

Package ‘Clustering’

January 7, 2021

Type Package

Title Execution of Multiple Clustering Algorithm

Version 1.7.2

Date 2020-12-21

Author Luis Alfonso Perez Martos [aut, cre]

Maintainer Luis Alfonso Perez Martos <lapm0001@gmail.com>

Description

The design of this package allows us to run different clustering packages and compare the results between them, to determine which algorithm behaves best from the data provided.

URL <https://github.com/laperez/clustering>

Depends R (>= 3.5.0)

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Repository CRAN

Imports advclust, amap, apcluster, cluster, ClusterR, data.table,
doParallel, dplyr, foreach, future, gama, ggplot2, gmp, pracma,
pvclust, shiny, sqldf, stats, tools, utils, xtable

Suggests DT, kableExtra, knitr, rmarkdown, shinyalert, shinyFiles,
shinyjs, shinythemes, shinyWidgets, tidyverse, shinycssloaders

VignetteBuilder knitr

NeedsCompilation no

Date/Publication 2021-01-07 10:30:06 UTC

R topics documented:

appClustering	2
basketball	3

best_ranked_external_metrics	3
best_ranked_internal_metrics	4
bolts	5
clustering	6
evaluate_best_validation_external_by_metrics	9
evaluate_best_validation_internal_by_metrics	10
evaluate_validation_external_by_metrics	11
evaluate_validation_internal_by_metrics	12
export_file_external	13
export_file_internal	13
plot_clustering	14
result_external_algorithm_by_metric	15
result_internal_algorithm_by_metric	16
sort.clustering	17
stock	18
stulong	19
weather	19
[.clustering	20

Index	22
--------------	-----------

appClustering	<i>Clustering GUI.</i>
---------------	------------------------

Description

Method that allows us to execute the main algorithm in graphic interface mode instead of through the console.

Usage

```
appClustering()
```

Details

The operation of this method is to generate a graphical user interface to be able to execute the clustering algorithm without knowing the parameters. Its operation is very simple, we can change the values and see the behavior quickly.

Value

GUI with the parameters of the algorithm and their representation in tables and graphs.

basketball	<i>This data set contains a series of statistics (5 attributes) about 96 basketball players:</i>
------------	--

Description

This data set contains a series of statistics about basketball players:

Usage

```
data(basketball)
```

Format

A data frame with 96 observations on 5 variables:

This data set contains a series of statistics about basketball players:

assists_per_minuteReal average number of assistances per minute

heightInteger height of the player

time_playedReal time played by the player

ageInteger number of years of the player

points_per_minuteReal average number of points per minute

Source

KEEL, <<http://www.keel.es/>>

best_ranked_external_metrics	<i>Best rated external metrics.</i>
------------------------------	-------------------------------------

Description

method in charge of searching for each algorithm those that have the best external classification.

Method that looks for those external attribute that are better classified, making use of the ranking column. In this way we discard the rest of the attribute and only work with those that give the best response to the algorithm in question.

Usage

```
best_ranked_external_metrics(df)
```

Arguments

df matrix or data frame with the result of running the clustering algorithm.

Value

Returns a data.frame with the best classified external attribute.

Examples

```
result = clustering(
  df = cluster::agriculture,
  min = 4,
  max = 5,
  algorithm='gmm',
  metrics=c("recall"),
  attributes = FALSE
)

best_ranked_external_metrics(df = result)

## Not run:
best_ranked_external_metrics(df = result$result)

## End(Not run)
```

best_ranked_internal_metrics

Best rated internal metrics.

Description

method in charge of searching for each algorithm those that have the best internal classification.

Method that looks for those internal attributes that are better classified, making use of the ranking column. In this way we discard the rest of the attributes and only work with those that give the best response to the algorithm in question.

Usage

```
best_ranked_internal_metrics(df)
```

Arguments

df matrix or data frame with the result of running the clustering algorithm.

Value

Returns a data.frame with the best classified internal attributes.

