

Package ‘IrisSpatialFeatures’

March 1, 2018

Type Package

Title A package to extract spatial features based on multiplex IF images

Version 1.3.0

Date 2017-10-11

Author Daniel Gusenleitner, Jason L. Weirather

Maintainer Daniel Gusenleitner <Daniel_Gusenleitner@dfci.harvard.edu>

Depends R (>= 3.4)

Imports Rcpp (>= 0.12.7), SpatialTools, gplots, spatstat, tiff, RColorBrewer, methods, grDevices, graphics, stats, utils, data.table, ggplot2, dplyr, magrittr, tibble

Description IrisSpatialFeatures reads the output of the PerkinElmer inForm software and calculates a variety of spatial statistics. In addition to simple counts, it can derive average nearest neighbors for each cell-type and interaction summary profiles for each celltype. These statistics are derived across images, both overall and regions of interest as defined by user defined masks.

License MIT + file LICENSE

LinkingTo Rcpp

LazyData true

VignetteBuilder knitr

RoxygenNote 6.0.1

Suggests knitr, rmarkdown, testthat

biocViews FeatureExtraction, SingleCell

NeedsCompilation yes

R topics documented:

as.data.frame,ImageSet-method	2
collapse_markers	3
Coordinate-class	4
extract_features	4
extract_interactions	5
extract_mask	6

extract_nearest_neighbor	6
extract_proximity	7
extract_ROI	8
feature_selection	9
get_all_interactions	9
get_all_nearest_neighbors	10
get_all_proximities	11
get_counts_per_mm2	11
get_counts_per_mm2_noncollapsed	12
get_count_ratios	13
get_interactions	13
get_nearest_neighbors	14
get_proximities	15
ImageSet-class	15
interaction_maps	16
IrisSpatialFeatures	17
IrisSpatialFeatures_data	17
neighbor_ray_plot	17
overview_plot	18
plot,ImageSet-method	19
plot_interactions	20
plot_nearest_neighbor	21
plot_proximities	22
raw_data-class	23
rayplot_single_coordinate	23
read_raw	24
Sample-class	25
threshold_dataset	25

Index 27

as.data.frame,ImageSet-method

Read inForm output and store it in an IrisSpatialFeatures ImageSet object.

Description

Read inForm output and store it in an IrisSpatialFeatures ImageSet object.

Usage

```
## S4 method for signature 'ImageSet'
as.data.frame(x)
```

Arguments

x	Iris ImageSet boject
...	Additional arguments

Value

A dataframe

Examples

```
dataset <- IrisSpatialFeatures_data
dataframe <- as.data.frame(dataset)
```

collapse_markers	<i>This function collapses two markers into one, and reruns the counting of cells. Mostly a convenience function for the Shiny interface so we start with a completely split set successively adding more markers</i>
------------------	---

Description

This function collapses two markers into one, and reruns the counting of cells. Mostly a convenience function for the Shiny interface so we start with a completely split set successively adding more markers

Usage

```
collapse_markers(image_set, marker1, marker2, combined)

## S4 method for signature 'ImageSet'
collapse_markers(image_set, marker1, marker2, combined)
```

Arguments

image_set	IrisSpatialFeatures ImageSet object.
marker1	Name of the first marker that should be collapsed.
marker2	Name of the second marker that should be collapsed.
combined	Name of the combined marker.

Value

IrisSpatialFeatures ImageSet object.

Examples

```
dataset <- IrisSpatialFeatures_data
ds <- collapse_markers(dataset, marker1="SOX10+ PDL1+", marker2="SOX10+ PDL1-", combined="SOX10+")
```

Coordinate-class	<i>Class to represent a single imaging coordinate</i>
------------------	---

Description

Class to represent a single imaging coordinate

Slots

ppp A spatstat ppp object that contains all coordinate information
 raw Includes all raw data that is output from inForm
 mask List of user defined masks that define regions of interest
 coordinate_name Name of the current coordinate
 size_in_px Size of the image in pixel, accounting for mask size

Examples

```
coord <- new("Coordinate")
```

extract_features	<i>Extract all spatial features</i>
------------------	-------------------------------------

Description

Extract all spatial features

Usage

```
extract_features(x, ...)  
  
## S4 method for signature 'ImageSet'  
extract_features(x, name = "", rm.na = FALSE)
```

Arguments

x	IrisSpatialFeatures ImageSet object
...	Additional arguments
name	Prefix for all features, e.g. 'invasive_margin' (Default: '')
rm.na	Should features with NA values be removed (Default: FALSE)

