Package 'imcdatasets'

November 12, 2026

Version 1.21.3

Date 2026-07-22

Title Collection of publicly available imaging mass cytometry (IMC) datasets

Description The imcdatasets package provides access to publicly available IMC datasets. IMC is a technology that enables measurement of > 40 proteins from tissue sections. The generated images can be segmented to extract single cell data. Datasets typically consist of three elements: a SingleCellExperiment object containing single cell data, a CytoImageList object containing multichannel images and a CytoImageList object containing the cell masks that were used to extract the single cell data from the images.

License GPL-3 + file LICENSE

NeedsCompilation no

Depends R (>= 4.6.0), SingleCellExperiment, SpatialExperiment, cytomapper,

Imports methods, utils, ExperimentHub, S4Vectors, DelayedArray, HDF5Array

Suggests BiocStyle, knitr, rmarkdown, markdown, testthat

biocViews ExperimentData, ExperimentHub, SingleCellData, SpatialData, Homo_sapiens_Data, ImmunoOncologyData, TechnologyData, PackageTypeData, ReproducibleResearch, Tissue

VignetteBuilder knitr

URL https://github.com/BodenmillerGroup/imcdatasets, https://bodenmillergroup.github.io/imcdatasets/

BugReports https://github.com/BodenmillerGroup/imcdatasets/issues

BiocType ExperimentData

Encoding UTF-8

Config/roxygen2/version 8.0.0

Config/Bioconductor/UnsupportedPlatforms windows

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

git_branch devel

git_last_commit 4c55227

git_last_commit_date 2026-08-03

Repository Bioconductor 3.24

Date/Publication 2026-11-12

Author Nicolas Damond [aut] (ORCID: <https://orcid.org/0000-0003-3027-8989>),
Nathan Steenbuck [aut, cre] (ORCID:
<https://orcid.org/0000-0003-2712-5359>),
Nils Eling [ctb] (ORCID: <https://orcid.org/0000-0002-4711-1176>),
Fischer Jana [ctb] (ORCID: <https://orcid.org/0000-0001-9047-8400>)

Maintainer Nathan Steenbuck <nathan.steenbuck@uzh.ch>

Contents

Damond_2019_Pancreas 2
HochSchulz_2022_Melanoma 5
IMMUcan_2022_CancerExample 8
JacksonFischer_2020_BreastCancer 11
listDatasets 14
Meyer_2025_TripleNegativeBreastCancer 14
SteenbuckDamond_2026_T1D 17
Zanotelli_2020_Spheroids 20

Index **24**

`Damond_2019_Pancreas` *Obtain the Damond_2019_Pancreas dataset*

Description

Obtain the Damond_2019_Pancreas dataset, which consists of three data objects: single cell data, multichannel images and cell segmentation masks. The data was obtained by imaging mass cytometry (IMC) of human pancreas sections from donors with type 1 diabetes.

Usage

```
Damond_2019_Pancreas(
  data_type = c("sce", "spe", "images", "masks"),
  full_dataset = FALSE,
  version = "latest",
  metadata = FALSE,
  on_disk = FALSE,
  h5FilesPath = NULL,
  force = FALSE
)
```

Arguments

`data_type`	type of object to load, ‘images‘ for multichannel images or ‘masks‘ for cell segmentation masks. Single cell data are retrieved using either ‘sce‘ for the `SingleCellExperiment` format or ‘spe‘ for the `SpatialExperiment` format.

`full_dataset`	if FALSE (default), a subset corresponding to 100 images is returned. If TRUE, the full dataset (corresponding to 845 images) is returned. Due to memory space limitations, this option is only available for single cell data and masks, not for `data_type = "images"`.
`version`	dataset version. By default, the latest version is returned.
`metadata`	if FALSE (default), the data object selected in `data_type` is returned. If TRUE, only the metadata associated to this object is returned.
`on_disk`	logical indicating if images in form of HDF5Array objects (as .h5 files) should be stored on disk rather than in memory. This setting is valid when downloading `images` and `masks`.
`h5FilesPath`	path to where the .h5 files for on disk representation are stored. This path needs to be defined when `on_disk = TRUE`. When files should only temporarily be stored on disk, please set `h5FilesPath = getHDF5DumpDir()`.
`force`	logical indicating if images should be overwritten when files with the same name already exist on disk.

Details

This is an Imaging Mass Cytometry (IMC) dataset from Damond et al. (2019):

- `images` contains a hundred 38-channel images in the form of a CytoImageList class object.
- `masks` contains the cell segmentation masks associated with the images, in the form of a CytoImageList class object.
- `sce` contains the single cell data extracted from the multichannel images using the cell segmentation masks, as well as the associated metadata, in the form of a SingleCellExperiment. This represents a total of 252,059 cells x 38 channels.
- `spe` same single cell data as for `sce`, but in the SpatialExperiment format.

All data are downloaded from ExperimentHub and cached for local re-use.

Mapping between the three data objects is performed via variables located in their metadata columns: `mcols()` for the CytoImageList objects and `ColData()` for the SingleCellExperiment and SpatialExperiment objects. Mapping at the image level can be performed with the `image_name` or `image_number` variables. Mapping between cell segmentation masks and single cell data is performed with the `cell_number` variable, the values of which correspond to the intensity values of the `masks` object. For practical examples, please refer to the "Accessing IMC datasets" vignette.

This dataset is a subset of the complete Damond et al. (2019) dataset comprising the data from three pancreas donors at different stages of type 1 diabetes (T1D). The three donors present clearly diverging characteristics in terms of cell type composition and cell-cell interactions, which makes this dataset ideal for benchmarking spatial and neighborhood analysis algorithms. If `full_dataset = TRUE`, the full dataset (845 images from 12 patients) is returned. This option is not available for multichannel images.

The `assay` slots of the SingleCellExperiment and SpatialExperiment objects contain three assays:

- `counts` contains raw mean ion counts per cell.
- `exprs` contains arsinh-transformed counts, with cofactor 1.
- `quant_norm` contains counts censored at the 99th percentile and scaled 0-1.

The marker-associated metadata, including antibody information and metal tags are stored in the `rowData` of the SingleCellExperiment / SpatialExperiment objects.

