

Package ‘dynamicTreeCut’

August 3, 2008

Version 1.12

Date 2008-06-03

Title Dynamic Tree Cut

Author Peter Langfelder <Peter.Langfelder@gmail.com> and Bin Zhang
<binzhang.ucla@gmail.com>, with contributions from Steve Horvath
<SHorvath@mednet.ucla.edu>

Maintainer Peter Langfelder <Peter.Langfelder@gmail.com>

Depends R (>= 2.3.0), stats

ZipData no

License GPL (>= 2)

Description Contains methods for detection of clusters in hierarchical clustering dendrograms.

Title Dynamic Tree Cut

URL <http://www.genetics.ucla.edu/labs/horvath/CoexpressionNetwork/BranchCutting/>

R topics documented:

| | |
|----------------------------------|---|
| cutreeDynamic | 2 |
| cutreeDynamicTree | 4 |
| cutreeHybrid | 5 |
| dynamicTreeCut-package | 7 |
| indentSpaces | 8 |
| printFlush | 9 |

| | |
|--------------|-----------|
| Index | 10 |
|--------------|-----------|

cutreeDynamic *Adaptive branch pruning of hierarchical clustering dendrograms.*

Description

This wrapper provides a common access point for two methods of adaptive branch pruning of hierarchical clustering dendrograms.

Usage

```
cutreeDynamic(dendro, cutHeight = NULL, minClusterSize = 20, method = "hybrid",
             distM = NULL,
             deepSplit = (ifelse(method=="hybrid", 1, FALSE)),
             maxCoreScatter = NULL, minGap = NULL,
             maxAbsCoreScatter = NULL, minAbsGap = NULL, clusterTrim = 0,
             labelUnlabeled = NULL,
             pamStage = TRUE,
             useMedoids = FALSE, maxDistToLabel = cutHeight,
             respectSmallClusters = TRUE,
             verbose = 2, indent = 0)
```

Arguments

| | |
|----------------|--|
| dendro | A hierarchical clustering dendrogram such as one returned by <code>hclust</code> . |
| cutHeight | Maximum joining heights that will be considered. For <code>method=="tree"</code> it defaults to 0.99. For <code>method=="hybrid"</code> it defaults to 99 percentile and the maximum of the joining heights on the dendrogram. |
| minClusterSize | Minimum cluster size. |
| method | Chooses the method to use. Recognized values are "hybrid" and "tree". |
| distM | Only used for method "hybrid". The distance matrix used as input to <code>hclust</code> . If not given and <code>method == "hybrid"</code> , the function will issue a warning and default to <code>method = "tree"</code> . |
| deepSplit | For method "hybrid", can be either logical or integer in the range 0 to 4. For method "tree", must be logical. In both cases, provides a rough control over sensitivity to cluster splitting. The higher the value (or if TRUE), the more and smaller clusters will be produced. For the "hybrid" method, a finer control can be achieved via <code>maxCoreScatter</code> and <code>minGap</code> below. |
| maxCoreScatter | Only used for method "hybrid". Maximum scatter of the core for a branch to be a cluster, given as the fraction of <code>cutHeight</code> relative to the 5th percentile of joining heights. See Details. |
| minGap | Only used for method "hybrid". Minimum cluster gap given as the fraction of the difference between <code>cutHeight</code> and the 5th percentile of joining heights. |