Value

dataframe of features

Examples

```
#' #loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_nearest_neighbor(dataset,min_num_cells=2)
dataset <- extract_proximity(dataset,only_closest=TRUE,radii=25)
dataset <- extract_interactions(dataset)
extract_features(dataset)
```

extract_interactions *Extract interactions between all cell-types*

Description

Extract interactions between all cell-types

Usage

```
extract_interactions(x, ...)
```

S4 method for signature 'ImageSet'

```
extract_interactions(x)
```

Arguments

x	IrisSpatialFeatures ImageSet object
...	Additional arguments

Value

list of interactions

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_interactions(dataset)
```

extract_mask	<i>Read inForm output from a single coordinate</i>
--------------	--

Description

Read inForm output from a single coordinate

Usage

```
extract_mask(filename)
```

Arguments

filename	Name of the .tif file that contains the mask.
----------	---

Value

Mask matrix

Examples

```
extract_mask(system.file("extdata",
                        "MEL2", "MEL2_080416_2_Invasive_Margin.tif",
                        package = "IrisSpatialFeatures"))
```

extract_nearest_neighbor	<i>Extract the distance to each nearest neighbor for each cell-type</i>
--------------------------	---

Description

Extract the distance to each nearest neighbor for each cell-type

Usage

```
extract_nearest_neighbor(x, ...)
```

```
## S4 method for signature 'ImageSet'
extract_nearest_neighbor(x, min_num_cells = 10)
```

Arguments

x	IrisSpatialFeatures ImageSet object
...	Additional arguments
min_num_cells	Minimum number of cell that a coordinate needs to have in order to calculate the statistics (Default: 10)

Value

distance to nearest neighbor for each

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
extract_nearest_neighbor(dataset)
```

extract_proximity	<i>Run a proximity analysis on all samples. There are two modes this function can be run. In the first mode it uses the major and minor axes for each cell as provided by inform. It then averages half of those two axes and adds an uncertainty margin, which are used to provide an estimate on whether two cells are touching. This mode can be used to approximate the interaction analysis. The second mode uses a user specified distance to count the cells within the proximity of a given cell-type. With increasing distances usually cells fall into the proximity of multiple cells of a given type so the function allows the restriction of only counting the cell only once.</i>
-------------------	--

Description

Run a proximity analysis on all samples. There are two modes this function can be run. In the first mode it uses the major and minor axes for each cell as provided by inform. It then averages half of those two axes and adds an uncertainty margin, which are used to provide an estimate on whether two cells are touching. This mode can be used to approximate the interaction analysis. The second mode uses a user specified distance to count the cells within the proximity of a given cell-type. With increasing distances usually cells fall into the proximity of multiple cells of a given type so the function allows the restriction of only counting the cell only once.

Usage

```
extract_proximity(x, ...)
```

```
## S4 method for signature 'ImageSet'
extract_proximity(x, radii = c("Entire.Cell.Major.Axis",
  "Entire.Cell.Minor.Axis"), uncertainty_margin = 1, only_closest = FALSE)
```

Arguments

x	IrisSpatialFeatures ImageSet object
...	Additional arguments
radii	The size of the radius in pixels or the default, which are the inForm output columns that indicated the minor and major axis of each cell.
uncertainty_margin	Only for the approximation of the interaction analysis, where it indicates how many pixels further should be search to find a touching cell (default: 1).
only_closest	For the proximity analysis a target cell can be in the vicinity of multiple source cells, so the counts are artificially inflated. E.g. a CD8 PD1+ T-cell is within <50 pixels of 30 HRS cells, this cell should only be counted for the closes HRS cell. (default: FALSE)

Value

The proximal events for each cell

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
extract_proximity(dataset)
```

extract_ROI	<i>Method that reduces the current dataset to a specific region of interest, discarding all cell coordinates outside of that region</i>
-------------	---

Description

Method that reduces the current dataset to a specific region of interest, discarding all cell coordinates outside of that region

Usage

```
extract_ROI(x, ...)  
  