The cell-associated metadata are stored in the `colData` of the SingleCellExperiment and SpatialExperiment objects. These metadata include cell types (in `colData(sce)$cell_type`) and broader cell categories, such as "immune" or "islet" cells (in `colData(sce)$cell_category`). In addition, for cells located inside pancreatic islets, the islet they belong to is indicated in `colData(sce)$islet_parent`. For cells not located in islets, the "islet_parent" value is set to 0 but the spatially closest islet can be identified with `colData(sce)$islet_closest`.

The donor-associated metadata are also stored in the `colData` of the SingleCellExperiment and SpatialExperiment objects. For instance, the donors' IDs can be retrieved with `colData(sce)$patient_id` and the donors' disease stage can be obtained with `colData(sce)$patient_stage`.

Neighborhood information, defined here as cells that are localized next to each other, is stored as a `SelfHits` object in the `colPairs` slot of the `SingleCellExperiment` and SpatialExperiment objects.

The three donors in the subset present the following characteristics:

- `6126` is a non-diabetic donor, with large islets containing many beta cells, severe infiltration of the exocrine pancreas with myeloid cells but limited infiltration of islets.
- `6414` is a donor with recent T1D onset (shortly after diagnosis) showing partial beta cell destruction and mild infiltration of islets with T cells.
- `6180` is a donor with long-duration T1D (11 years after diagnosis), showing near-total beta cell destruction and limited immune cell infiltration in both the islets and the pancreas.

For information about other donors in the full dataset, please refer to the Damond et al. publication.

Dataset versions: a `version` argument can be passed to the function to specify which dataset version should be retrieved.

- `` `v0` ``: original version (Bioconductor <= 3.15).
- `` `v1` ``: consistent object formatting across datasets.

File sizes:

- `` `images` ``: size in memory = 7.4 Gb, size on disk = 1.7 Gb.
- `` `masks` ``: size in memory = 200 Mb, size on disk = 8.2 Mb.
- `` `sce` ``: size in memory = 353 Mb, size on disk = 204 Mb.
- `` `spe` ``: size in memory = 372 Mb, size on disk = 205 Mb.
- `` `sce_full` ``: size in memory = 2.4 Gb, size on disk = 1.5 Gb.
- `` `spe_full` ``: size in memory = 2.5 Gb, size on disk = 1.5 Gb.
- `` `masks_full` ``: size in memory = 1.4 Gb, size on disk = 60 Mb.

When storing images on disk, these need to be first fully read into memory before writing them to disk. This means the process of downloading the data is slower than directly keeping them in memory. However, downstream analysis will lose its memory overhead when storing images on disk.

Original source: Damond et al. (2019): https://doi.org/10.1016/j.cmet.2018.11.014

Original link to raw data, also containing the entire dataset: https://data.mendeley.com/datasets/cydmwsfztj/2

Value

A SingleCellExperiment object with single cell data, a SpatialExperiment object with single cell data, a CytoImageList object containing multichannel images, or a CytoImageList object containing cell segmentation masks.

Author(s)

Nicolas Damond

References

Damond N et al. (2019). A Map of Human Type 1 Diabetes Progression by Imaging Mass Cytometry. *Cell Metab* 29(3), 755-768.

Examples

```
# Load single cell data
sce <- Damond_2019_Pancreas(data_type = "sce")
print(sce)

# Display metadata
Damond_2019_Pancreas(data_type = "sce", metadata = TRUE)

# Load masks on disk
library(HDF5Array)
masks <- Damond_2019_Pancreas(data_type = "masks", on_disk = TRUE,
h5FilesPath = getHDF5DumpDir())
print(head(masks))
```

`HochSchulz_2022_Melanoma` *Obtain the HochSchulz_2022_Melanoma dataset*

Description

Obtain the HochSchulz_2022_Melanoma dataset, which is composed of two panels (rna and protein) that were acquired on consecutive sections. Each dataset (panel) is composed of three data objects: single cell data, multichannel images and cell segmentation masks. The data was obtained by imaging mass cytometry (IMC) of a tissue microarray (TMA) with multiple cores of formalin-fixed paraffin-embedded (FFPE) tissue from 69 patients with metastatic melanoma.

Usage

```
HochSchulz_2022_Melanoma(
  data_type = c("sce", "spe", "images", "masks"),
  panel = "rna",
  full_dataset = FALSE,
  version = "latest",
  metadata = FALSE,
  on_disk = FALSE,
  h5FilesPath = NULL,
  force = FALSE
)
```

Arguments

data_type	type of object to load, 'images' for multichannel images or 'masks' for cell segmentation masks. Single cell data are retrieved using either 'sce' for the SingleCellExperiment format or 'spe' for the SpatialExperiment format.
panel	which panel should be returned? Can be set to "rna" (default) or "protein".
full_dataset	if FALSE (default), a subset corresponding to the 50 images containing the most B cells is returned. If TRUE, the full dataset (corresponding to 166 images) is returned. Due to memory space limitations, this option is only available for single cell data and masks, not for data_type = "images".
version	dataset version. By default, the latest version is returned.
metadata	if FALSE (default), the data object selected in data_type is returned. If TRUE, only the metadata associated to this object is returned.
on_disk	logical indicating if images in form of HDF5Array objects (as .h5 files) should be stored on disk rather than in memory. This setting is valid when downloading images and masks.
h5FilesPath	path to where the .h5 files for on disk representation are stored. This path needs to be defined when on_disk = TRUE. When files should only temporarily be stored on disk, please set h5FilesPath = getHDF5DumpDir().
force	logical indicating if images should be overwritten when files with the same name already exist on disk.

Details

This is an Imaging Mass Cytometry (IMC) dataset from Hoch, Schulz et al. (2022):

- images contains fifty 38-channel images in the form of a CytoImageList class object.
- masks contains the cell segmentation masks associated with the images, in the form of a CytoImageList class object.
- sce contains the single cell data extracted from the multichannel images using the cell segmentation masks, as well as the associated metadata, in the form of a SingleCellExperiment object.
- spe same single cell data as for sce, but in the SpatialExperiment format.

All data are downloaded from ExperimentHub and cached for local re-use.