| | |
|-----------------------------------|---|
| <code>maxAbsCoreScatter</code> | Only used for method "hybrid". Maximum scatter of the core for a branch to be a cluster given as absolute heights. If given, overrides <code>maxCoreScatter</code> . |
| <code>minAbsGap</code> | Only used for method "hybrid". Minimum cluster gap given as absolute height difference. If given, overrides <code>minGap</code> . |
| <code>clusterTrim</code> | Only used for method "hybrid". Specifies the fraction of the cluster height that will be trimmed before stage 2; all objects joining above the <code>1-ClusterTrim</code> fraction of the cluster height will be trimmed. If set to 0 or less, nothing will be trimmed. |
| <code>labelUnlabeled</code> | Deprecated, please use <code>pamStage</code> instead. Only used for method "hybrid". If TRUE, the second (PAM-like) stage will be performed. |
| <code>pamStage</code> | Only used for method "hybrid". If TRUE, the second (PAM-like) stage will be performed. |
| <code>useMedoids</code> | Only used for method "hybrid" and only if <code>labelUnlabeled==TRUE</code> . If TRUE, the second stage will be use object to medoid distance; if FALSE, it will use average object to cluster distance. The default (FALSE) is recommended. |
| <code>maxDistToLabel</code> | Only used for method "hybrid" and only if <code>labelUnlabeled==TRUE</code> . Maximum object distance to closest cluster that will result in the object assigned to that cluster. |
| <code>respectSmallClusters</code> | Only used for method "hybrid" and only if <code>labelUnlabeled==TRUE</code> . If TRUE, branches that failed to be clusters in stage 1 only because of insufficient size will be assigned together in stage 2. If FALSE, all objects will be assigned individually. |
| <code>verbose</code> | Controls the verbosity of the output. 0 will make the function completely quiet, values up to 4 gradually increase verbosity. |
| <code>indent</code> | Controls indentation of printed messages (see <code>verbose</code> above). Each unit adds two spaces before printed messages; useful when several functions' output is to be nested. |

Details

This is a wrapper for two related but different methods for cluster detection in hierarchical clustering dendrograms.

In order to make the shape parameters `maxCoreScatter` and `minGap` more universal, their values are interpreted relative to `cutHeight` and the 5th percetile of the merging heights (we arbitrarily chose the 5th percetile rather than the minimum for reasons of stability). Thus, the absolute maximum allowable core scatter is calculated as $\text{maxCoreScatter} * (\text{cutHeight} - \text{refHeight}) + \text{refHeight}$ and the absolute minimum allowable gap as $\text{minGap} * (\text{cutHeight} - \text{refHeight})$, where `refHeight` is the 5th percentile of the merging heights.

Value

A vector of numerical labels giving assignment of objects to modules. Unassigned objects are labeled 0, the largest module has label 1, next largest 2 etc.

Author(s)

Peter Langfelder, <Peter.Langfelder@gmail.com>

References

Langfelder P, Zhang B, Horvath S, 2007. <http://www.genetics.ucla.edu/labs/horvath/CoexpressionNetwork/BranchCutting>

See Also

[hclust](#), [cutreeHybrid](#), [cutreeDynamicTree](#).

cutreeDynamicTree *Dynamic dendrogram pruning based on dendrogram only*

Description

Detect clusters in a hierarchical dendrogram using a variable cut height approach. Uses only the information in the dendrogram itself is used (which may give incorrect assignment for outlying objects).

Usage

```
cutreeDynamicTree(dendro, maxTreeHeight = 1, deepSplit = TRUE, minModuleSize = 50)
```

Arguments

| | |
|---------------|--|
| dendro | Hierarchical clustering dendrogram such produced by hclust . |
| maxTreeHeight | Maximum joining height of objects to be considered part of clusters. |
| deepSplit | If TRUE, method will favor sensitivity and produce more smaller clusters. When FALSE, there will be fewer bigger clusters. |
| minModuleSize | Minimum module size. Branches containing fewer than minModuleSize objects will be left unlabeled. |

Details

A variable height branch pruning technique for dendrograms produced by hierarchical clustering. Initially, branches are cut off at the height `maxTreeHeight`; the resulting clusters are then examined for substructure and if subclusters are detected, they are assigned separate labels. Subclusters are detected by structure and are required to have a minimum of `minModuleSize` objects on them to be assigned a separate label. A rough degree of control over what it means to be a subcluster is implemented by the parameter `deepSplit`.

Value

A vector of numerical labels giving assignment of objects to modules. Unassigned objects are labeled 0, the largest module has label 1, next largest 2 etc.

Author(s)

Bin Zhang, binzhang.ucla@gmail.com, with contributions by Peter Langfelder, Peter.Langfelder@gmail.com.

