

# Package ‘PAC’

April 10, 2016

**Type** Package

**Title** Partition-Assisted Clustering

**Version** 1.0.5

**Date** 2016-02-15

**Author** Dangna Li, Henry Li

**Maintainer** Dangna Li <ldangna@gmail.com>

**Description** Implements Partition-Assisted Clustering, which utilizes a collection of partition based nonparametric density estimation techniques to improve the robustness and accuracy of downstream clustering.  
The package also provides functions for effectively visualizing the clustering results.  
It is particularly useful for finding and visualizing subpopulations in single-cell data analysis.

**License** GPL-3

**Imports** Rcpp (>= 0.12.2),NMF,RJSONIO,data.table,dplyr

**Suggests** knitr

**VignetteBuilder** knitr

**LinkingTo** Rcpp

**RoxygenNote** 5.0.1

**NeedsCompilation** yes

**SystemRequirements** C++11

**Repository** CRAN

**Date/Publication** 2016-04-10 16:25:22

## R topics documented:

aggregateData . . . . .	2
BSPLeaveCenter . . . . .	2
clusterNames . . . . .	3
clusterPropHeatmap . . . . .	3
filteredResults_JSON . . . . .	4
fmeasure . . . . .	5

makeList . . . . .	5
PAC . . . . .	6
packedCircleInput . . . . .	6
signalLevelHeatmap . . . . .	7

<b>Index</b>	<b>8</b>
--------------	----------

---

aggregateData	<i>Aggregates results from the clustering and merging step.</i>
---------------	---

---

### Description

Aggregates results from the clustering and merging step.

### Usage

```
aggregateData(dataInput, labelsInput)
```

### Arguments

dataInput	File name of processed data in tab delimited format. Rows are events and columns are features measured.
labelsInput	File name of output from clustering and merging step.

### Value

The aggregated data of dataInput, with average signal levels for all clusters and sample combinations.

---

BSPLeaveCenter	<i>Finds N Leaf centers in the data</i>
----------------	---

---

### Description

Finds N Leaf centers in the data

### Usage

```
BSPLeaveCenter(data, N = 40, method = "dsp")
```

### Arguments

data	a n x p data matrix
N	: number of leaves centers
method	: partition method, either "dsp(discrepancy based partition)", or "bsp(bayesian sequantial partition)"

**Value**

leafctr : N leaves centers

---

clusterNames	<i>Get the top features that differentiate the clusters based on the maximum cluster signal levels across all samples. The maximum signal levels of all features of each cluster are found, and then the maximum signal levels are normalized across clusters. Next, within each cluster, the ranks of the features are obtained based on the normalized signal levels. Top ranked features are used to annotate each cluster</i>
--------------	---

---

**Description**

Get the top features that differentiate the clusters based on the maximum cluster signal levels across all samples. The maximum signal levels of all features of each cluster are found, and then the maximum signal levels are normalized across clusters. Next, within each cluster, the ranks of the features are obtained based on the normalized signal levels. Top ranked features are used to annotate each cluster

**Usage**

```
clusterNames(aggregatedData, Num_TopFeatures = 3)
```

**Arguments**

aggregatedData Variable name of aggregated data.

Num\_TopFeatures

The number of top features in descending order of rank of importance.

**Value**

The annotation of clusters based on top features

---

clusterPropHeatmap	<i>Make heatmap illustrating the cluster proportions across multiple samples. The aggregated data are first padded to assign size of 0 to missing clusters in some samples. Next, the numbers of events in each cluster in each sample are obtained. These values are normalized across samples to find the cluster proportions by samples. The higher the cluster proportion in one sample, the more specific the cluster is to that sample.</i>
--------------------	---

---

**Description**

Make heatmap illustrating the cluster proportions across multiple samples. The aggregated data are first padded to assign size of 0 to missing clusters in some samples. Next, the numbers of events in each cluster in each sample are obtained. These values are normalized across samples to find the cluster proportions by samples. The higher the cluster proportion in one sample, the more specific the cluster is to that sample.