Examples

```
result = clustering(  
  df = cluster::agriculture,  
  min = 4,  
  max = 5,  
  algorithm='gmm',  
  metrics=c("recall"),  
  attributes = FALSE  
)  
  
best_ranked_internal_metrics(df = result)  
  
## Not run:  
best_ranked_internal_metrics(df = result$result)  
  
## End(Not run)
```

bolts	<i>Data from an experiment on the affects of machine adjustments on the time to count bolts.</i>
-------	--

Description

A manufacturer of automotive accessories provides hardware, e.g. nuts, bolts, washers and screws, to fasten the accessory to the car or truck. Hardware is counted and packaged automatically. Specifically, bolts are dumped into a large metal dish. A plate that forms the bottom of the dish rotates counterclockwise. This rotation forces bolts to the outside of the dish and up along a narrow ledge. Due to the vibration of the dish caused by the spinning bottom plate, some bolts fall off the ledge and back into the dish. The ledge spirals up to a point where the bolts are allowed to drop into a pan on a conveyor belt. As a bolt drops, it passes by an electronic eye that counts it. When the electronic counter reaches the preset number of bolts, the rotation is stopped and the conveyor belt is moved forward

Usage

```
data(bolts)
```

Format

A data frame with 40 observations on 8 variables:

A manufacturer of automotive accessories provides hardware, e.g. nuts, bolts, washers and screws, to fasten the accessory to the car or truck. Hardware is counted and packaged automatically. Specifically, bolts are dumped into a large metal dish. A plate that forms the bottom of the dish rotates counterclockwise. This rotation forces bolts to the outside of the dish and up along a narrow ledge. Due to the vibration of the dish caused by the spinning bottom plate, some bolts fall off the ledge and back into the dish. The ledge spirals up to a point where

the bolts are allowed to drop into a pan on a conveyor belt. As a bolt drops, it passes by an electronic eye that counts it. When the electronic counter reaches the preset number of bolts, the rotation is stopped and the conveyor belt is moved forward

RUNInteger is the order in which the data were collected

SPEED1Integer a speed setting that controls the speed of rotation of the plate at the bottom of the dish

TOTALInteger total number of bolts (TOTAL) to be counted

SPEED2Integer a second speed setting that is used to change the speed of rotation (usually slowing it down) for the last few bolts

NUMBER2Integer the number of bolts to be counted at this second speed

SENSInteger the sensitivity of the electronic eye

TIMERReal The measured response is the time, in seconds

T20BOLTRReal in order to put times on a equal footing the response to be analyzed is the time to count 20 bolts

Details

There are several adjustments on the machine that affect its operation. These include; a speed setting that controls the speed of rotation (SPEED1Integer) of the plate at the bottom of the dish, a total number of bolts (TOTAL) to be counted, a second speed setting (SPEED2Integer) that is used to change the speed of rotation (usually slowing it down) for the last few bolts, the number of bolts to be counted at this second speed (NUMBER2Integer), and the sensitivity of the electronic eye (SENSInteger). The sensitivity setting is to insure that the correct number of bolts are counted. Too few bolts packaged causes customer complaints. Too many bolts packaged increases costs. For each run conducted in this experiment the correct number of bolts was counted. From an engineering standpoint if the correct number of bolts is counted, the sensitivity should not affect the time to count bolts. The measured response is the time (TIMERReal), in seconds, it takes to count the desired number of bolts. In order to put times on a equal footing the response to be analyzed is the time to count 20 bolts (T20BOLTRReal). Below are the data for 40 combinations of settings. RUNInteger is the order in which the data were collected.

Source

KEEL, <<http://www.keel.es/>>

clustering

Clustering algorithm.

Description

Discovering the behavior of attributes in a set of clustering packages based on evaluation metrics.

Usage

```
clustering(
  path = NULL,
  df = NULL,
  packages = NULL,
  algorithm = NULL,
  min = 3,
  max = 4,
  metrics = NULL,
  attributes = FALSE
)
```