References

<http://www.genetics.ucla.edu/labs/horvath/CoexpressionNetwork/BranchCutting>

See Also

[hclust](#), [cutreeHybrid](#)

| | |
|--------------|--|
| cutreeHybrid | <i>Hybrid adaptive tree cut for hierarchical clustering dendrograms.</i> |
|--------------|--|

Description

Detect clusters in a dendrogram produced by the function `hclust`.

Usage

```
cutreeHybrid(dendro, distM, cutHeight = NULL, minClusterSize = 20,
             deepSplit = 1,
             maxCoreScatter = NULL, minGap = NULL,
             maxAbsCoreScatter = NULL, minAbsGap = NULL, clusterTrim = 0,
             labelUnlabeled = NULL,
             pamStage = TRUE,
             useMedoids = FALSE, maxDistToLabel = cutHeight,
             respectSmallClusters = TRUE, verbose = 2, indent = 0)
```

Arguments

| | |
|----------------|---|
| dendro | a hierarchical clustering dendrogram such as one returned by <code>hclust</code> . |
| distM | Distance matrix that was used as input to <code>hclust</code> . |
| cutHeight | Maximum joining heights that will be considered. It defaults to 99 of the range between the 5th percentile and the maximum of the joining heights on the dendrogram. |
| minClusterSize | Minimum cluster size. |
| deepSplit | Either logical or integer in the range 0 to 4. Provides a rough control over sensitivity to cluster splitting. The higher the value, the more and smaller clusters will be produced. A finer control can be achieved via <code>maxCoreScatter</code> and <code>minGap</code> below. |

| | |
|-----------------------------------|--|
| <code>maxCoreScatter</code> | Maximum scatter of the core for a branch to be a cluster, given as the fraction of <code>cutHeight</code> relative to the 5th percentile of joining heights. See Details. |
| <code>minGap</code> | Minimum cluster gap given as the fraction of the difference between <code>cutHeight</code> and the 5th percentile of joining heights. |
| <code>maxAbsCoreScatter</code> | Maximum scatter of the core for a branch to be a cluster given as absolute heights. If given, overrides <code>maxCoreScatter</code> . |
| <code>minAbsGap</code> | Minimum cluster gap given as absolute height difference. If given, overrides <code>minGap</code> . |
| <code>clusterTrim</code> | Specifies the fraction of the cluster height that will be trimmed before stage 2; all objects joining above the <code>1-ClusterTrim</code> fraction of the cluster height will be trimmed. If set to 0 or less, nothing will be trimmed. |
| <code>labelUnlabeled</code> | Deprecated, please use <code>pamStage</code> instead. Only used for method "hybrid". If TRUE, the second (PAM-like) stage will be performed. |
| <code>pamStage</code> | Only used for method "hybrid". If TRUE, the second (PAM-like) stage will be performed. |
| <code>useMedoids</code> | if TRUE, the second stage will be use object to medoid distance; if FALSE, it will use average object to cluster distance. The default (FALSE) is recommended. |
| <code>maxDistToLabel</code> | Maximum object distance to closest cluster that will result in the object assigned to that cluster. |
| <code>respectSmallClusters</code> | If TRUE, branches that failed to be clusters in stage 1 only because of insufficient size will be assigned together in stage 2. If FALSE, all objects will be assigned individually. |
| <code>verbose</code> | Controls the verbosity of the output. 0 will make the function completely quiet, values up to 4 gradually increase verbosity. |
| <code>indent</code> | Controls indentation of printed messages (see <code>verbose</code> above). Each unit adds two spaces before printed messages; useful when several functions' output is to be nested. |

Details

The function detects clusters in a hierarchical dendrogram based on the shape of branches on the dendrogram. For details on the method, see <http://www.genetics.ucla.edu/labs/horvath/CoexpressionNetwork/BranchCutting>.

In order to make the shape parameters `maxCoreScatter` and `minGap` more universal, their values are interpreted relative to `cutHeight` and the 5th percetile of the merging heights (we arbitrarily chose the 5th percetile rather than the minimum for reasons of stability). Thus, the absolute maximum allowable core scatter is calculated as $\text{maxCoreScatter} * (\text{cutHeight} - \text{refHeight}) + \text{refHeight}$ and the absolute minimum allowable gap as $\text{minGap} * (\text{cutHeight} - \text{refHeight})$, where `refHeight` is the 5th percentile of the merging heights.

