

Package ‘Rclusterpp’

February 19, 2015

Type Package

Title Linkable C++ clustering

Version 0.2.3

Date 2013-09-23

Description Provide flexible native clustering routines that can be linked against in downstream packages.

License MIT + file LICENSE

LazyData yes

Depends R (>= 2.12.0), Rcpp (>= 0.10.4), RcppEigen (>= 0.1.2)

LinkingTo Rcpp, RcppEigen

Suggests RUnit, rbenchmark, fastcluster, inline

URL <https://github.com/nolanlab/Rclusterpp>

BugReports <https://github.com/nolanlab/Rclusterpp/issues>

NeedsCompilation yes

Author Michael Linderman [aut],
Robert Bruggner [cre]

Maintainer Robert Bruggner <rbruggner@gmail.com>

Repository CRAN

Date/Publication 2013-11-06 09:22:35

R topics documented:

Rclusterpp-package	2
Rclusterpp.hclust	2
Rclusterpp.linkageKinds	4
Rclusterpp.package.skeleton	5
Rclusterpp.setThreads	6
Index	8

Rclusterpp-package *Linkable C++ clustering*

Description

Rclusterpp provides flexible native clustering routines optimized for performance and minimal memory requirements. Like Rcpp, Rclusterpp is designed to be linked against by downstream packages.

Details

Rclusterpp provides replacements for hierarchical clustering functions, e.g. hclust, as a linkable library for building native implementations of custom clustering algorithms.

Author(s)

Michael D. Linderman, Robert Bruggner

Maintainer: Robert Bruggner <rbruggner@gmail.com>

References

Murtagh, F. (1983), "A survey of recent advances in hierarchical clustering algorithms", Computer Journal, 26, 354-359. Sibson, R. (1973), "SLINK: An optimally efficient algorithm for the single-link cluster method", Computer Journal, 16, 30-34.

See Also

[Rcpp RcppEigen](#)

Rclusterpp.hclust *Hierarchical Clustering*

Description

Hierarchical clustering on both dissimilarities and data

Usage

```
Rclusterpp.hclust(x, method = "ward", members = NULL, distance = "euclidean", p = 2)
```

Arguments

x	A numeric data matrix, data frame or a dissimilarity structure as produced by <code>dist</code> .
method	The agglomeration method to be used. This must be one of "ward", "single", "complete" or "average".
members	NULL or a vector with length size of x. See hclust .
distance	The distance measure to be used. This must be one of "euclidean", "manhattan", "maximum", or "minkowski".
p	The power of the Minkowski distance.

Details

If x is a dissimilarity matrix, execution defaults to standard `hclust`. If x is a set of observations, specialized native clustering routines are invoked. These routines are optimized for O(n) memory footprint and multicore execution to permit clustering of large datasets.

Value

An object of class `*hclust*` which describes the tree produced by the clustering process. See [hclust](#).

Note

Support for different agglomeration methods and distance metrics is evolving.

Author(s)

Michael Linderman

References

Murtagh, F. (1983), "A survey of recent advances in hierarchical clustering algorithms", *Computer Journal*, 26, 354-359. Sibson, R. (1973), "SLINK: An optimally efficient algorithm for the single-link cluster method", *Computer Journal*, 16, 30-34.

See Also

[hclust](#)

Examples

```
h <- Rclusterpp.hclust(USArrests, method="ward", distance="euclidean")
```

Rclusterpp.linkageKinds

Linkage and distance kinds available in Rclusterpp

Description

Functions that return character vectors with the available linkage and distance kinds that correspond with internal enums. Intended for use with ‘pmatch’.

Usage

```
Rclusterpp.linkageKinds()  
Rclusterpp.distanceKinds()
```

Details

Returns character vectors whose ordering enables integer to enum conversion for ‘LinkageKinds’ and ‘DistanceKinds’ enums within the Rclusterpp library.

Value

Character vectors.