Arguments

<code>path</code>	The path of file. <code>NULL</code> It is only allowed to use <code>path</code> or <code>df</code> but not both at the same time. Only files in <code>.dat</code> , <code>.csv</code> or <code>arff</code> format are allowed.
<code>df</code>	data matrix or data frame, or dissimilarity matrix. <code>NULL</code> If you want to use training and test basketball attributes.
<code>packages</code>	character vector with the packets running the algorithm. <code>NULL</code> The seven packages implemented are: <code>cluster</code> , <code>ClusterR</code> , <code>advclust</code> , <code>amap</code> , <code>apcluster</code> , <code>gama</code> , <code>pvclust</code> . By default runs all packages.
<code>algorithm</code>	character vector with the algorithms implemented within the package. <code>NULL</code> The algorithms implemented are: <code>fuzzy_cm</code> , <code>fuzzy_gg</code> , <code>fuzzy_gk</code> , <code>hclust</code> , <code>apclusterK</code> , <code>agnes</code> , <code>clara</code> , <code>daisy</code> , <code>diana</code> , <code>fanny</code> , <code>mona</code> , <code>pam</code> , <code>gmm</code> , <code>kmeans_arma</code> , <code>kmeans_rcpp</code> , <code>mini_kmeans</code> , <code>gama</code> , <code>pvclust</code> .
<code>min</code>	An integer with the minimum number of clusters This data is necessary to indicate the minimum number of clusters when grouping the data. The default value is 3.
<code>max</code>	An integer with the maximum number of clusters. This data is necessary to indicate the maximum number of clusters when grouping the data. The default value is 4.
<code>metrics</code>	Character vector with the metrics implemented to evaluate the distribution of the data in clusters. <code>NULL</code> The night metrics implemented are: <code>entropy</code> , <code>variation_information</code> , <code>precision</code> , <code>recall</code> , <code>f_measure</code> , <code>fowlkes_mallows_index</code> , <code>connectivity</code> , <code>dunn</code> , <code>silhouette</code> .
<code>attributes</code>	an boolean which indicates that if we want to show as a result the attributes of the datasets or the numerical value of the calculation of the metrics. The default value is <code>F</code> .

Details

The operation of this algorithm is to evaluate how the attributes of a dataset or a set of datasets behave in different grouping algorithms. To do this, it is necessary to indicate the type of evaluation you want to make on the distribution of the data. To be able to execute the algorithm it is necessary to indicate the number of clusters `min` and `max`, the algorithms `algorithm` or packages `packages`

that we want to cluster, the metrics `metrics` and if we want that the results of evaluation are the own classified attributes or numerical values `attributes`.

Value

a matrix with the result of running all the metrics of the algorithms contained in the packages we indicated. We also obtain information with the types of metrics, algorithms and packages executed.

- `result` It is a list with the algorithms, metrics and variables defined in the execution of the algorithm.
- `has_internal_metrics` Boolean field to indicate if there are internal metrics such as: `dunn`, `silhouette` and `connectivity`.
- `has_external_metrics` Boolean field to indicate if there are external metrics such as: `precision`, `recall`, `f-measure`, `entropy`, `variation information` and `fowlkes-mallows`.
- `algorithms_execute` Character vector with the algorithms executed. These algorithms have been mentioned in the definition of the parameters.
- `measures_execute` Character vector with the measures executed. These measures have been mentioned in the definition of the parameters.

How does this algorithm work?

This algorithm improves and complements existing implementations of clustering algorithms.

The approaches that exist, are many algorithms that run parallel to the algorithms, without being able to be compared between them. In addition, it was necessary to indicate which variable of the dataset is required to be executed. In addition, depending on the package there are some implementations or others to evaluate the groupings of data, so it is sometimes complicated to compare the groupings between different packages.

With this algorithm we can solve the problems mentioned above and determine which algorithm has the best behavior for the set of attributes as well as the most efficient number of clusters.

Examples

```
clustering(
  df = cluster::agriculture,
  min = 4,
  max = 5,
  algorithm='gmm',
  metrics='precision',
  attributes = TRUE
)

## Not run:
clustering(
  df = Clustering::weather,
  min = 2,
  max = 3,
  algorithm= c("gmm", "kmeans_armaa"),
  metrics= c("precision", "dunn"),
```



```
        attributes = FALSE
    )

    ## End(Not run)
```

```
evaluate_best_validation_external_by_metrics
```

Evaluation of the algorithms by measures of dissimilarity.

Description

Method that calculates which algorithm and which metric behaves best for the datasets provided.

Usage

```
evaluate_best_validation_external_by_metrics(df)
```

Arguments

df data matrix or data frame with the result of running the clustering algorithm.

