Package ‘flowDensity’

April 25, 2017

Type   Package
Title  Sequential Flow Cytometry Data Gating
Version 1.10.0
Date 2016-06-01
Author M. Jafar Taghiyar, Mehrnoush Malek
Maintainer Mehrnoush Malek <mmalekes@bccrc.ca>
Description This package provides tools for automated sequential gating analogous to the manual gating strategy based on the density of the data.
Imports flowCore, graphics, car, gplots, RFOC, GEOmap, methods, grDevices
Depends R (>= 2.10.0), methods
License Artistic-2.0
biocViews Bioinformatics, FlowCytometry, CellBiology, Clustering, Cancer, FlowCytData, StemCells, DensityGating
LazyLoad yes
NeedsCompilation no

R topics documented:

CellPopulation-class ................................. 2
deGate ........................................... 3
flowDensity ........................................ 4
flowDensity-methods ................................ 5
getflowFrame ....................................... 6
nmRemove ........................................... 7
notSubFrame ......................................... 7
plotDens ........................................... 8

Index 10
CellPopulation-class  Class "CellPopulation"

Description
This class represents the output of `flowDensity(.)` function from flowDensity package.

Objects from the Class
Objects can be created by calls of the form `new("CellPopulation", ...).

Slots
- `flow.frame`: Object of class "flowFrame" representing the flow cytometry data of the cell population
- `proportion`: Object of class "numeric" representing proportion of the cell population with respect to its parent cell population
- `cell.count`: Object of class "numeric" representing cell count of the cell population
- `channels`: Object of class "character" representing channel names corresponding to the 2 dimensions where the cell population is extracted
- `position`: Object of class "logical" representing position of the cell population in the 2-dimensional space
- `gates`: Object of class "numeric" representing thresholds on each channel used to gate the cell population
- `filter`: Object of class "matrix" representing boundary of the cell population using a convex polygon
- `index`: Object of class "numeric" representing indices of the data points in the cell population with respect to its parent cell population

Methods
- `flowDensity` signature(obj = "CellPopulation", channels = "ANY", position = "logical", singlet.gate = "missing")
- `flowDensity` signature(obj = "CellPopulation", channels = "missing", position = "missing", singlet.gate = "logical")
- `getflowFrame` signature(obj = "CellPopulation"): ...
- `plot` signature(x = "flowFrame", y = "CellPopulation"): ....

Author(s)
Jafar Taghiyar <email: <jtaghiyar@bccrc.ca>

Examples
`showClass("CellPopulation")`
deGate

ID density gating method

Description

Find the best threshold for a single channel in flow cytometry data based on its density distribution.

Usage

deGate(flow.frame, channel, n.sd=1.5, use.percentile = FALSE, percentile = 0.95, use.upper=FALSE, upper = NA, talk=TRUE, alpha = 0.1, sd.threshold = FALSE, graphs = FALSE, all.cuts = FALSE, tinypeak.removal=1/25, adjust.dens=1)

Arguments

flow.frame  a 'FlowFrame' object.
channel  a channel’s name or its corresponding index in the 'flow.frame'.
n.sd  an integer coefficient for the standard deviation to determine the threshold based on the standard deviation if 'sd.threshold' is TRUE.
use.percentile  if TRUE, forces to return the 'percentile'th threshold.
percentile  a value in [0,1] that is used as the percentile. The default value is 0.95.
upper  if TRUE, finds the change in the slope at the tail of the density curve, if FALSE, finds it at the head. Default value is set to 'NA'.
use.upper  Logical. If TRUE, forces to return the inflection point based on the first (last) peak if upper=F (upper=T). Default value is set to 'FALSE'.
talk  Logical. If TRUE, Prints a message if only one peak is found, or when inflection point is used to set the gates.
alpha  a value in [0,1) specifying the significance of change in the slope being detected. This is by default 0.1, and typically need not be changed.
sd.threshold  if TRUE, uses 'n.sd' times standard deviation as the threshold. Default value is set to 'FALSE'.
graphs  if TRUE, generates density distribution plot plus its corresponding threshold.
all.cuts  if TRUE, returns all the identified cutoff points, i.e. potential thresholds for that channel. Default value is set to 'FALSE'.
tinypeak.removal  A number in [0,1] to exclude/include tiny peaks in density distribution.
adjust.dens  The smoothness of density in [0,Inf] to be used in density(.). The default value is 1 and should not be changed unless necessary

Value

an integer value (vector) of cutoff(s), i.e. threshold(s), on the specified channel

Author(s)

Jafar Taghiyar <<jtaghiyar@bccrc.ca>>
Examples

```r
data_dir <- system.file("extdata", package = "flowDensity")
load(list.files(pattern = "sampleFCS_1", data_dir, full = TRUE))
# Find the threshold for CD20
cd19.gate <- deGate(f, channel="PerCP-Cy5-5-A")
# Gate out the CD20- populations using the notSubFrame
plotDens(f, c("APC-H7-A", "PerCP-Cy5-5-A"))
abline(h=cd19.gate, lty=3, col=2)
```

flowDensity

Automated Sequential Gating Tool for Flow Cytometry

Description

flowDensity is an automated clustering algorithm which aims to emulate the current practice of manual sequential gating. It is designed to identify the predefined cell subsets based on the density distribution of the parent cell population by analyzing the peaks of the density curve.