Linkage Kinds: "ward", "average", "single", "complete"

Distance Kinds: "euclidean", "manhattan", "maximum", "minkowski"

Author(s)

Michael Linderman

See Also

[Rclusterpp.hclust](#)

Examples

```
Rclusterpp.linkageKinds()  
Rclusterpp.distanceKinds()
```

Rclusterpp.package.skeleton

Create a skeleton for a new package that intends to use Rclusterpp

Description

Rclusterpp.package.skeleton automates the creation of a new source package that intends to use features of Rclusterpp.

It is based on the [RcppEigen.package.skeleton](#) and executes [package.skeleton](#) internally.

Usage

```
Rclusterpp.package.skeleton(name = "anRpackage", list = character(),
environment = .GlobalEnv, path = ".", force = FALSE, namespace = TRUE,
code_files = character(), example_code = TRUE)
```

Arguments

name	See package.skeleton
list	See package.skeleton
environment	See package.skeleton
path	See package.skeleton
force	See package.skeleton
namespace	See package.skeleton
code_files	See package.skeleton
example_code	If TRUE, example C++ code using Rclusterpp is added to the package

Details

In addition to [package.skeleton](#) :

The 'DESCRIPTION' file gains a Depends line requesting that the package depends on Rcpp and Rclusterpp and a LinkingTo line so that the package finds Rcpp and Rclusterpp header files.

The 'NAMESPACE', if any, gains a useDynLib directive.

The 'src' directory is created if it does not exist and a 'Makevars' file is added setting the environment variable 'PKG_LIBS' to accommodate the necessary flags to link with the Rcpp library.

If the example_code argument is set to TRUE, example files 'rclusterpp_hello_world.h' and 'rclusterpp_hello_world.cpp' are also created in the 'src'. An R file 'rclusterpp_hello_world.R' is expanded in the 'R' directory, the rclusterpp_hello_world function defined in this file makes use of the C++ function 'rclusterpp_hello_world' defined in the C++ file. These files are given as an example and should eventually be removed from the generated package.

Value

Nothing, used for its side effects

References

Read the *Writing R Extensions* manual for more details.

Once you have created a *source* package you need to install it: see the *R Installation and Administration* manual, [INSTALL](#) and [install.packages](#).

See Also

[RcppEigen.package.skeleton](#) [package.skeleton](#)

Examples

```
## Not run:  
Rclusterpp.package.skeleton( "foobar" )  
  
## End(Not run)
```

Rclusterpp.setThreads *Set number of threads used in clustering*

Description

Sets the number of threads used by the OpenMP based parallelism in hierarchical clustering.

Usage

```
Rclusterpp.setThreads(threads = 1)
```

Arguments

threads	Desired number of threads. NULL will set number of threads to number of processors.
---------	---

Details

Simple wrapper for `omp_set_num_threads` when OpenMP is available, otherwise behaves as a NOOP. When `threads==NULL`, the number of processors will be determined via `omp_get_num_procs` and number of threads set to that value. Note on modern Intel processors with hyperthreading, this number is typically the number of hyperthread cores, and thus typically two times the number of physical cores. Setting the threading that high is not always advantageous. Note that number of threads can also be set via the `OMP_NUM_THREADS` environment variable.

Value

Integer number of threads

Author(s)

Michael Linderman

References

<http://openmp.org/>

Examples

```
# Set to one thread  
Rclusterpp.setThreads(1)
```

```
# Set to number of threads equal to number of processors  
Rclusterpp.setThreads()
```

Index

*Topic **package**

Rclusterpp-package, 2

*Topic **programming**

Rclusterpp.package.skeleton, 5

hclust, 3

INSTALL, 6

install.packages, 6

package.skeleton, 5, 6

Rclusterpp (Rclusterpp-package), 2

Rclusterpp-package, 2

Rclusterpp.distanceKinds
(Rclusterpp.linkageKinds), 4

Rclusterpp.hclust, 2, 4

Rclusterpp.linkageKinds, 4

Rclusterpp.package.skeleton, 5

Rclusterpp.setThreads, 6

Rcpp, 2

RcppEigen, 2

RcppEigen.package.skeleton, 5, 6