## S4 method for signature 'ImageSet'  
extract_ROI(x, ROI = "invasive_margin")
```

Arguments

x	IrisSpatialFeatures ImageSet object
...	Additional arguments
ROI	Region of interest (default: 'invasive_margin')

Value

IrisSpatialFeatures ImageSet object

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
im_area <- extract_ROI(dataset,ROI='invasive_margin')
```

feature_selection *Function to extract all numeric features*

Description

Function to extract all numeric features

Usage

```
feature_selection(dat, lab)
```

Arguments

dat	A data matrix with features as rows and samples as columns
lab	Label annotation that contains 2 classes, which corresponds to the samples in the column

Value

t-test and wilcox test between the 2 classes

Examples

```
dat <- cbind(matrix(runif(400),ncol=10),matrix(runif(400)+0.2,ncol=10))
lab <- c(rep('classA',10),rep('classB',10))
rownames(dat) <- paste0('F',1:nrow(dat))
feature_selection(dat,lab)
```

get_all_interactions *Get all interactions between all cell-types*

Description

Get all interactions between all cell-types

Usage

```
get_all_interactions(x, ...)
```

S4 method for signature 'ImageSet'

```
get_all_interactions(x)
```

Arguments

x	An IrisSpatialFeatures ImageSet object.
...	Additional arguments.

Value

For each cell-type return interactions

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_interactions(dataset)
get_all_interactions(dataset)
```

```
get_all_nearest_neighbors
```

Get the nearest neighbor for each cell-type

Description

Get the nearest neighbor for each cell-type

Usage

```
get_all_nearest_neighbors(x, ...)
```

```
## S4 method for signature 'ImageSet'
get_all_nearest_neighbors(x)
```

Arguments

x	An IrisSpatialFeatures ImageSet object
...	Additional arguments

Value

Nearest neighbor for each cell-type

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_nearest_neighbor(dataset)
get_all_nearest_neighbors(dataset)
```

get_all_proximities *Get all proximity data for all cell-types in a sample*

Description

Get all proximity data for all cell-types in a sample

Usage

```
get_all_proximities(x, ...)
```

```
## S4 method for signature 'ImageSet'
```

```
get_all_proximities(x)
```

Arguments

x An IrisSpatialFeatures ImageSet object
 ... Additional arguments

Value

all proximity data for cell-types

Examples

```
#loading pre-read dataset
```

```
dataset <- IrisSpatialFeatures_data
```

```
get_all_proximities(dataset)
```

get_counts_per_mm2 *Get all the counts on a per mm2 basis*

Description

Get all the counts on a per mm2 basis

Usage

```
get_counts_per_mm2(x, ...)
```

```
## S4 method for signature 'ImageSet'
```

```
get_counts_per_mm2(x, digits = 2, blank = FALSE)
```

Arguments

x An IrisSpatialFeatures ImageSet object
 ... Additional arguments
 digits Number of digits that are shown in the output (default: 2)
 blank (default: FALSE)

Value

counts per mm2 per sample, collapsing each coordinate and returning mean and standard error

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
get_counts_per_mm2(dataset)
```

```
get_counts_per_mm2_noncollapsed
```

Get all the counts on a per mm2 basis non-collapsed

Description

Get all the counts on a per mm2 basis non-collapsed

Usage

```
get_counts_per_mm2_noncollapsed(x, ...)
```

S4 method for signature 'ImageSet'

```
get_counts_per_mm2_noncollapsed(x)
```

Arguments

x	IrisSpatialFeatures ImageSet object.
...	Additional arguments

Value

IrisSpatialFeatures ImageSet object.

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
get_counts_per_mm2_noncollapsed(dataset)
```

get_count_ratios *Get ratio of counts between two markers*

Description

Get ratio of counts between two markers

Usage

```
get_count_ratios(x, ...)  
  
## S4 method for signature 'ImageSet'  
get_count_ratios(x, marker1, marker2, digits = 2)
```

Arguments

x	An IrisSpatialFeatures object
...	Additional arguments.
marker1	First cell-type.
marker2	Second cell-type.
digits	Number of digits that should be shown in the the results. (Default: 2)

Value

Count ratio between two markers

Examples

```
#loading pre-read dataset  
dataset <- IrisSpatialFeatures_data  
get_count_ratios(dataset, 'SOX10+ PDL1-', 'SOX10+ PDL1+')
```

get_interactions *Get interactions for a specific marker*

Description

Get interactions for a specific marker

Usage

```
get_interactions(x, ...)  
  
## S4 method for signature 'ImageSet'  
get_interactions(x, marker, normalize = TRUE)
```

Arguments

x	An IrisSpatialFeatures ImageSet object
...	Additional arguments.
marker	Cell-type for which the interactions should be pulled
normalize	Flag to indicated whether to normalize each sample so all interactions sum up to 1 (Default: 1)

Value

interactions for a specific marker

Examples

```
#' #loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_interactions(dataset)
get_interactions(dataset, 'CD8+ PD1+')
```

get_nearest_neighbors *Get the nearest neighbor for a specified cell-type*

Description

Get the nearest neighbor for a specified cell-type

Usage

```
get_nearest_neighbors(x, ...)

## S4 method for signature 'ImageSet'
get_nearest_neighbors(x, marker)
```