Mapping between the three data objects is performed via variables located in their metadata columns: mcols() for the CytoImageList objects and ColData() for the SingleCellExperiment and SpatialExperiment objects. Mapping at the image level can be performed with the image_name or image_number variables. Mapping between cell segmentation masks and single cell data is performed with the cell_number variable, the values of which correspond to the intensity values of the masks object. For practical examples, please refer to the "Accessing IMC datasets" vignette.

The assay slots of the SingleCellExperiment and SpatialExperiment objects contain three assays:

- counts contains raw mean ion counts per cell.
- exprs contains arsinh-transformed counts, with cofactor 1.
- scaled_counts contains scaled counts.
- scaled_exprs contains scaled asinh-transformed counts.

The marker-associated metadata, including antibody information and metal tags are stored in the `rowData` of the SingleCellExperiment / SpatialExperiment objects.

The cell-associated metadata are stored in the `colData` of the SingleCellExperiment and SpatialExperiment objects. These metadata include various information about cells, milieu, samples, and patients. For instance, cell types can be retrieved with `colData(sce)$cell_type` and cell clusters with `colData(sce)$cell_cluster`.

Neighborhood information, defined here as cells that are localized next to each other, is stored as a `SelfHits` object in the `colPairs` slot of the `SingleCellExperiment` and SpatialExperiment objects.

For more information, please refer to the Hoch, Schulz, et al. publication.

Dataset versions: a `version` argument can be passed to the function to specify which dataset version should be retrieved.

- `` `v1` ``: first published version

File sizes:

- `` `images_rna` ``: size in memory = 13.9 Gb, size on disk = 954 Mb.
- `` `masks_rna` ``: size in memory = 347 Mb, size on disk = 11 Mb.
- `` `sce_rna` ``: size in memory = 774 Mb, size on disk = 401 Mb.
- `` `masks_full_rna` ``: size in memory = 1.1 Gb, size on disk = 30 Mb.
- `` `sce_full_rna` ``: size in memory = 2.0 Gb, size on disk = 1.1 Gb.
- `` `images_protein` ``: size in memory = 16.8 Gb, size on disk = 1.2 Gb.
- `` `masks_protein` ``: size in memory = 374 Mb, size on disk = 12 Mb.
- `` `sce_protein` ``: size in memory = 856 Mb, size on disk = 531 Mb.
- `` `masks_full_protein` ``: size in memory = 1.2 Gb, size on disk = 35 Mb.
- `` `sce_full_protein` ``: size in memory = 2.2 Gb, size on disk = 1.4 Gb.

When storing images on disk, these need to be first fully read into memory before writing them to disk. This means the process of downloading the data is slower than directly keeping them in memory. However, downstream analysis will lose its memory overhead when storing images on disk.

Original source: Hoch, Schulz et al. (2022): https://doi.org/10.1126/sciimmunol.abk1692

Original link to raw data: https://doi.org/10.5281/zenodo.5994136.

Value

A SingleCellExperiment object with single cell data, a SpatialExperiment object with single cell data, a CytoImageList object containing multichannel images, or a CytoImageList object containing cell segmentation masks.

Author(s)

Nicolas Damond

References

Hoch, Schulz et al. (2022). Multiplexed imaging mass cytometry of the chemokine milieus in melanoma characterizes features of the response to immunotherapy *Sci Immunol* 7(70):eabk1692.

Examples

```
# Load single cell data
sce <- HochSchulz_2022_Melanoma(data_type = "sce")
print(sce)

# Display metadata
HochSchulz_2022_Melanoma(data_type = "sce", metadata = TRUE)

# Load masks on disk
library(HDF5Array)
masks <- HochSchulz_2022_Melanoma(data_type = "masks", on_disk = TRUE,
h5FilesPath = getHDF5DumpDir())
print(head(masks))
```

IMMUcan_2022_CancerExample

Obtain the IMMUcan_2022_CancerExample dataset

Description

Obtain the IMMUcan_2022_CancerExample dataset, which consists of three data objects: single cell data, multichannel images and cell segmentation masks. Data were obtained by imaging mass cytometry (IMC) of sections of 4 patients with different tumor indications.

Usage

```
IMMUcan_2022_CancerExample(
  data_type = c("sce", "spe", "images", "masks"),
  version = "latest",
  metadata = FALSE,
  on_disk = FALSE,
  h5FilesPath = NULL,
  force = FALSE
)
```

Arguments

`data_type`	type of object to load, ‘images‘ for multichannel images or ‘masks‘ for cell segmentation masks. Single cell data are retrieved using either ‘sce‘ for the `SingleCellExperiment` format or ‘spe‘ for the `SpatialExperiment` format.
`version`	dataset version. By default, the latest version is returned.
`metadata`	if FALSE (default), the data object selected in `data_type` is returned. If TRUE, only the metadata associated to this object is returned.
`on_disk`	logical indicating if images in form of HDF5Array objects (as .h5 files) should be stored on disk rather than in memory. This setting is valid when downloading `images` and `masks`.
`h5FilesPath`	path to where the .h5 files for on disk representation are stored. This path needs to be defined when `on_disk = TRUE`. When files should only temporarily be stored on disk, please set `h5FilesPath = getHDF5DumpDir()`.
`force`	logical indicating if images should be overwritten when files with the same name already exist on disk.

Details

This is an Imaging Mass Cytometry (IMC) dataset used in the IMC data analysis book

- `images` contains 14 multichannel images, each containing 50 channels, in the form of a CytoImageList class object.
- `masks` contains the cell segmentation masks associated with the images, in the form of a CytoImageList class object.
- `sce` contains the single cell data extracted from the multichannel images using the cell segmentation masks, as well as the associated metadata, in the form of a SingleCellExperiment object. Single cell data can also be retrieved as a SpatialExperiment object. This represents a total of 46,825 cells x 40 channels.

All data are downloaded from ExperimentHub and cached for local re-use.

Mapping between the three data objects is performed via variables located in their metadata columns: `mcols()` for the CytoImageList objects and `ColData()` for the SingleCellExperiment object. Mapping at the image level can be performed with the `sample_id` or `image_name` variables. Mapping between cell segmentation masks and single cell data is performed with the `cell_number` variable, the values of which correspond to the intensity values of the `masks` object. For practical examples, please refer to the "Accessing IMC datasets" vignette.