**Usage**

```
clusterPropHeatmap(data_agg, Colv_order)
```

**Arguments**

data_agg	Variable name of aggregated data.
Colv_order	Variable vector specifying the order of heatmap columns

**Value**

Plots cluster proportion heatmap plot and returns a matrix of the proportion values.

---

filteredResults\_JSON *Obtain the input to zoomable packed circles plot. The clusters are filtered by size.*

---

**Description**

Obtain the input to zoomable packed circles plot. The clusters are filtered by size.

**Usage**

```
filteredResults_JSON(toJSON_Input, CountFilter = 1000,  
  filename = "zoomablePackedCirclesInput.txt")
```

**Arguments**

toJSON_Input	Variable name of the JSON input file.
CountFilter	The minimum size of retained clusters in each sample. Set to 0 for no filtering.
filename	The filename for zoomable packed circles plot.

**Value**

The JSON format input to D3 zoomable packed circles plot.

---

fmeasure	<i>Compute the F measure between the ground truth and the estimated label</i>
----------	---

---

**Description**

Compute the F measure between the ground truth and the estimated label

**Usage**

```
fmeasure(g, t)
```

**Arguments**

g : the ground truth  
t : estimated labels

**Value**

f : the F measure

---

makeList	<i>Helper function to obtain a JSON-friendly format of a matrix formatted object</i>
----------	--

---

**Description**

Helper function to obtain a JSON-friendly format of a matrix formatted object

**Usage**

```
makeList(x)
```

**Arguments**

x : A matrix.

**Value**

A list for generating JSON format output.

---

PAC	<i>PAC (Partition Assisted Clustering)</i>
-----	--

---

### Description

PAC (Partition Assisted Clustering)

### Usage

```
PAC(data, K, maxlevel = 40, method = "dsp", max.iter = 50)
```

### Arguments

<code>data</code>	: a $n \times p$ data matrix
<code>K</code>	: number of final clusters in the output
<code>maxlevel</code>	: the maximum level of the partition
<code>method</code>	: partition method, either "dsp(discrepancy based partition)", or "bsp(bayesian sequential partition)"
<code>max.iter</code>	: maximum iteration for the kmeans step

### Value

`y` : cluter labels for the input

---

<code>packedCircleInput</code>	<i>Make packed circles plot illustrating the cluster proportions across multiple samples.</i>
--------------------------------	---

---

### Description

Make packed circles plot illustrating the cluster proportions across multiple samples.

### Usage

```
packedCircleInput(data_agg, cluster_names, CountFilter = 1000,  
  filename = "zoomablePackedCirclesInput.txt")
```

### Arguments

<code>data_agg</code>	Variable name of aggregated data.
<code>cluster_names</code>	The annotated cluster names
<code>CountFilter</code>	The filter threshold to remove clusters with event numbers below the threshold.
<code>filename</code>	Set to zoomablePackedCirclesInput.txt or the name of choice for the html that draws the packed circles.

**Value**

Saves the D3 packed circles plot input in the working directory.

---

signalLevelHeatmap	<i>Make heatmap illustrating the signal levels for clusters. For multiple samples, signal levels differ for the same cluster in different samples. Signal levels can be plotted easily using built-in R functions such as mean, median, and max.</i>
--------------------	--

---

**Description**

Make heatmap illustrating the signal levels for clusters. For multiple samples, signal levels differ for the same cluster in different samples. Signal levels can be plotted easily using built-in R functions such as mean, median, and max.

**Usage**

```
signalLevelHeatmap(data_agg, signal_level = max)
```

**Arguments**

data_agg	Variable name of aggregated data.
signal_level	Function used to find signal levels for visualization.

**Value**

Saves the cluster signal level heatmap plot in the working directory.

# Index

[aggregateData](#), [2](#)

[BSPLeaveCenter](#), [2](#)

[clusterNames](#), [3](#)

[clusterPropHeatmap](#), [3](#)

[filteredResults\\_JSON](#), [4](#)

[fmeasure](#), [5](#)

[makeList](#), [5](#)

[PAC](#), [6](#)

[packedCircleInput](#), [6](#)

[signalLevelHeatmap](#), [7](#)