Arguments

x	An IrisSpatialFeatures ImageSet object
...	Additional arguments
marker	Cell type for which the nearest neighbor should be calculated

Value

nearest neighbors for the specified cell-type

Examples

```
#' #loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_nearest_neighbor(dataset, min_num_cells=2)
get_nearest_neighbors(dataset, "SOX10+ PDL1+")
```

get_proximities	<i>Get proximity data for a given cell-type</i>
-----------------	---

Description

Get proximity data for a given cell-type

Usage

```
get_proximities(x, ...)

## S4 method for signature 'ImageSet'
get_proximities(x, marker, normalize = TRUE)
```

Arguments

x	An IrisSpatialFeatures ImageSet object.
...	Additional arguments.
marker	Cell type for which the proximity data should be extracted.
normalize	Flag indicating whether the populations should be normalized so that the sum of all is 1 (default: TRUE).

Value

proximities for a specific cell-type

ImageSet-class	<i>Class to represent an imaging dataset.</i>
----------------	---

Description

Class to represent an imaging dataset.

Slots

samples	A list of samples each containing multiple coordinates.
counts	A list of counts of different cell types for each coordinate in each sample.
nearest_neighbors	A list of mean and std of nearest neighbor distances for each samples.
interactions	A list of interaction information for each sample.
proximity	A list of mean and std of nearest neighbor distances for each sample.
microns_per_pixel	Scalar value that indicates the length of a pixel in micrometers.
markers	A vector of strings indicating all different cell types considered.
invasive_margin_in_px	The width of the invasive margin in pixels
readMasks	Flag indicating whether the "_Tumor.tif" and "_Invasive_Margin.tif" should be read (default: True)
ignore_scoring	Flag indicating whether the scoring file should be ignored (default: FALSE)

interaction_maps *Plot interaction maps for all samples*

Description

Plot interaction maps for all samples

Usage

```
interaction_maps(x, ...)

## S4 method for signature 'ImageSet'
interaction_maps(x, int_markers, int_marker_cols,
  silent_markers = c(), silent_col = c(), outline_transparency = 0.9,
  use_dapi = FALSE, outdir = "interaction_maps", useMask = NULL,
  format = ".png")
```

Arguments

x	An IrisSpatialFeatures ImageSet object
...	Additional arguments.
int_markers	Cell-types that should be considered. If two cells from different cell-types interact they are filled in, if a cell is not interacting it is just outlined.
int_marker_cols	Colors for the cell-types
silent_markers	Cell-types that should only be outlined (Default: c())
silent_col	Colors for silent markers (Default: c())
outline_transparency	Dimming factor for the outlines cells(Default: 0.9)
use_dapi	Use the DAPI channel as a background (Default: FALSE)
outdir	Output directory (Default: './interaction_maps')
useMask	(Default: NULL)
format	Output format of the images. Can be '.png' or '.tiff' (Default: '.png')

Value

plot of interactions for all samples

Examples

```
## #loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_interactions(dataset)
get_interactions(dataset, 'CD8+ PD1+')
int_markers <- c('CD8+ PD1+', 'SOX10+ PDL1+')
int_marker_cols <- c('#dd1c77', '#99d8c9')
silent_markers <- c('CD8+ PD1-')
silent_col=c('yellow')
p <- interaction_maps(dataset, int_markers, int_marker_cols, silent_markers,
  silent_col)
```

IrisSpatialFeatures *IrisSpatialFeatures: A package to extract spatial features based on multiplex IF images*

Description

Iris reads the output of the PerkinElmer inForm software and calculates a variety of spatial statistics. In addition to simple counts, it can derive average nearest neighbors for each cell-type and interaction summary profiles for each celltype. These statistics are derived across images, both overall and regions of interest as defined by user defined masks.

Author(s)

Daniel Gusenleitner <Daniel_Gusenleitner@dfci.harvard.edu>

IrisSpatialFeatures_data
Example IrisSpatialFeatures dataset

Description

A dataset that contains two small samples, with 5 imaging coordinates per samples

Usage

IrisSpatialFeatures_data

Format

IrisSpatialFeatures dataset

neighbor_ray_plot *Plot nearest neighbor ray plots for each samples*

Description

Plot nearest neighbor ray plots for each samples

Usage

```
neighbor_ray_plot(x, ...)

## S4 method for signature 'ImageSet'
neighbor_ray_plot(x, from_type, to_type,
  from_col = "#EE7600", to_col = "#028482", format = ".pdf",
  plot_dir = "./", lineColor = "#666666", use_pixel = FALSE, height = 7,
  width = 10)
```