This imaging mass cytometry dataset serves as an example to demonstrate downstream analysis tools including spatial data analysis. The data was generated as part of the Integrated iMMUnoprofiling of large adaptive CANcer patient cohorts (IMMUcan) project (immucan.eu) using the Hyperion imaging system.

Relevant entries to the `colData` slot are as follows:

- `sample_id` image name.
- `cell_number` cell identifier.
- `width_px` width of the image.
- `height_px` height of the image.
- `patient_id` patient identifier.
- `ROI` region of interest identifier.
- `indication` cancer type.
- `cell_labels` labels of manually labelled cells.
- `cell_type` cell type as defined by classification.
- `spatial_community` identifiers of each spatial tumor or non-tumor community
- `cn_celltypes` cellular neighborhoods as defined by clustering cells based on the frequency of neighboring cell types.
- `cn_expression` cellular neighborhoods as defined by clustering cells based on the mean expression of neighboring cells
- `lisa_clusters` cellular neighborhoods as detected by the lisaClust package.
- `spatial_context` spatial contexts defined in `cn_celltype`.
- `spatial_context_filtered` filtered spatial context identifiers.
- `patch_id` identifier of the spatial tumor patch.
- `cell_x` spatial x coordinate.
- `cell_y` spatial y coordinate.

The marker-associated metadata, including antibody information and metal tags are stored in the `rowData` of the SingleCellExperiment object.

The `assay` slot of the SingleCellExperiment object contains two assays:

- `counts`: mean ion counts per cell
- `exprs`: arsinh-transformed counts per cell, with cofactor 1.

The `colPair` slot of the SingleCellExperiment object contains the following spatial object graphs:

- `neighborhood` steinbock generated graph.
- `knn_interaction_graph` 20-nearest neighbor graph.
- `expansion_interaction_graph` expansion graph using a threshold of 20.
- `delaunay_interaction_graph` interaction graph constructed by delaunay triangulation.
- `knn_spatialcontext_graph` 40-nearest neighbor graph.

File sizes:

- `` `images` ``: size in memory = 1.5 Gb, size on disk = 786 Mb.
- `` `masks` ``: size in memory = 19 Mb, size on disk = 1.2 Mb.
- `` `sce` ``: size in memory = 182 Mb, size on disk = 82 Mb.
- `` `spe` ``: size in memory = 183 Mb, size on disk = 81 Mb.

When storing images on disk, these need to be first fully read into memory before writing them to disk. This means the process of downloading the data is slower than directly keeping them in memory. However, downstream analysis will lose its memory overhead when storing images on disk.

Value

A SingleCellExperiment object with single cell data, a CytoImageList object containing multichannel images, or a CytoImageList object containing cell segmentation masks.

Author(s)

Nils Eling

Examples

```
# Load single cell data
sce <- IMMUcan_2022_CancerExample(data_type = "sce")
print(sce)

# Display metadata
IMMUcan_2022_CancerExample(data_type = "sce", metadata = TRUE)

# Load masks on disk
library(HDF5Array)
masks <- IMMUcan_2022_CancerExample(data_type = "masks", on_disk = TRUE,
h5FilesPath = getHDF5DumpDir())
print(head(masks))
```

`JacksonFischer_2020_BreastCancer` *Obtain the JacksonFischer_2020_BreastCancer dataset*

Description

Obtain the JacksonFischer_2020_BreastCancer dataset, which consists of three data objects: single cell data, multichannel images and cell segmentation masks. The data was obtained by imaging mass cytometry (IMC) of tumour tissue from patients with breast cancer.

Usage

```
JacksonFischer_2020_BreastCancer(
  data_type = c("sce", "spe", "images", "masks"),
  full_dataset = FALSE,
  cohort = "Basel",
  version = "latest",
  metadata = FALSE,
  on_disk = FALSE,
  h5FilesPath = NULL,
  force = FALSE
)
```

Arguments

`data_type`	type of object to load, ‘images‘ for multichannel images or ‘masks‘ for cell segmentation masks. Single cell data are retrieved using either ‘sce‘ for the `SingleCellExperiment` format or ‘spe‘ for the `SpatialExperiment` format.
`full_dataset`	if FALSE (default), a subset corresponding to 100 images is returned. If TRUE, the full dataset is returned, including both "Basel" and "Zurich" cohorts. Due to memory space limitations, this option is only available for single cell data and masks, not for `data_type = "images"`.
`cohort`	which patient cohort should be returned? Can be set to "Basel" (default) or "Zurich". Ignored if `full_dataset` is set to TRUE.
`version`	dataset version. By default, the latest version is returned.
`metadata`	if FALSE (default), the data object selected in `data_type` is returned. If TRUE, only the metadata associated to this object is returned.
`on_disk`	logical indicating if images in form of HDF5Array objects (as .h5 files) should be stored on disk rather than in memory. This setting is valid when downloading `images` and `masks`.
`h5FilesPath`	path to where the .h5 files for on disk representation are stored. This path needs to be defined when `on_disk = TRUE`. When files should only temporarily be stored on disk, please set `h5FilesPath = getHDF5DumpDir()`.
`force`	logical indicating if images should be overwritten when files with the same name already exist on disk.

Details

This is an Imaging Mass Cytometry (IMC) dataset from Jackson, Fischer et al. (2020):

- `images` contains a hundred 42-channel images in the form of a CytoImageList class object.
- `masks` contains the cell segmentation masks associated with the images, in the form of a CytoImageList class object.
- `sce` contains the single cell data extracted from the multichannel images using the cell segmentation masks, as well as the associated metadata, in the form of a SingleCellExperiment. This represents a total of 285,851 cells x 42 channels.
- `spe` same single cell data as for `sce`, but in the SpatialExperiment format.

All data are downloaded from ExperimentHub and cached for local re-use.

Mapping between the three data objects is performed via variables located in their metadata columns: `mcols()` for the CytoImageList objects and `ColData()` for the SingleCellExperiment and SpatialExperiment objects. Mapping at the image level can be performed with the `image_name` variable. Mapping between cell segmentation masks and single cell data is performed with the `cell_number` variable, the values of which correspond to the intensity values of the `masks` object. For practical examples, please refer to the "Accessing IMC datasets" vignette.

