors the expectation for reference datasets obtained via fetchReference.

Value

x and labels are used to create a SummarizedExperiment that is saved into path. NULL is invisibly returned.

Author(s)

Aaron Lun

See Also

uploadDirectory, to upload the saved dataset to the gypsum backend.

fetchReference, to download an existing dataset into the current sesssion.

Examples

```
# Mocking up some data to be saved.
x <- matrix(rnorm(1000), ncol=10)
rownames(x) <- sprintf("GENE_%i", seq_len(nrow(x)))
labels <- DataFrame(
    labels.broad = sample(c("B", "T", "NK"), ncol(x), replace=TRUE),
    labels.fine = sample(c("PC", "pre-B", "pro-B", "Th2", "CD4+T", "NK"),
        ncol(x), replace=TRUE)
)

# Making up some metadata as well.
meta <- list(
    title="New reference dataset",
    description="This is a new reference dataset, generated from blah blah.",
    genome="GRCm38",
    taxonomy_id="10090",
    sources=list(
        list(provider="GEO", id="GSE123456"),
        list(provider="PubMed", id="123456"),
        list(provider="URL", id="https://reference.data.com", version="2024-02-26")
    ),
    maintainer_name="Chihaya Kisaragi",
    maintainer_email="kisaragi.chihaya@765.com"
)

# Actually saving it.
tmp <- tempfile()
saveReference(x, labels, tmp, meta)

# Reloading it to make sure it looks good.
alabaster.base::readObject(tmp)
str(jsonlite::fromJSON(file.path(tmp, "_bioconductor.json")))
```

---

searchReferences *Search reference metadata*

Description

Search for reference datasets of interest based on matching text in the associated metadata.

Usage

```
searchReferences(
  query,
  cache = cacheDirectory(),
  overwrite = FALSE,
  latest = TRUE
)
```

Arguments

query	String or a gypsum.search.object, see Examples.
cache, overwrite	Arguments to pass to fetchMetadataDatabase.
latest	Whether to only consider the latest version of each dataset.

Details

The returned DataFrame contains the usual suspects like the title and description for each dataset, the number of rows and columns, the organisms and genome builds involved, whether the dataset has any pre-computed reduced dimensions, and so on. More details can be found in the Bioconductor metadata schema at https://github.com/ArtifactDB/bioconductor-metadata-index.

Value

A DataFrame where each row corresponds to a dataset, containing various columns of metadata. Some columns may be lists to capture 1:many mappings.

Author(s)

Aaron Lun

See Also

surveyReferences, to easily obtain a listing of all available datasets.

Examples

```
searchReferences(defineTextQuery("immun%", partial=TRUE))[,c("name", "title")]
searchReferences(defineTextQuery("10090", field="taxonomy_id"))[,c("name", "title")]
searchReferences(
   defineTextQuery("10090", field="taxonomy_id") &
   defineTextQuery("immun%", partial=TRUE)
)[,c("name", "title")]
```

---

surveyReferences *Survey reference metadata*

Description

Metadata survey for all available reference datasets in the **celldex** package.

Usage

```
surveyReferences(cache = cacheDirectory(), overwrite = FALSE, latest = TRUE)
```

Arguments

cache, overwrite	Arguments to pass to fetchMetadataDatabase.
latest	Whether to only consider the latest version of each dataset.

Details

The returned DataFrame contains the usual suspects like the title and description for each dataset, the number of samples and types of labels, the organisms and genome builds involved, and so on. More details can be found in the Bioconductor metadata schema at https://github.com/ArtifactDB/bioconductor-metadata-index.

Value

A DataFrame where each row corresponds to a dataset, containing various columns of metadata. Some columns may be lists to capture 1:many mappings.

Author(s)

Aaron Lun

See Also

searchReferences, to search on the metadata for specific datasets.

Examples

```
surveyReferences()
```

Index