Package ‘celldex’

August 8, 2024

Title Index of Reference Cell Type Datasets

Version 1.14.0

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 RELEASE_3_19

git_last_commit 7c1768e

git_last_commit_date 2024-04-30

Repository Bioconductor 3.19

Date/Publication 2024-08-08
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


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+

Examples

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


Examples

```r
ref.se <- DatabaseImmuneCellExpressionData()
```
fetchReference  
*Fetch a reference dataset*

**Description**

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

**Usage**

```r
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`. 
Function

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

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


Examples

```r
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**

```r
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


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")
```
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

```r
MonacoImmuneData(
  ensemb1 = FALSE,
  cell.ont = c("all", "nonna", "none"),
  legacy = FALSE
)
```

Arguments

- `ensemb1`: 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
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
MouseRNAseqData

References

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

ensemb1 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
MouseRNAseqData

- 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


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()
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


Examples

ref.se <- NovershternHematopoieticData()
saveReference  

**Save a reference dataset**

**Description**

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

**Usage**

```r
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 session.
Examples

```r
# 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**

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

**Usage**

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

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](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

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

---

**Description**

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

**Usage**

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

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](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

* **internal**
  reexports, 17

BlueprintEncodeData, 2

colData, 3–5, 8, 9, 11–15, 17

DatabaseImmuneCellExpressionData, 4

DataFrame, 18, 20, 21

defineTextQuery, 17

defineTextQuery(reexports), 17

dgCMatrix, 6

fetchLatestVersion(listReferences), 10

fetchMetadata(fetchReference), 6

fetchMetadataDatabase, 20, 21

fetchMetadataSchema, 18

fetchReference, 6, 18

HumanPrimaryCellAtlasData, 7

ImmGenData, 9

listReferences, 7, 10

listVersions, 7

listVersions(listReferences), 10

MonacoImmuneData, 11

MouseRNAseqData, 13

NovershternHematopoieticData, 15

readObject, 6

reexports, 17

ReloadedArray, 6

saveFile, 6

saveReference, 7, 18

saveVersion, 6

searchReferences, 19, 21

SummarizedExperiment, 3, 5, 7–9, 12, 14, 17, 18

surveyReferences, 20, 20

uploadDirectory, 7, 18