This dataset is a subset of the complete Jackson, Fischer et al. (2020) dataset comprising the data from tumour tissue from 100 patients with breast cancer (one image per patient). By default, data from the "Basel" cohort are returned. By setting `cohort = "Zurich"`, data from the "Zurich" cohort, corresponding to images and associated data from 72 patients, are returned. For details about the patient cohorts, refer to the publication. If `full_dataset = TRUE`, the full dataset is returned (including both "Basel" and "Zurich" patient cohorts). This option is not available for multichannel images.

The `assay` slot of the SingleCellExperiment object contains three assays:

- `counts` contains mean ion counts per cell.
- `exprs` contains arsinh-transformed counts, with cofactor 1.
- `quant_norm` contains quantile-normalized counts (0 to 1, 99th percentile).

The marker-associated metadata, including antibody information and metal tags are stored in the `rowData` of the SingleCellExperiment and SpatialExperiment objects.

The cell-associated metadata are stored in the `colData` of the SingleCellExperiment and SpatialExperiment objects. These metadata include clusters (in `colData(sce)$cell_cluster_phenograph`) and metaclusters (in `colData(sce)$cell_metacluster`), as well as spatial information (e.g., cell areas are stored in `colData(sce)$cell_area`).

The clinical data are also stored in the `colData` of the SingleCellExperiment and SpatialExperiment objects. For instance, the tumor grades can be retrieved with `colData(sce)$tumor_grade`.

Dataset versions: a `version` argument can be passed to the function to specify which dataset version should be retrieved.

- `` `v0` ``: original version (Bioconductor <= 3.15).
- `` `v1` ``: consistent object formatting across datasets.
- `` `v2` ``: added full datasets and Zurich cohort.

File sizes:

- `` `images_basel` ``: size in memory = 19 Gb, size on disk = 2.0 Gb.
- `` `masks_basel` ``: size in memory = 433 Mb, size on disk = 10 Mb.

- `sce_basel`: size in memory = 513 Mb, size on disk = 270 Mb.
- `images_zurich`: size in memory = 6.0 Gb, size on disk = 724 Mb.
- `masks_zurich`: size in memory = 137 Mb, size on disk = 3.4 Mb.
- `sce_zurich`: size in memory = 188 Mb, size on disk = 105 Mb.
- `masks_full`: size in memory = 2.1 Gb, size on disk = 10 Mb.
- `sce_full`: size in memory = 2.2 Gb, size on disk = 1.2 Gb.

When storing images on disk, these need to be first fully read into memory before writing them to disk. This means the process of downloading the data is slower than directly keeping them in memory. However, downstream analysis will lose its memory overhead when storing images on disk.

Original source: Jackson, Fischer et al. (2020): https://doi.org/10.1038/s41586-019-1876-x

Original link to raw data, containing the entire dataset: https://doi.org/10.5281/zenodo.3518284

Value

A SingleCellExperiment object with single cell data, a SpatialExperiment object with single cell data, a CytoImageList object containing multichannel images, or a CytoImageList object containing cell segmentation masks.

Author(s)

Nicolas Damond

References

Jackson, Fischer et al. (2020). The single-cell pathology landscape of breast cancer. *Nature* 578(7796), 615-620.

Examples

```
# Load single cell data
sce <- JacksonFischer_2020_BreastCancer(data_type = "sce")
print(sce)

# Display metadata
JacksonFischer_2020_BreastCancer(data_type = "sce", metadata = TRUE)

# Load masks on disk
library(HDF5Array)
masks <- JacksonFischer_2020_BreastCancer(data_type = "masks", on_disk =
TRUE, h5FilesPath = getHDF5DumpDir())
print(head(masks))
```

listDatasets *List all available datasets*

Description

Summary information for all available datasets in the **imcdatasets** package.

Usage

```
listDatasets()
```

Details

Each dataset contains single-cell data, multichannel images and cell segmentation masks.

Value

A DataFrame where each row corresponds to a dataset, containing the fields:

- `FunctionCall`, the R function call required to construct the dataset.
- `Species`, species of origin.
- `Tissue`, the tissue that was imaged.
- `NumberOfCells`, the total number of cells in the dataset.
- `NumberOfImages`, the total number of images in the dataset.
- `NumberOfChannels`, the number of channels per image.
- `Reference`, a Markdown-formatted citation to `scripts/ref.bib` in the **imcdatasets** installation directory.

Examples

```
listDatasets()
```

Meyer_2025_TripleNegativeBreastCancer *Obtain the Meyer_2025_TripleNegativeBreastCancer dataset*

Description

Obtain the Meyer_2025_TripleNegativeBreastCancer dataset. The dataset is composed of three data objects: single cell data, multichannel images and cell segmentation masks. The data was obtained by imaging mass cytometry (IMC) of a tissue microarray (TMA) with multiple cores of formalin-fixed paraffin-embedded (FFPE) tissue from 215 patients with triple-negative breast cancer.

Usage

```
Meyer_2025_TripleNegativeBreastCancer(
  data_type = c("sce", "spe", "images", "masks"),
  full_dataset = FALSE,
  version = "latest",
  metadata = FALSE,
  on_disk = FALSE,
  h5FilesPath = NULL,
  force = FALSE
)
```

Arguments

`data_type`	type of object to load, 'images' for multichannel images or 'masks' for cell segmentation masks. Single cell data are retrieved using either 'sce' for the `SingleCellExperiment` format or 'spe' for the `SpatialExperiment` format.
`full_dataset`	if FALSE (default), a subset corresponding to to 125 images from 60 patients sampled across proposed patient groups is returned. This includes all images visually presented in the publication. If TRUE, the full dataset (corresponding to 450 images) is returned. Due to memory space limitations, this option is only available for single cell data and masks, not for `data_type = "images"`.
`version`	dataset version. By default, the latest version is returned.
`metadata`	if FALSE (default), the data object selected in `data_type` is returned. If TRUE, only the metadata associated to this object is returned.
`on_disk`	logical indicating if images in form of HDF5Array objects (as .h5 files) should be stored on disk rather than in memory. This setting is valid when downloading `images` and `masks`.
`h5FilesPath`	path to where the .h5 files for on disk representation are stored. This path needs to be defined when `on_disk = TRUE`. When files should only temporarily be stored on disk, please set `h5FilesPath = getHDF5DumpDir()`.
`force`	logical indicating if images should be overwritten when files with the same name already exist on disk.