Arguments

x	An IrisSpatialFeatures ImageSet object
...	Additional arguments.
from_type	Cell type from which the rays are drawn
to_type	Cell type to which the rays are drawn
from_col	Color for the 'from' cell-type (Default: '#EE7600')
to_col	Color for the 'to' cell-type (Default: '#028482')
format	Format of the output file, can be '.pdf' or '.png' (Default: '.pdf')
plot_dir	Directory in which the images are written (Default: './')
lineColor	Color of the rays (Default: '#666666')
use_pixel	use pixels instead of micrometer for distance measurements (default: FALSE)
height	Height of the pdf. (Default: 7)
width	Width of the pdf. (Default: 10)

Value

nearest neighbor ray plots

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_nearest_neighbor(dataset)
get_nearest_neighbors(dataset, "SOX10+ PDL1+")
plot_dir <- file.path('./ray_plots')
if (!file.exists(plot_dir)){
  dir.create(file.path(plot_dir))
}
```

overview_plot

Plot all coordinates in a given dataset

Description

Plot all coordinates in a given dataset

Usage

```
overview_plot(x, ...)
```

```
## S4 method for signature 'ImageSet'
overview_plot(x, outdir = "./", palette = NULL,
  type = "pdf", width = 10, height = 7)
```

Arguments

x	Sample ImageSet object of the IrisSpatialFeatures package.
...	Additional arguments
outdir	Output directory (default: './')
palette	Color palette used for the different cell-types. (default: NULL)
type	File format for the plots. Can be either 'pdf' or 'png'. (default: 'pdf')
width	Width of the plot in inches for pdf and pixels for png. (default: 10)
height	Height of the plot in inches for pdf and pixels for png. (default: 7)

Value

plot of all coordinates

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
overview_plot(dataset)
```

plot,ImageSet-method *Plot all frames of an IrisSpatialFeatures ImageSet object.*

Description

Plot all frames of an IrisSpatialFeatures ImageSet object.

Usage

```
## S4 method for signature 'ImageSet'
plot(x)
```

Arguments

x	Iris ImageSet object
---	----------------------

Value

A plot

Examples

```
dataset <- IrisSpatialFeatures_data
plot(dataset)
```

plot_interactions *Interaction summary plot for all cell-types and all samples in a dataset*

Description

Interaction summary plot for all cell-types and all samples in a dataset

Usage

```
plot_interactions(x, ...)

## S4 method for signature 'ImageSet'
plot_interactions(x, label, ordering = NULL,
  normalize = TRUE, palette = NULL, celltype_order = NULL,
  xlim_fix = 13, topbar_cols = "darkgrey")
```

Arguments

x	IrisSpatialFeatures ImageSet object to be plotted
...	Additional arguments
label	The cell type the interaction profile should be plotted for
ordering	Ordering of the samples (default: NULL)
normalize	Normalize the interactions with a given cell-type, so they sum up to 1 (default: TRUE)
palette	Color palette for all the cell-types (default: Spectral scheme from RColorbrewer)
celltype_order	Order in which the cell-types are displayed. (default: Alphabetically)
xlim_fix	Whitespace on the right side so the legend can be displayed clearly. (default: 13)
topbar_cols	Color of the barplots that are shown on top. (default: 'darkgrey')

Value

plot of all cell-types and samples interactions

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_interactions(dataset)
plot_interactions(dataset, 'SOX10+ PDL1+', xlim_fix=3)
```

`plot_nearest_neighbor` *Plot average nearest neighbor barplots for two cell types. This measurement is not symmetric, so if 'from' and 'to' are switched it will result in different results. For the 'to' parameter this function allows a cell-type without '+' or '-' in the end. Indicating that the distances from the first cell-type should be calculated against both '+/-' and a paired t-test should be calculated. For example we want to calculate the average distance between SOX10 PDL1+ melanoma cells against both CD8 PD1+ and CD8 PD1- cells, the 'CD8 PD1' would be specified as 'to' parameter, 2 distances would be calculated for each sample and a two-sided paired t-test calculated to test for significant differences.*

Description

Plot average nearest neighbor barplots for two cell types. This measurement is not symmetric, so if 'from' and 'to' are switched it will result in different results. For the 'to' parameter this function allows a cell-type without '+' or '-' in the end. Indicating that the distances from the first cell-type should be calculated against both '+/-' and a paired t-test should be calculated. For example we want to calculate the average distance between SOX10 PDL1+ melanoma cells against both CD8 PD1+ and CD8 PD1- cells, the 'CD8 PD1' would be specified as 'to' parameter, 2 distances would be calculated for each sample and a two-sided paired t-test calculated to test for significant differences.

Usage

```
plot_nearest_neighbor(x, ...)

