Package ‘ttservice’

October 14, 2022

Type Package

Title A Service for Tidy Transcriptomics Software Suite

Version 0.2.2

Description It provides generic methods that are used by more than one package, avoiding conflicts. This package will be imported by 'tidySingleCellExperiment' and 'tidyseurat'.

License GPL-3

Depends R (>= 4.0.0)

Imports dplyr

Suggests methods

Encoding UTF-8

RoxygenNote 7.2.0

NeedsCompilation no

Author Stefano Mangiola [aut, cre]

Maintainer Stefano Mangiola <`mangiolastefano@gmail.com`>

Repository CRAN

Date/Publication 2022-06-24 06:50:02 UTC

R topics documented:

`bind_rows` *#’ Efficiently bind multiple data frames by row and column*

Description

This is an efficient implementation of the common pattern of ‘do.call(rbind, dfs)‘ or ‘do.call(cbind, dfs)‘ for binding many data frames into one.

This is an efficient implementation of the common pattern of ‘do.call(rbind, dfs)‘ or ‘do.call(cbind, dfs)‘ for binding many data frames into one.

Usage

```
bind_rows(..., .id = NULL, add.cell.ids = NULL)

bind_cols(..., .id = NULL)
```

Arguments

`...`	Data frames to combine. Each argument can either be a data frame, a list that could be a data frame, or a list of data frames. When row-binding, columns are matched by name, and any missing columns will be filled with NA. When column-binding, rows are matched by position, so all data frames must have the same number of rows. To match by value, not position, see mutate-joins.
`.id`	Data frame identifier. When ‘.id‘ is supplied, a new column of identifiers is created to link each row to its original data frame. The labels are taken from the named arguments to ‘bind_rows()‘. When a list of data frames is supplied, the labels are taken from the names of the list. If no names are found a numeric sequence is used instead.
`add.cell.ids`	from Seurat 3.0 A character vector of length(x = c(x, y)). Appends the corresponding values to the start of each objects’ cell names.

Details

The output of ‘bind_rows()‘ will contain a column if that column appears in any of the inputs.

The output of ‘bind_rows()‘ will contain a column if that column appears in any of the inputs.

Value

‘bind_rows()‘ and ‘bind_cols()‘ return the same type as the first input, either a data frame, ‘tbl_df‘, or ‘grouped_df‘.

‘bind_rows()‘ and ‘bind_cols()‘ return the same type as the first input, either a data frame, ‘tbl_df‘, or ‘grouped_df‘.

Examples

```
print("small_pbmc |> bind_rows(small_pbmc)")



print("small_pbmc |> bind_cols(annotation_column)")
```

`join_features` *join_features*

Description

join_features() extracts and joins information for specific features

Usage

```
join_features(
  .data,
  features = NULL,
  all = FALSE,
  exclude_zeros = FALSE,
  shape = "long",
  ...
)
```

Arguments

`.data`	A tidy SingleCellExperiment object
`features`	A vector of feature identifiers to join
`all`	If TRUE return all
`exclude_zeros`	If TRUE exclude zero values
`shape`	Format of the returned table "long" or "wide"
`...`	Parameters to pass to join wide, i.e. assay name to extract feature abundance from and gene prefix, for shape="wide"

Details

This function extracts information for specified features and returns the information in either long or wide format.

Value

A ‘tbl‘ containing the information.for the specified features

Examples

```
print("this is a method definition. Example is not applicable")
# <SCE_object> |> join_features(features=c("HLA-DRA", "LYZ"))
```

Index