When the density distribution has only one peak:

- the first argument that would be used is the percentile, the default value is 95th.
- If set to ‘NA’ then upper will be used when set to ‘FALSE/TRUE’.
- If both upper and percentiles are set to ‘NA’ sd.threshold is used when set to ‘TRUE’.
- If either all these arguments are ‘NA’, or one of them fails, the algorithm find the best cutoff based on the inflection points and by comparing the position of the peak and the mean of the population.

Usage

```r
flowDensity(obj, channels, position, singlet.gate, ...)
```

Arguments

- `obj` a `CellPopulation` or `flowFrame` object.
- `channels` a vector of two channel names or their corresponding indices.
- `position` a vector of two logical values specifying the position of the cell subset of interest on the 2D plot.
- `singlet.gate` if TRUE, singlet cell gate is derived and the corresponding cell population is returned.
- `...` This can be used to pass one of the following arguments:
  - ‘use.percentile’ if TRUE, returns the ‘percentile’th threshold.
  - ‘percentile’ a value in [0,1] that is used as the percentile if ‘use.percentile’ is TRUE.
  - ‘upper’ if ‘TRUE’, it finds the change in the slope after the peak with index ‘peak.ind’.
  - ‘use.upper’ if ‘TRUE’, forces to return the inflection point based on the first (last) peak if upper=F (upper=T)
  - ‘sd.threshold’ if ‘TRUE’, it uses ‘n.sd’ times standard deviation for gating.
• 'n.sd' an integer that is multiplied to the standard deviation to determine the place of threshold if 'sd.threshold' is 'TRUE'.
• 'use.control' if TRUE, it finds the threshold using a matched control population and uses it for gating.
• 'control' a `flowFrame` or `CellPopulation` object used for calculating the gating threshold when 'use.control' is set to TRUE. If a control population is used, the other arguments ('upper', 'percentile', etc.) are applied to the control data when finding the threshold (i.e. not to 'obj').
• 'alpha' a value in [0,1) specifying the significance of change in the slope which would be detected. This is by default 0.1, and typically need not be changed.
• 'debris.gate' if TRUE, it would try to remove the debris and gate the lymphocyte cells.
• 'avg' if TRUE, it uses the mean of all identified cutoff points as a threshold for gating.
• 'ellip.gate' if TRUE, it fits an ellipse on the data as a gate, otherwise the rectangle gating results are returned
• 'remove.neg' if TRUE, it removes negative values when removing margin events of scatter channels.
• 'scale' a value in [0,1) that scales the size of ellipse to fit if 'ellip.gate' is TRUE
• 'graphs' if TRUE, the ellipse is added to the current plot of the output of the 'flowDensity', otherwise a new plot is drawn and the ellipse is added on that.

Value
A CellPopulation object

Author(s)
Jafar Taghiyar <<jtaghiyar@bccrc.ca>> Mehrnoush Malek <<mmalekes@bccrc.ca>>

Examples

data_dir <- system.file("extdata", package = "flowDensity")
load(list.files(pattern = 'sampleFCS_1', data_dir, full = TRUE))
lymph <- flowDensity(obj=f, channels=c("FSC-A", "SSC-A"),
                     position=c(TRUE, FALSE), upper= c(NA, TRUE), debris.gate=c(TRUE, FALSE))
slotNames(lymph)
getflowFrame

Methods

signature(obj = "CellPopulation", channels = "ANY", position = "logical", singlet.gate = "missing")

signature(obj = "CellPopulation", channels = "missing", position = "missing", singlet.gate = "logical")

signature(obj = "flowFrame", channels = "ANY", position = "logical", singlet.gate = "missing")

---

getflowFrame 'CellPopulation' class accessor.

Description

an accessor for 'CellPopulation' class to get its 'FlowFrame' object. This will remove all the NA values in the frame.

Usage

getflowFrame(obj)

Arguments

obj a 'CellPopulation' object.

Value

a 'FlowFrame' object.

Author(s)

Jafar Taghiyar <jtaghiyar@bccrc.ca>

Examples

data_dir <- system.file("extdata", package = "flowDensity")
load(list.files(pattern = "sampleFCS_1", data_dir, full = TRUE))
lymph <- flowDensity(obj=f, channels=c("FSC-A", "SSC-A"),
position=c(TRUE, FALSE), upper= c(NA, TRUE), debris.gate=c(TRUE, FALSE))
f.lymph <- getflowFrame(lymph)
**nmRemove**

**Preprocessing helper function for flow cytometry data**

**Description**

Remove the margin events on the axes. Usually, these events are considered as debris or artifacts. This is specifically useful for 'FSC' and 'SSC' channels in a 'FlowFrame' object. However, any channel can be input as an argument.