Details

Method that calculates the behavior of dissimilarity measures by algorithm, so we can evaluate which of the different measures of dissimilarity used by the algorithms presents the best behavior. This method should be used to determine which dissimilarity measure has the best behavior for external evaluation measures.

Value

a data.frame with the algorithms classified by measures of dissimilarity.

Examples

```
result = clustering(
  df = cluster::agriculture,
  min = 4,
  max = 5,
  algorithm='kmeans_rcpp',
  metrics=c("f_measure"),
  attributes = TRUE
)

evaluate_best_validation_external_by_metrics(result)
```

`evaluate_best_validation_internal_by_metrics`*Evaluation of the algorithms by measures of dissimilarity.*

Description

Method that calculates which algorithm and which metric behaves best for the datasets provided.

Usage

```
evaluate_best_validation_internal_by_metrics(df)
```

Arguments

`df` data matrix or data frame with the result of running the clustering algorithm.

Details

Method that calculates the behavior of dissimilarity measures by algorithm, so we can evaluate which of the different measures of dissimilarity used by the algorithms presents the best behavior. This method should be used to determine which dissimilarity measure has the best behavior for internal evaluation measures.

Value

a data.frame with the algorithms classified by measures of dissimilarity.

Examples

```
result = clustering(df = cluster::agriculture, min = 4, max = 5,  
algorithm='gmm', attributes = TRUE)
```

```
evaluate_best_validation_internal_by_metrics(result)
```

```
result = clustering(  
  df = cluster::agriculture,  
  min = 4,  
  max = 5,  
  algorithm='gmm',  
  metrics=c("connectivity"),  
  attributes = TRUE  
)
```

```
evaluate_best_validation_internal_by_metrics(result)
```

evaluate_validation_external_by_metrics
Evaluate external validations by algorithm.

Description

Method that calculates which algorithm behaves best for the datasets provided.

Usage

```
evaluate_validation_external_by_metrics(df)
```

Arguments

df data matrix or data frame with the result of running the clustering algorithm.

Details

The operation of this method is to determine which algorithm has better behavior regardless of the measure of dissimilarity calculated, so we can determine which algorithm returns better results from the attributes and measures of dissimilarity.

Value

a data.frame with all the algorithms that obtain the best results regardless of the dissimilarity measure used.

Examples

```
result = clustering(  
  df = cluster::agriculture,  
  min = 4,  
  max = 5,  
  algorithm='kmeans_arma',  
  metrics=c("precision"),  
  attributes = TRUE  
)  
  
evaluate_validation_external_by_metrics(result)  
  
## Not run:  
evaluate_validation_external_by_metrics(result$result)  
  
## End(Not run)
```

`evaluate_validation_internal_by_metrics`*Evaluate internal validations by algorithm.*

Description

Method that calculates which algorithm behaves best for the datasets provided.

Usage

```
evaluate_validation_internal_by_metrics(df)
```

Arguments

`df` data matrix or data frame with the result of running the clustering algorithm.

Details

The operation of this method is to determine which algorithm has better behavior regardless of the measure of dissimilarity calculated, so we can determine which algorithm returns better results from the attributes and measures of dissimilarity.

Value

a data.frame with all the algorithms that obtain the best results regardless of the dissimilarity measure used.

Examples

```
result = clustering(  
  df = cluster::agriculture,  
  min = 4,  
  max = 5,  
  algorithm='kmeans_rcpp',  
  metrics=c("silhouette"),  
  attributes = TRUE  
)  
  
evaluate_validation_internal_by_metrics(result)  
  
## Not run:  
evaluate_validation_internal_by_metrics(result$result)  
  
## End(Not run)
```

export_file_external *Export result of external metrics in latex.*

Description

Method that exports the results of internal measurements in latex format to a file.

Usage

```
export_file_external(df, path = NULL)
```

Arguments

df is a dataframe that contains as a parameter a table in latex format with the results of the internal validations.

path it's a string with the path to a directory where a file is to be stored in latex format.

Details

When we work in latex format and we need to create a table to export the results, with this method we can export the results of the clustering algorithm to latex.

Examples

```
result = clustering(  
  df = cluster::agriculture,  
  min = 4,  
  max = 5,  
  algorithm='gmm',  
  metrics=c("precision"),  
  attributes = TRUE  
)  
  
export_file_external(result)  
file.remove("external_data.tex")
```

export_file_internal *Export result of internal metrics in latex.*

Description

Method that exports the results of external measurements in latex format to a file.