## S4 method for signature 'ImageSet'
plot_nearest_neighbor(x, from, to, ttest = TRUE,
  transposed = FALSE, remove_NAs = FALSE, use_pixel = FALSE)
```

Arguments

<code>x</code>	IrisSpatialFeatures ImageSet object.
<code>...</code>	Additional arguments.
<code>from</code>	Cell-type from which the nearest neighbor is calculated.
<code>to</code>	Cell-type to which the nearest neighbor is calculated.
<code>ttest</code>	Flag indicating whether a paired t-test should be calculated. (default: TRUE)
<code>transposed</code>	Switches 'from' and 'to' cell-type. This way the (default: FALSE)
<code>remove_NAs</code>	dont plot samples with less than min cells
<code>use_pixel</code>	show the distances in pixels or micrometers (default: FALSE)

Value

plot average nearest neighbor barplots for two cell types

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_nearest_neighbor(dataset)
p <- plot_nearest_neighbor(dataset,'CD8+ PD1+', 'SOX10+ PDL1')
```

plot_proximities *Plot proximity analysis data*

Description

Plot proximity analysis data

Usage

```
plot_proximities(x, ...)

## S4 method for signature 'ImageSet'
plot_proximities(x, label, ordering = NULL,
  normalize = TRUE, palette = NULL, celltype_order = NULL,
  xlim_fix = 13, topbar_cols = "darkgrey")
```

Arguments

x	An IrisSpatialFeatures ImageSet object
...	Additional parameters.
label	Cell-type for which the proximit profile is plotted
ordering	Ordering of the samples (Default: NULL)
normalize	Flag, should the populations of different cell-types sum up to one in each sample? (Default: TRUE)
palette	Color palette, by default it uses Spectral from RColorbrewer (Default:NULL)
celltype_order	Ordering of the cell-type. (Default: NULL)
xlim_fix	Space on the right side to show the legend (Default: 13)
topbar_cols	Color of the barplots on top (Default: 'darkgrey')

Value

plot proximity analysis

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_proximity(dataset,only_closest=TRUE,radii=25)
p <- plot_proximities(dataset,"SOX10+ PDL1-",xlim_fix=3)
```

raw_data-class	<i>Class to represent the raw data that is output by inForm</i>
----------------	---

Description

Class to represent the raw data that is output by inForm

Slots

data Raw coordinate data sheet
summary Inform summary (optional)
score Score table that indicates intensity cutoff for certain markers
mem_seg_map Membrane map as output by inForm
nuc_seg_map Nuclear map as output by inForm (optional)
component_tiff component tiff as output by inForm (optional)

rayplot_single_coordinate	<i>Plot nearest neighbor ray plots for a single coordinate</i>
---------------------------	--

Description

Plot nearest neighbor ray plots for a single coordinate

Usage

```
rayplot_single_coordinate(x, from_type, to_type, samp_name = "",
  from_col = "#EE7600", to_col = "#028482", lineColor = "#666666",
  use_pixel = FALSE, microns_per_pixel = 0.496)

## S4 method for signature 'Coordinate'
rayplot_single_coordinate(x, from_type, to_type,
  samp_name = "", from_col = "#EE7600", to_col = "#028482",
  lineColor = "#666666", use_pixel = FALSE, microns_per_pixel = 0.496)
```

Arguments

x	An Coordinate object
from_type	Cell type from which the rays are drawn
to_type	Cell type to which the rays are drawn
samp_name	Name of the sample
from_col	Color for the 'from' cell-type (Default: '#EE7600')
to_col	Color for the 'to' cell-type (Default: '#028482')
lineColor	Color for the line (Default: '#666666')
use_pixel	Express units as pixels (Default: FALSE)
microns_per_pixel	Conversion (Default: 0.496)

Value

a plot

Examples

```
#loading pre-read dataset
dataset <- IrisSpatialFeatures_data
dataset <- extract_nearest_neighbor(dataset)
rayplot_single_coordinate(x = dataset@samples[[1]]@coordinates[[1]],
                          samp_name = dataset@samples[[1]]@sample_name,
                          from_type = "SOX10+ PDL1+",
                          to_type = "CD8+ PD1+")
```

read_raw	<i>Read inForm output and store it in an IrisSpatialFeatures ImageSet object.</i>
----------	---

Description

Read inForm output and store it in an IrisSpatialFeatures ImageSet object.

Usage