**Usage**

```
nmRemove(flow.frame, channels, neg=FALSE, talk=FALSE)
```

**Arguments**

- `flow.frame`: a 'FlowFrame' object.
- `channels`: a vector of channel names or their corresponding indices.
- `neg`: if TRUE, negative events are also removed
- `talk`: if TRUE, it prints the margin event in each channel

**Value**

a 'FlowFrame' object.

**Author(s)**

Jafar Taghiyar <<jtaghiyar@bccrc.ca>> Mehrnoush Malek <<mmalekes@bccrc.ca>>

**Examples**

```
data_dir <- system.file("extdata", package = "flowDensity")
load(list.files(pattern = '/quotesingle.Var sampleFCS_2/quotesingle.Var', data_dir, full = TRUE))
# Removing margin events of FSC-A and SSC-A channels
no.margin <- nmRemove(f2, c("FSC-A","SSC-A"), talk=TRUE)
plotDens(f2, c("FSC-A","SSC-A"))
plotDens(no.margin, c("FSC-A","SSC-A"))
```

**notSubFrame**

**Removing a subset of a FlowFrame object**

**Description**

Remove a subset of a FlowFrame object specified by gates from the flowDensity method. It comes in handy when one needs the complement of a cell population in the input flow cytometry data.

**Usage**

```
notSubFrame(flow.frame, channels, position, gates, filter)
```
### plotDens

Plot flow cytometry data with density-based colors.

#### Description

Generate a scatter dot plot with colors based on the distribution of the density of the provided channels.

#### Usage

```r
plotDens(flow.frame, channels, col, main, xlab, ylab, pch=".", s=FALSE, outdir, file.name, ...)
```

#### Arguments

- `flow.frame`: a 'FlowFrame' object.
- `channels`: a vector of two channel names or their corresponding indices in the 'flow.frame'.
- `s`: logical variable, if TRUE the output graph is saved.
- `outdir`: an optional string value specifying the output directory for saving the graph if 's' is TRUE.

#### Examples

```r
data_dir <- system.file("extdata", package = "flowDensity")
load(list.files(pattern = "Var", data_dir, full = TRUE))
#Find the threshold for CD20
cd20.gate <- deGate(f, channel="APC-H7-A")
# Gate out the CD20- populations using the notSubFrame
CD20.pos <- notSubFrame(f, channels=c("APC-H7-A", "PerCP-Cy5-5-A"), position=c(FALSE, NA), gates=c(cd20.gate, NA))
#Plot the CD20+ cells on same channels
plotDens(CD20.pos@flow.frame, c("APC-H7-A", "PerCP-Cy5-5-A"))
```
plotDens

- **file.name**  
an optional string value used as the name for the saved file.
- **col**  
A specification for the default plotting color: see `?par`.
- **main**  
an overall title for the plot: see `?plot`
- **xlab**  
a title for the x axis: see `?plot`
- **ylab**  
a title for the y axis: see `?plot`
- **pch**  
Either an integer specifying a symbol or a single character to be used as the default in plotting points: see `?par`.

... can be used to provide desired arguments for the plot() function used to plot the output results.

**Value**

a scatter dot plot with density-based colors.

**Author(s)**

Jafar Taghiyar <<jtaghiyar@bccrc.ca>>  
Mehrnoush Malek <<mmalekes@bccrc.ca>>

**Examples**

data_dir <- system.file("extdata", package = "flowDensity")
load(list.files(pattern = "/quotesingle.Var sampleFCS_1\quotesingle.Var", data_dir, full = TRUE))

#Plot CD3 vs. CD19 to see the distribution of cell populations and their density
plotDens(f,c("V450-A","PerCP-Cy5-5-A"))
Index

*Topic Automated gating
  plotDens, 8
*Topic FlowCytData
  plotDens, 8
*Topic \textasciitilde\textasciitilde
  other possible keyword(s)
  \textasciitilde\textasciitilde
  flowDensity-methods, 5
*Topic classes
  CellPopulation-class, 2
*Topic methods
  flowDensity-methods, 5
  .flowDensity (flowDensity), 4
  CellPopulation-class, 2
deGate, 3
  flowDensity, 4
  flowDensity, CellPopulation, ANY, logical, missing-method
  (flowDensity-methods), 5
  flowDensity, CellPopulation, missing, missing, logical-method
  (flowDensity-methods), 5
  flowDensity, flowFrame, ANY, logical, missing-method
  (flowDensity-methods), 5
  flowDensity-methods, 5
  getflowFrame, 6
  getflowFrame, CellPopulation-method
  (getflowFrame), 6
  getflowFrame, CellPopulation-method
  (CellPopulation-class), 2
  nmRemove, 7
  notSubFrame, 7
  plot, flowFrame,
  CellPopulation-method
  (plotDens), 8
  plot, flowFrame, CellPopulation-method
  (CellPopulation-class), 2
  plotDens, 8