Package 'caviarpd'

December 20, 2022

Type Package

Title Cluster Analysis via Random Partition Distributions

Version 0.3.6

Description Cluster analysis is performed using pairwise distance information and a random partition distribution. The method is
implemented for two random partition distributions. It draws samples and then obtains and plots clustering estimates.
An implementation of a selection algorithm is provided for the mass parameter of the partition distribution. Since
pairwise distances are the principal input to this procedure, it is most comparable to the hierarchical and k-medoids
clustering methods. The method is Dahl, Andros, Carter (2022+) <doi:10.1002/sam.11602>.

License MIT + file LICENSE | Apache License 2.0

BugReports https://github.com/dbdahl/fangs/issues

Depends R (>= 4.2.0)

Suggests salso (>= 0.3.0)

SystemRequirements Cargo (>= 1.56) for installation from source: see
INSTALL file

Encoding UTF-8

RoxygenNote 7.2.3

NeedsCompilation yes

Author David B. Dahl [aut, cre] (<https://orcid.org/0000-0002-8173-1547>),
R. Jacob Andros [aut] (<https://orcid.org/0000-0002-1289-385X>),
J. Brandon Carter [aut] (<https://orcid.org/0000-0003-1687-0564>),
Alex Crichton [cph] (Rust crates: proc-macro2, cfg-if. See AUTHORS
file.),
``bluss'' Ulrik Sverdrup [cph] (Rust crate: rawpointer, matrixmultiply,
scopeguard, ndarray. See AUTHORS file.),
Brendan Zabarauskas [cph] (Rust crate: approx. See AUTHORS file.),
David B. Dahl [cph] (Rust crates: roxido, roxido_macro. See AUTHORS
file.),
David Tolnay [cph] (Rust crates: syn, proc-macro2, quote,

unicode-ident. See AUTHORS file.),
David Schultz [cph] (Rust crates: libm. See AUTHORS file.),
DutchGhost [cph] (Rust crate: matrixmultiply. See AUTHORS file.),
Enthought, Inc. [cph] (Rust crate: ndarray. See AUTHORS file.),
Gilad Naaman [cph] (Rust crate: memoffset. See AUTHORS file.),
Jim Turner [cph] (Rust crate: ndarray. See AUTHORS file.),
Jorge Aparicio [cph] (Rust crate: libm. See AUTHORS file.),
Josh Stone [cph] (Rust crate: autocfg. See AUTHORS file.),
Melissa O'Neill [cph] (Rust crate: rand_pcg. See AUTHORS file.),
Mikhail Vorotilov [cph] (Rust crate: roots. See AUTHORS file.),
Paul Dicker [cph] (Rust crate: rand_pcg. See AUTHORS file.),
PCG Project contributors [cph] (Rust crate: rand_pcg. See AUTHORS file.),
Ralf Jung [cph] (Rust crate: memoffset. See AUTHORS file.),
rawpointer developers [cph] (Rust crate: rawpointer. See AUTHORS file.),
R. Janis Goldschmidt [cph] (Rust crate: matrixmultiply. See AUTHORS file.),
SciPy Developers [cph] (Rust crate: ndarray. See AUTHORS file.),
Sean McArthur [cph] (Rust crate: num_cpus. See AUTHORS file.),
Sun Microsystems, Inc. [cph] (Rust crate: libm. See AUTHORS file.),
Stefan Lankes [cph] (Rust crate: hermit-abi. See AUTHORS file.),
The Cranelift Project Developers [cph] (Rust crate: wasi. See AUTHORS file.),
The Crossbeam Project Developers [cph] (Rust crates: crossbeam, crossbeam-channel, crossbeam-deque, crossbeam-epoch, crossbeam-queue, crossbeam-utils. See AUTHORS file.),
The CryptoCorrosion Contributors [cph] (Rust crates: ppv-lite86, rand_chacha. See AUTHORS file.),
The Go Authors [cph] (Rust crate: crossbeam-channel. See AUTHORS file.),
The PCG Project Contributors [cph] (Rust crate: rand_pcg. See AUTHORS file.),
The matrixmultiply Authors [cph] (Rust crate: matrixmultiply. See AUTHORS file.),
The ndarray Developers [cph] (Rust crate: ndarray. See AUTHORS file.),
The Rand Project Developers [cph] (Rust crates: rand_core, rand_chacha, rand_pcg, rand, getrandom, rand_distr. See AUTHORS file.),
The Rust Project Developers [cph] (Rust crates: crossbmean-channel, rand_chacha, num-integer, rand_core, num-complex, libc, num-traits, rand. See AUTHORS file.),
The scopeguard Developers [cph] (Rust crates: scopeguard. See AUTHORS file.),
Unicode, Inc. [cph] (Rust crate: unicode-ident. See AUTHORS file.)

Maintainer David B. Dahl <dahl@stat.byu.edu>

Repository CRAN

Date/Publication 2022-12-20 03:50:08 UTC

R topics documented:

`caviarpd` *Cluster Analysis via Random Partition Distributions*

Description

Returns a clustering estimate given pairwise distances using the CaviarPD method.

Usage

```
caviarpd(
  distance,
  nClusters,
  mass = NULL,
  nSamples = 200,
  gridLength = 5,
  loss = "binder",
  temperature = 100,
  similarity = c("exponential", "reciprocal")[1],
  maxNClusters = 0,
  nRuns = 4,
  nCores = nRuns
)
```

Arguments

`distance`	An object of class 'dist' or a pairwise distance matrix.
`nClusters`	A numeric vector that specifies the range for the number of clusters to consider in the search for a clustering estimate.
`mass`	The mass value to use for sampling. If `NULL`, the mass value is found by inverting values from `nClusters`.
`nSamples`	The number of samples drawn per candidate estimate.
`gridLength`	The number of candidate estimates to consider. The final estimate is obtained from `nSamples` $\times$ `gridLength` total samples.
`loss`	The SALSO method (Dahl, Johnson, Müller, 2021) tries to minimize this expected loss when searching the partition space for an optimal estimate. This must be either "binder" or "VI".
`temperature`	A positive number that accentuates or dampens distance between observations.
`similarity`	Either `"exponential"` or `"reciprocal"` to indicate the desired similarity function.
`maxNClusters`	The maximum number of clusters that can be considered by the SALSO method.

nRuns	The number of runs of the SALSO algorithm.
nCores	The number of CPU cores to use. A value of zero indicates to use all cores on the system.

Details

A range for the number of clusters to be considered is supplied using the nClusters argument.

Value

A object of class salso.estimate, which provides a clustering estimate (a vector of cluster labels) that can be displayed and plotted.

References

D. B. Dahl, J. Andros, J. B. Carter (2022+), Cluster Analysis via Random Partition Distributions, *Statistical Analysis and Data Mining*, accepted. doi:10.1002/sam.11602.

D. B. Dahl, D. J. Johnson, and P. Müller (2022), Search Algorithms and Loss Functions for Bayesian Clustering, *Journal of Computational and Graphical Statistics*, doi:10.1080/10618600.2022.2069779.
’

Examples

```
# To reduce load on CRAN servers, limit the number of samples, grid length, and CPU cores.
set.seed(34)
iris.dis <- dist(iris[,-5])
est <- caviarpd(distance=iris.dis, nClusters=c(2,4), nSamples=20, nCores=1)
if ( require("salso") ) {
  summ <- summary(est, orderingMethod=2)
  plot(summ, type="heatmap")
  plot(summ, type="mds")
}
```

Index