Usage

```
export_file_internal(df, path = NULL)
```

Arguments

df	is a dataframe that contains as a parameter a table in latex format with the results of the external validations.
path	it's a string with the path to a directory where a file is to be stored in latex format.

Details

When we work in latex format and we need to create a table to export the results, with this method we can export the results of the clustering algorithm to latex.

Examples

```

result = clustering(
    df = cluster::agriculture,
    min = 4,
    max = 5,
    algorithm='gmm',
    metrics=c("dunn"),
    attributes = TRUE
)

export_file_internal(result)
file.remove("internal_data.tex")

```

plot_clustering

Graphic representation of the evaluation measures.

Description

Graphical representation of the evaluation measures grouped by cluster.

Usage

```
plot_clustering(df, metric)
```

Arguments

df	data matrix or data frame with the result of running the clustering algorithm.
metric	it's a string with the name of the metric select to evaluate.

Details

In certain cases the review or filtering of the data is necessary to select the data, that is why thanks to the graphic representations this task is much easier. Therefore with this method we will be able to filter the data by metrics and see the data in a graphical way.

Examples

```
result = clustering(  
    df = cluster::agriculture,  
    min = 4,  
    max = 5,  
    algorithm='gmm',  
    metrics=c("precision"),  
    attributes = TRUE  
)  
  
plot_clustering(result,c("precision"))
```

```
result_external_algorithm_by_metric  
    External results by algorithm
```

Description

Method that returns a data.frame with the algorithm and the metric indicated as parameters.

Usage

```
result_external_algorithm_by_metric(df, algorithm)
```

Arguments

df	data matrix or data frame with the result of running the clustering algorithm.
algorithm	It's a string with the algorithm we want to evaluate your results

Details

The functionality of this method is to return as a result a data.frame with the algorithm indicated as a parameter along with the rest of the dissimilarity measurements and the external evaluation metrics.

Value

a data.frame with the results of the algorithm indicated as parameter.

Examples

```
result = clustering(  
    df = cluster::agriculture,  
    min = 4,  
    max = 5,
```

```

        algorithm='gmm',
        metrics=c("precision"),
        attributes = FALSE
    )

result_external_algorithm_by_metric(result,'daisy')

```

```

result_internal_algorithm_by_metric
    Internal results by algorithm

```

Description

Method that returns a data.frame with the algorithm and the metric indicated as parameters.

Usage

```
result_internal_algorithm_by_metric(df, algorithm)
```

Arguments

df	data matrix or data frame with the result of running the clustering algorithm.
algorithm	It's a string with the algorithm we want to evaluate your results

Details

The functionality of this method is to return as a result a data.frame with the algorithm indicated as a parameter along with the rest of the dissimilarity measurements and the internal evaluation metrics.

Value

a data.frame with the results of the algorithm indicated as parameter.

Examples

```

result = clustering(
  df = cluster::agriculture,
  min = 4,
  max = 5,
  algorithm='gmm',
  metrics=c("silhouette"),
  attributes = FALSE
)

result_internal_algorithm_by_metric(result,'gmm')

```

sort.clustering	Returns the clustering result sorted by a set of metrics.
-----------------	---

Description

This function receives a clustering object and sorts the indicator columns by parameter. By default it performs sorting by the algorithm field.

Usage

```
## S3 method for class 'clustering'  
sort(x, decreasing = TRUE, ...)
```

Arguments

x	is an clustering object.
decreasing	A logical indicating if the sort should be increasing or decreasing. By default, decreasing.
...	Additional parameters as "by", a String with the name of the evaluation measure to order by. Valid values are: Algorithm,Distance,Clusters,Dataset,Ranking,timeExternal,entropy,va

Details

The additional argument in "..." is the 'by' argument, which is a array with the name of the evaluation measure to order by. Valid value are: Algorithm,Distance,Clusters,Dataset,Ranking,timeExternal,entropy,va

Value

another clustering object with the evaluation measures sorted

Examples

```
library(Clustering)  
  
result <-  
clustering(df = cluster::agriculture,min = 4, max = 4,algorithm='gmm',  
metrics='recall')  
  
sort(result, FALSE, 'recall')
```

stock	<i>The data provided are daily stock prices from January 1988 through October 1991, for ten aerospace companies.</i>
-------	--

Description

The data provided are daily stock prices from January 1988 through October 1991, for ten aerospace companies.