Details

This is an Imaging Mass Cytometry (IMC) dataset from Meyer et al. (2025):

- `images` contains 125 39-channel images in the form of a CytoImageList class object.
- `masks` contains the cell segmentation masks associated with the images, in the form of a CytoImageList class object.
- `sce` contains the single cell data extracted from the multichannel images using the cell segmentation masks, as well as the associated metadata, in the form of a SingleCellExperiment object.
- `spe` same single cell data as for `sce`, but in the SpatialExperiment format.

All data are downloaded from ExperimentHub and cached for local re-use.

Mapping between the three data objects is performed via variables located in their metadata columns: `mcols()` for the CytoImageList objects and `ColData()` for the SingleCellExperiment and SpatialExperiment objects. Mapping at the image level can be performed with the `image_name` or

image_number variables. Mapping between cell segmentation masks and single cell data is performed with the cell_number variable, the values of which correspond to the intensity values of the masks object. For practical examples, please refer to the "Accessing IMC datasets" vignette.

The assay slots of the SingleCellExperiment and SpatialExperiment objects contain three assays:

- counts contains raw mean ion counts per cell.
- exprs contains arsinh-transformed counts, with cofactor 1.
- min_max contains 0-1 normalized .

The marker-associated metadata, including antibody information and metal tags are stored in the rowData of the SingleCellExperiment / SpatialExperiment objects.

The cell-associated metadata are stored in the colData of the SingleCellExperiment and SpatialExperiment objects. These metadata include various information about cells, tumors and patients. For instance, cell metacluster can be retrieved with colData(sce)$cell_metacluster and patient groups with colData(sce)$patient_patientgroup.

Neighborhood information, defined here as cells that are localized next to each other, is stored as a SelfHits object in the colPairs slot of the SingleCellExperiment and SpatialExperiment objects.

For more information, please refer to the Meyer et al. publication.

Dataset versions: a version argument can be passed to the function to specify which dataset version should be retrieved.

- `v1`: first published version

File sizes:

- `images`: size in memory = 20.9 Gb, size on disk = 1.6 Gb.
- `masks`: size in memory = 269 Mb, size on disk = 8 Mb.
- `masks_full`: size in memory = 942 Mb, size on disk = 29 Mb.
- `sce`: size in memory = 451 Mb, size on disk = 241 Mb.
- `sce_full`: size in memory = 1.6 Gb, size on disk = 866 Mb.

When storing images on disk, these need to be first fully read into memory before writing them to disk. This means the process of downloading the data is slower than directly keeping them in memory. However, downstream analysis will lose its memory overhead when storing images on disk.

Original source: Meyer et al. (2025): https://doi.org/10.1016/j.ccell.2025.06.019

Original link to raw data: https://zenodo.org/records/15304181.

Value

A SingleCellExperiment object with single cell data, a SpatialExperiment object with single cell data, a CytoImageList object containing multichannel images, or a CytoImageList object containing cell segmentation masks.

Author(s)

Lasse Meyer

References

Meyer, Jackson et al. (2025). A stratification system for breast cancer based on basoluminal tumor cells and spatial tumor architecture *Cancer Cell* 43(9):1637–1655.e9.

Examples

```
# Load single cell data
sce <- Meyer_2025_TripleNegativeBreastCancer(data_type = "sce")
print(sce)

# Display metadata
Meyer_2025_TripleNegativeBreastCancer(data_type = "sce", metadata = TRUE)

# Load masks on disk
library(HDF5Array)
masks <- Meyer_2025_TripleNegativeBreastCancer(data_type = "masks", on_disk = TRUE,
h5FilesPath = getHDF5DumpDir())
print(head(masks))
```

`SteenbuckDamond_2026_T1D` *Obtain the SteenbuckDamond_2026_T1D dataset*

Description

Obtain the SteenbuckDamond_2026_T1D dataset, which is composed of two panels (Islet + Immune) that were acquired on two consecutive 4 μm sections. Each dataset (panel) is composed of three data objects: single cell data, multichannel images and cell segmentation masks. The data was obtained by imaging mass cytometry (IMC) measuring formalin-fixed paraffin-embedded (FFPE) tissue from 88 individuals along the disease progression of T1D. It contains, both Controls, single and multi-auto-antibody positive T1D donors (sAAb+/mAAb+), and donors with recent and long-standing T1D (Onset/LD).

Usage

```
SteenbuckDamond_2026_T1D(
  data_type = c("sce", "spe", "images", "masks"),
  panel = "islet",
  full_dataset = FALSE,
  version = "latest",
  metadata = FALSE,
  on_disk = FALSE,
  h5FilesPath = NULL,
  force = FALSE
)
```

Arguments

`data_type`	type of object to load, ‘images‘ for multichannel images or ‘masks‘ for cell segmentation masks. Single cell data are retrieved using either ‘sce‘ for the `SingleCellExperiment` format or ‘spe‘ for the `SpatialExperiment` format.

panel	which panel should be returned? Can be set to "islet" (default) or "immune".
full_dataset	if FALSE (default), a subset of 120 informative images is returned. These are sampled along the disease progression of T1D. It also contains insulitic images, i.e. T-cell infiltration of islets. If TRUE, the full dataset (corresponding to 7024 images) is returned. Due to memory space limitations, this option is only available for single cell data and masks, not for data_type = "images". All images can be retrieved from Zenodo.
version	dataset version. By default, the latest version is returned.
metadata	if FALSE (default), the data object selected in data_type is returned. If TRUE, only the metadata associated to this object is returned.
on_disk	logical indicating if images in form of HDF5Array objects (as .h5 files) should be stored on disk rather than in memory. This setting is valid when downloading images and masks.
h5FilesPath	path to where the .h5 files for on disk representation are stored. This path needs to be defined when on_disk = TRUE. When files should only temporarily be stored on disk, please set h5FilesPath = getHDF5DumpDir().
force	logical indicating if images should be overwritten when files with the same name already exist on disk.