```
read_raw(path, label_fix = list(), format = "Vectra", dir_filter = "",
         read_nuc_seg_map = FALSE, MicronsPerPixel = 0.496,
         invasive_margin_in_px = 100, readMasks = TRUE, ignore_scoring = FALSE,
         read_only_relevant_markers = TRUE)
```

```
## S4 method for signature 'character'
read_raw(path, label_fix = list(), format = "Vectra",
         dir_filter = "", read_nuc_seg_map = FALSE, MicronsPerPixel = 0.496,
         invasive_margin_in_px = 100, readMasks = TRUE, ignore_scoring = FALSE,
         read_only_relevant_markers = TRUE)
```

Arguments

path	Directory that contains the raw files
label_fix	List of length 2 character vector that is used to fix filenames.
format	Output format: Currently only "Vectra" and "Mantra" are supported.
dir_filter	Filter to select only certain directory names.
read_nuc_seg_map	Flag indicating whether the nuclear map should be read.
MicronsPerPixel	Length of one pixel. Default: 0.496, corresponding to a 20x Mantra/Vectra images
invasive_margin_in_px	The width of the invasive margin in pixels
readMasks	Flag indicating whether the "_Tumor.tif" and "_Invasive_Margin.tif" should be read (default: True)

`ignore_scoring` Flag indicating whether the scoring file should be ignored (default: False)
`read_only_relevant_markers` Flag that indicates whether all read inform output should be kept or only the relevant markers

Value

IrisSpatialFeatures ImageSet object.

Examples

```
raw_data <- read_raw(path=system.file("extdata", package = "IrisSpatialFeatures"),
  format='Mantra')
```

Sample-class	<i>An S3 class to represent a single imaging sample with multiple coordinates.</i>
--------------	--

Description

An S3 class to represent a single imaging sample with multiple coordinates.

Slots

`coordinates` A list of coordinate objects that contain all of the raw and coordinate data.
`sample_name` Name of the contained sample.

<code>threshold_dataset</code>	<i>This function reads the manually determined thresholds of certain markers (e.g. PD1, PD-L1) and splits selected celltypes into marker+ and marker- celltypes.</i>
--------------------------------	--

Description

This function reads the manually determined thresholds of certain markers (e.g. PD1, PD-L1) and splits selected celltypes into marker+ and marker- celltypes.

Usage

```
threshold_dataset(image_set, marker, marker_name, base = NULL,
  pheno_name = "Phenotype", remove_blanks = TRUE)
```

```
## S4 method for signature 'ImageSet'
threshold_dataset(image_set, marker, marker_name,
  base = NULL, pheno_name = "Phenotype", remove_blanks = TRUE)
```

Arguments

<code>image_set</code>	IrisSpatialFeatures ImageSet object.
<code>marker</code>	Name of the marker used in the score file.
<code>marker_name</code>	corresponding name, which should be appended at the selected cell types.
<code>base</code>	Vector of cell types for which the marker should be used.
<code>pheno_name</code>	Name of the phenotype column to be used. (Default from inForm is "Phenotype")
<code>remove_blanks</code>	Flag that indicates whether or not not called cells are to be removed. (Default: TRUE)

Value

IrisSpatialFeatures ImageSet object.

Examples