Value

A list containing the following elements:

| | |
|-------------|---|
| labels | Numerical labels of clusters, with 0 meaning unassigned, label 1 labeling the largest cluster etc. |
| cores | Numerical labels indicating cores of found clusters. |
| smallLabels | Numerical labels for branches that failed to be recognized clusters only because of insufficient number of objects. |
| trimmed | Numerical labels indicating objects that have been trimmed from clusters. |
| branches | A list detailing the detected branch structure. |

Author(s)

Peter Langfelder, Peter.Langfelder@gmail.com

References

Langfelder P, Zhang B, Horvath S, 2007. <http://www.genetics.ucla.edu/labs/horvath/CoexpressionNetwork/BranchCutting>

See Also

[hclust](#), [as.dist](#)

dynamicTreeCut-package

Methods for detection of clusters in hierarchical clustering dendrograms.

Description

Contains methods for detection of clusters in hierarchical clustering dendrograms.

Details

Package: dynamicTreeCut
Version: 1.12
Date: 2008-06-03
Depends: R, stats
ZipData: no
License: GPL version 2 or newer
URL: <http://www.genetics.ucla.edu/labs/horvath/CoexpressionNetwork/BranchCutting/>
Packaged: Tue Jun 03 12:32:38 2008; plangfelder
Built: R 2.4.1; ; 2008-06-03 12:48:47; unix

Index:

| | |
|--------------------------------|---|
| <code>cutreeDynamic</code> | Adaptive branch pruning of hierarchical clustering dendrograms. |
| <code>cutreeDynamicTree</code> | Dynamic dendrogram pruning based on dendrogram only |
| <code>cutreeHybrid</code> | Hybrid adaptive tree cut for hierarchical clustering dendrograms. |
| <code>indentSpaces</code> | Spaces for indented output. |
| <code>printFlush</code> | Print arguments and flush the console. |
| <code>treecut-package</code> | Methods for detection of clusters in hierarchical clustering dendrograms. |

Author(s)

Peter Langfelder <Peter.Langfelder@gmail.com> and Bin Zhang <binzhang.ucla@gmail.com>, with contributions from Steve Horvath <SHorvath@mednet.ucla.edu>

Maintainer: Peter Langfelder <Peter.Langfelder@gmail.com>

`indentSpaces` *Spaces for indented output.*

Description

Returns a character string containing two times `indent` spaces.

Usage

```
indentSpaces(indent = 0)
```

Arguments

`indent` Desired level of indentation. The number of returned spaces will be twice this argument.

Value

A character string containing spaces, of length twice `indent`.

Author(s)

Peter Langfelder, <Peter.Langfelder@gmail.com>

Examples

```
spaces = indentSpaces(0);
print(paste(spaces, "This output is not indented..."));
spaces = indentSpaces(1);
print(paste(spaces, "...while this one is."))
```

| | |
|-------------------------|---|
| <code>printFlush</code> | <i>Print arguments and flush the console.</i> |
|-------------------------|---|

Description

Passes all its arguments unchanged to the standard `print` function; after the execution of `print` it flushes the console, if possible.

Usage

```
printFlush(...)
```

Arguments

... Arguments to be passed to the standard `print` function.

Details

Passes all its arguments unchanged to the standard `print` function; after the execution of `print` it flushes the console, if possible.

Value

Returns the value of the `print` function.

Author(s)

Peter Langfelder, <Peter.Langfelder@gmail.com>

See Also

`print`

Index

*Topic **cluster**

cutreeDynamic, 1

cutreeHybrid, 5

*Topic **misc**

cutreeDynamicTree, 4

*Topic **package**

dynamicTreeCut-package, 7

*Topic **print**

indentSpaces, 8

printFlush, 9

as.dist, 7

cutreeDynamic, 1

cutreeDynamicTree, 4, 4

cutreeHybrid, 4, 5, 5

dynamicTreeCut

(dynamicTreeCut-package), 7

dynamicTreeCut-package, 7

hclust, 4, 5, 7

indentSpaces, 8

print, 9

printFlush, 9