Details

This is an Imaging Mass Cytometry (IMC) dataset from Steenbuck, Damond et al. (2026):

- images contains 120 47-channel images in the form of a CytoImageList class object.
- masks contains the cell segmentation masks associated with the images, in the form of a CytoImageList class object.
- sce contains the single cell data extracted from the multichannel images using the cell segmentation masks, as well as the associated metadata, in the form of a SingleCellExperiment object.
- spe same single cell data as for sce, but in the SpatialExperiment format.

All data are downloaded from ExperimentHub and cached for local re-use.

Mapping between the three data objects is performed via variables located in their metadata columns: mcols() for the CytoImageList objects and ColData() for the SingleCellExperiment and SpatialExperiment objects. Mapping at the image level can be performed with the image_name or image_number variables. Mapping between cell segmentation masks and single cell data is performed with the cell_number variable, the values of which correspond to the intensity values of the masks object. For practical examples, please refer to the "Accessing IMC datasets" vignette.

The assay slots of the SingleCellExperiment and SpatialExperiment objects contain three assays:

- raw contains raw mean ion counts per cell.
- exprs contains arsinh-transformed counts, with cofactor 1.
- scaled contains scaled asinh-transformed raw counts.

The marker-associated metadata, including antibody information and metal tags are stored in the rowData of the SingleCellExperiment / SpatialExperiment objects.

The cell-associated metadata are stored in the colData of the SingleCellExperiment and SpatialExperiment objects. These metadata include various information about cells, milieu, samples, and patients. For instance, cell types can be retrieved with colData(sce)$cell_type and immune cell clusters with colData(sce)$cell_cluster_scaled. The latter for the Immune panel only.

Neighborhood information, defined here as cells that are localized next to each other, is stored as a `SelfHits` object in the `colPairs` slot of the `SingleCellExperiment` and SpatialExperiment objects.

For more information, please refer to the Steenbuck, Damond, et al. publication.

Dataset versions: a `version` argument can be passed to the function to specify which dataset version should be retrieved.

- `` `v1` ``: first published version

File sizes:#'

- `` `images_immune` ``: size in memory = 6.3 Gb, size on disk = 650.3 Mb.
- `` `masks_immune` ``: size in memory = 138 Mb, size on disk = 4.7 Mb.
- `` `sce_immune` ``: size in memory = 442 Mb, size on disk = 217.1 Mb.
- `` `masks_full_immune` ``: size in memory = 8.6 Gb, size on disk = 337.3 Mb.
- `` `sce_full_immune` ``: size in memory = 21.5 Gb, size on disk = 13.0 Gb.
- `` `images_islet` ``: size in memory = 3.8 Gb, size on disk = 738.8 Mb.
- `` `masks_islet` ``: size in memory = 83 Mb, size on disk = 2.8 Mb.
- `` `sce_islet` ``: size in memory = 277 Mb, size on disk = 128.5 Mb.
- `` `masks_full_islet` ``: size in memory = 5.1 Gb, size on disk = 194.8 Mb.
- `` `sce_full_islet` ``: size in memory = 12.8 Gb, size on disk = 6.8 Gb.

When storing images on disk, these need to be first fully read into memory before writing them to disk. This means the process of downloading the data is slower than directly keeping them in memory. However, downstream analysis will lose its memory overhead when storing images on disk.

Original source: Steenbuck, Damond, et al. (2026): https://doi.org/10.1038/s42255-026-01559-z

Original link to raw data: https://doi.org/10.5281/zenodo.14968076.

Value

A SingleCellExperiment object with single cell data, a SpatialExperiment object with single cell data, a CytoImageList object containing multichannel images, or a CytoImageList object containing cell segmentation masks.

Author(s)

Nathan Steenbuck

References

Steenbuck, N., Damond, N., et al. (2026). Imaging mass cytometry reveals functional and immunological changes during type 1 diabetes progression in human pancreata. *Nat Metab* https://doi.org/10.1038/s42255-026-01559-z

Examples

```
# Load single cell data
sce <- SteenbuckDamond_2026_T1D(data_type = "sce")
print(sce)

# Display metadata
SteenbuckDamond_2026_T1D(data_type = "sce", metadata = TRUE)

# Load masks on disk
library(HDF5Array)
masks <- SteenbuckDamond_2026_T1D(data_type = "masks", on_disk = TRUE,
h5FilesPath = getHDF5DumpDir())
print(head(masks))
```

`Zanotelli_2020_Spheroids` *Obtain the Zanotelli_2020_Spheroids dataset*

Description

Obtain the Zanotelli_2020_Spheroids dataset, which consists of three data objects: single cell data, multichannel images and cell segmentation masks. The data were obtained by imaging mass cytometry (IMC) of sections of 3D spheroids generated from different cell lines.

Usage

```
Zanotelli_2020_Spheroids(
  data_type = c("sce", "spe", "images", "masks"),
  version = "latest",
  metadata = FALSE,
  on_disk = FALSE,
  h5FilesPath = NULL,
  force = FALSE
)
```

Arguments

`data_type`	type of object to load, 'images' for multichannel images or 'masks' for cell segmentation masks. Single cell data are retrieved using either 'sce' for the `SingleCellExperiment` format or 'spe' for the `SpatialExperiment` format.
`version`	dataset version. By default, the latest version is returned.
`metadata`	if FALSE (default), the data object selected in `data_type` is returned. If TRUE, only the metadata associated to this object is returned.
`on_disk`	logical indicating if images in form of HDF5Array objects (as .h5 files) should be stored on disk rather than in memory. This setting is valid when downloading `images` and `masks`.
`h5FilesPath`	path to where the .h5 files for on disk representation are stored. This path needs to be defined when `on_disk = TRUE`. When files should only temporarily be stored on disk, please set `h5FilesPath = getHDF5DumpDir()`.
`force`	logical indicating if images should be overwritten when files with the same name already exist on disk.