```
dataset <- read_raw(path=system.file("extdata", package = "IrisSpatialFeatures"),
                    format='Mantra')
dataset <- threshold_dataset(dataset,
                             marker='PD-Ligand-1 (Opal 690)',
                             marker_name='PDL1',
                             base=c('SOX10+'))
dataset <- threshold_dataset(dataset,
                             marker='PD-1 (Opal 540)',
                             marker_name='PD1',
                             base=c('CD8+', 'OTHER'))
```

Index

*Topic **datasets**

- IrisSpatialFeatures_data, [17](#)
- as.data.frame, ImageSet-method, [2](#)
- collapse_markers, [3](#)
- collapse_markers, ANY, ANY-method
(collapse_markers), [3](#)
- collapse_markers, ImageSet-method
(collapse_markers), [3](#)
- Coordinate (Coordinate-class), [4](#)
- Coordinate-class, [4](#)
- extract.nearest.neighbor, ANY, ANY-method
(extract_nearest_neighbor), [6](#)
- extract_features, [4](#)
- extract_features, ANY, ANY-method
(extract_features), [4](#)
- extract_features, ImageSet-method
(extract_features), [4](#)
- extract_interactions, [5](#)
- extract_interactions, ANY, ANY-method
(extract_interactions), [5](#)
- extract_interactions, ImageSet-method
(extract_interactions), [5](#)
- extract_mask, [6](#)
- extract_nearest_neighbor, [6](#)
- extract_nearest_neighbor, ImageSet-method
(extract_nearest_neighbor), [6](#)
- extract_proximity, [7](#)
- extract_proximity, ImageSet-method
(extract_proximity), [7](#)
- extract_ROI, [8](#)
- extract_ROI, ANY, ANY-method
(extract_ROI), [8](#)
- extract_ROI, ImageSet-method
(extract_ROI), [8](#)
- feature_selection, [9](#)
- get_all_interactions, [9](#)
- get_all_interactions, ANY, ANY-method
(get_all_interactions), [9](#)
- get_all_interactions, ImageSet-method
(get_all_interactions), [9](#)
- get_all_nearest_neighbors, [10](#)
- get_all_nearest_neighbors, ANY, ANY-method
(get_all_nearest_neighbors), [10](#)
- get_all_nearest_neighbors, ImageSet-method
(get_all_nearest_neighbors), [10](#)
- get_all_proximities, [11](#)
- get_all_proximities, ANY, ANY-method
(get_all_proximities), [11](#)
- get_all_proximities, ImageSet-method
(get_all_proximities), [11](#)
- get_count_ratios, [13](#)
- get_count_ratios, ANY, ANY-method
(get_count_ratios), [13](#)
- get_count_ratios, ImageSet-method
(get_count_ratios), [13](#)
- get_counts_per_mm2, [11](#)
- get_counts_per_mm2, ImageSet-method
(get_counts_per_mm2), [11](#)
- get_counts_per_mm2_noncollapsed, [12](#)
- get_counts_per_mm2_noncollapsed, ANY, ANY-method
(get_counts_per_mm2_noncollapsed),
[12](#)
- get_counts_per_mm2_noncollapsed, ImageSet-method
(get_counts_per_mm2_noncollapsed),
[12](#)
- get_interactions, [13](#)
- get_interactions, ANY, ANY-method
(get_interactions), [13](#)
- get_interactions, ImageSet-method
(get_interactions), [13](#)
- get_nearest_neighbors, [14](#)
- get_nearest_neighbors, ANY, ANY-method
(get_nearest_neighbors), [14](#)
- get_nearest_neighbors, ImageSet-method
(get_nearest_neighbors), [14](#)
- get_proximities, [15](#)
- get_proximities, ANY, ANY-method
(get_proximities), [15](#)
- get_proximities, ImageSet-method
(get_proximities), [15](#)
- ImageSet (ImageSet-class), [15](#)
- ImageSet-class, [15](#)
- interaction_maps, [16](#)

- interaction_maps, ANY, ANY-method
 - (interaction_maps), 16
- interaction_maps, ImageSet-method
 - (interaction_maps), 16
- IrisSpatialFeatures, 17
- IrisSpatialFeatures-package
 - (IrisSpatialFeatures), 17
- IrisSpatialFeatures_data, 17

- neighbor.ray.plot, ANY, ANY-method
 - (neighbor_ray_plot), 17
- neighbor_ray_plot, 17
- neighbor_ray_plot, ImageSet-method
 - (neighbor_ray_plot), 17

- overview_plot, 18
- overview_plot, ANY, ANY-method
 - (overview_plot), 18
- overview_plot, ImageSet-method
 - (overview_plot), 18

- plot, ImageSet-method, 19
- plot_interactions, 20
- plot_interactions, ANY, ANY-method
 - (plot_interactions), 20
- plot_interactions, ImageSet-method
 - (plot_interactions), 20
- plot_nearest_neighbor, 21
- plot_nearest_neighbor, ANY, ANY-method
 - (plot_nearest_neighbor), 21
- plot_nearest_neighbor, ImageSet-method
 - (plot_nearest_neighbor), 21
- plot_proximities, 22
- plot_proximities, ANY, ANY-method
 - (plot_proximities), 22
- plot_proximities, ImageSet-method
 - (plot_proximities), 22

- raw_data (raw_data-class), 23
- raw_data-class, 23
- rayplot_single_coordinate, 23
- rayplot_single_coordinate, Coordinate-method
 - (rayplot_single_coordinate), 23
- read_raw, 24
- read_raw, ANY, ANY-method (read_raw), 24
- read_raw, character-method (read_raw), 24
- run_proximity, ANY, ANY-method
 - (extract_proximity), 7

- Sample (Sample-class), 25
- Sample-class, 25

- threshold_dataset, 25
- threshold_dataset, ANY, ANY-method
 - (threshold_dataset), 25
- threshold_dataset, ImageSet-method
 - (threshold_dataset), 25