Details

This is an Imaging Mass Cytometry (IMC) dataset from Zanotelli et al. (2020), consisting of three data objects:

- `images` contains 517 multichannel images, each containing 51 channels, in the form of a CytoImageList class object.
- `masks` contains the cell segmentation masks associated with the images, in the form of a CytoImageList class object.
- `sce` contains the single cell data extracted from the multichannel images using the cell segmentation masks, as well as the associated metadata, in the form of a SingleCellExperiment. This represents a total of 229,047 cells x 51 channels.
- `spe` same single cell data as for `sce`, but in the SpatialExperiment format.

All data are downloaded from ExperimentHub and cached for local re-use.

Mapping between the three data objects is performed via variables located in their metadata columns: `mcols()` for the CytoImageList objects and `ColData()` for the SingleCellExperiment and SpatialExperiment objects. Mapping at the image level can be performed with the `image_name` or `image_number` variables. Mapping between cell segmentation masks and single cell data is performed with the `cell_number` variable, the values of which correspond to the intensity values of the `masks` object. For practical examples, please refer to the "Accessing IMC datasets" vignette.

This dataset was obtained as following (the names of the experimental variables, located in the `colData` of the SingleCellExperiment and SpatialExperiment objects, are indicated in parentheses): *i)* Cells from four different cell lines (`cell_line`) were seeded at three different densities (`treatment_concentration`, relative densities) and grown for either 72 or 96 hours (`treatment_time_point`, duration in hours). In the appropriate experimental conditions (see the paper for details), the cells aggregate into 3D spheroids. *ii)* Cells were harvested and pooled into 60-well barcoding plates. *iii)* A pellet of each spheroid pool was generated and cut into several 6 um-thick sections. *iv)* A subset of these sections (`site_id`) were stained with an IMC panel and acquired as one or more acquisitions (`acquisition_id`) containing multiple spheres each. *v)* Spheres in these acquisitions were identified by computer vision and cropped into individual images (`image_number`).

Other relevant cell metadata include:

- `treatment_name`: experimental conditions in the format: `"Cell line name"_c"seeding density"_tp"time point"`.
- `cell_x/cell_y`: cell centroid position in the image.
- `cell_area`: area of the cell (um^2).
- `distance_rim`: estimated distance to spheroid border.
- `distance_sphere`: distance to spheroid section border.
- `distance_other_sphere`: distance to the closest of the other spheroid sections in the same image (if there is any).
- `distance_background`: distance to background pixels.

For a full description of the other experimental variables, please refer to the publication (https://doi.org/10.15252/msb.202 and to the original dataset repository (https://doi.org/10.5281/zenodo.4271910).

The marker-associated metadata, including antibody information and metal tags are stored in the `rowData` of the SingleCellExperiment and SpatialExperiment objects. The channels with names starting with "BC_" are the channels used for barcoding. Post-transcriptional modification of the protein targets are indicated in brackets.

The `assay` slots of the SingleCellExperiment and SpatialExperiment objects contain three assays:

- `counts` contains raw mean ion counts per cell.
- `exprs` contains arsinh-transformed counts, with cofactor 1.
- `quant_norm` contains counts censored at the 99th percentile and scaled 0-1.

In addition, the `altExp` slot of the SingleCellExperiment object contains another SingleCellExperiment object where the counts matrix represents raw mean ion counts for cells neighboring the current cell.

Neighborhood information, defined here as cells that are localized next to each other, is stored as a `SelfHits` object in the `colPairs` slot of the `SingleCellExperiment` and SpatialExperiment objects. Cells in the `SelfHits` object are represented by unique integers that map to the `cell_number_absolute` column of `colData(sce)`.

Dataset versions: a `version` argument can be passed to the function to specify which dataset version should be retrieved.

- `` `v0` ``: original version (Bioconductor <= 3.15).
- `` `v1` ``: consistent object formatting across datasets.

File sizes:

- `` `images` ``: size in memory = 21.2 Gb, size on disk = 860 Mb.
- `` `masks` ``: size in memory = 426 Mb, size on disk = 12 Mb.
- `` `sce` ``: size in memory = 564 Mb, size on disk = 319 Mb.
- `` `spe` ``: size in memory = 596 Mb, size on disk = 320 Mb.

When storing images on disk, these need to be first fully read into memory before writing them to disk. This means the process of downloading the data is slower than directly keeping them in memory. However, downstream analysis will lose its memory overhead when storing images on disk.

Original source: Zanotelli et al. (2020): https://doi.org/10.15252/msb.20209798

Original link to raw data, also containing the entire dataset: https://doi.org/10.5281/zenodo.4271910

Value

A SingleCellExperiment object with single cell data, a SpatialExperiment object with single cell data, a CytoImageList object containing multichannel images, or a CytoImageList object containing cell segmentation masks.

Author(s)

Nicolas Damond

References

Zanotelli VRT et al. (2020). A quantitative analysis of the interplay of environment, neighborhood, and cell state in 3D spheroids *Mol Syst Biol* 16(12), e9798.

Examples

```
# Load single cell data
sce <- Zanotelli_2020_Spheroids(data_type = "sce")
print(sce)

# Display metadata
Zanotelli_2020_Spheroids(data_type = "sce", metadata = TRUE)

# Load masks on disk
library(HDF5Array)
masks <- Zanotelli_2020_Spheroids(data_type = "masks", on_disk = TRUE,
h5FilesPath = getHDF5DumpDir())
print(head(masks))
```

Index

CytoImageList, *3, 4*, *6, 7*, *9, 10*, *12, 13*, *15, 16*, *18, 19*, *21, 22*

Damond_2019_Pancreas, 2
DataFrame, *14*

HDF5Array, *3*, *6*, *8*, *11*, *15*, *18*, *20*
HochSchulz_2022_Melanoma, 5

IMMUcan_2022_CancerExample, 8

JacksonFischer_2020_BreastCancer, 11

listDatasets, 14

Meyer_2025_TripleNegativeBreastCancer, 14

SingleCellExperiment, *3, 4*, *6, 7*, *9, 10*, *12, 13*, *15, 16*, *18, 19*, *21, 22*
SpatialExperiment, *3, 4*, *6, 7*, *9*, *12, 13*, *15, 16*, *18, 19*, *21, 22*
SteenbuckDamond_2026_T1D, 17

Zanotelli_2020_Spheroids, 20