Usage

`data(stock)`

Format

A data frame with 950 observations on 10 variables:

The data provided are daily stock prices from January 1988 through October 1991, for ten aerospace companies.

Company1 company1 details

Company2 company2 details

Company3 company3 details

Company4 company4 details

Company5 company5 details

Company6 company6 details

Company7 company7 details

Company8 company8 details

Company9 company9 details

Company10 company10 details

Source

KEEL, <<http://www.keel.es/>>

stulong	<i>The study was performed at the 2nd Department of Medicine, 1st Faculty of Medicine of Charles University and Charles University Hospital. The data were transferred to electronic form by the European Centre of Medical Informatics, Statistics and Epidemiology of Charles University and Academy of Sciences.</i>
---------	---

Description

The study was performed at the 2nd Department of Medicine, 1st Faculty of Medicine of Charles University and Charles University Hospital. The data were transferred to electronic form by the European Centre of Medical Informatics, Statistics and Epidemiology of Charles University and Academy of Sciences.

Usage

data(stulong)

Format

A data frame with 1417 observations on 5 variables.

The study was performed at the 2nd Department of Medicine, 1st Faculty of Medicine of Charles University and Charles University Hospital. The data were transferred to electronic form by the European Centre of Medical Informatics, Statistics and Epidemiology of Charles University and Academy of Sciences.

a1 Height

a2 Weight

a3 Blood pressure I systolic (mm Hg)

a4 Blood pressure I diastolic (mm Hg)

a5 percentage Cholesterol in mg

Source

KEEL, <<http://www.keel.es/>>

weather	<i>One of the most known testing data sets in machine learning. This data sets describes several situations where the weather is suitable or not to play sports, depending on the current outlook, temperature, humidity and wind.</i>
---------	--

Description

One of the most known testing data sets in machine learning. This data sets describes several situations where the weather is suitable or not to play sports, depending on the current outlook, temperature, humidity and wind.

Usage

```
data(weather)
```

Format

A data frame with 14 observations on 5 variables:

One of the most known testing data sets in machine learning. This data sets describes several situations where the weather is suitable or not to play sports, depending on the current outlook, temperature, humidity and wind.

Outlook sunny, overcast, rainy

Temperature hot, mild, cool

Humidity high, normal

Windy true, false

Play yes, no

Source

KEEL, <<http://www.keel.es/>>

[.clustering	<i>Filter metrics in a clustering object returning a new clustering object.</i>
--------------	---

Description

Generates a new clustering object containing the metrics that passed the filter specified.

Usage

```
## S3 method for class 'clustering'
clustering[condition = T]
```

Arguments

clustering	The clustering object to filter.
condition	Expression to filter the clustering object

Details

This function allows you to filter the data set for a given evaluation metric. The evaluation metrics available are: Algorithm,Distance,Clusters,Dataset,Ranking,timeExternal,entropy,variation_information,p

Examples

```
library(Clustering)

result <- clustering(package = 'clusterr', df = Clustering::basketball,
min=3, max=4)

result[precision > 0.14 & dunn > 0.11]
```

Index

* datasets

- [basketball, 3](#)
 - [bolts, 5](#)
 - [stock, 18](#)
 - [stulong, 19](#)
 - [weather, 19](#)
- [\[.clustering, 20](#)
- [appClustering, 2](#)

- [basketball, 3](#)
- [best_ranked_external_metrics, 3](#)
- [best_ranked_internal_metrics, 4](#)
- [bolts, 5](#)

- [clustering, 6](#)

- [evaluate_best_validation_external_by_metrics, 9](#)
- [evaluate_best_validation_internal_by_metrics, 10](#)
- [evaluate_validation_external_by_metrics, 11](#)
- [evaluate_validation_internal_by_metrics, 12](#)
- [export_file_external, 13](#)
- [export_file_internal, 13](#)

- [plot_clustering, 14](#)

- [result_external_algorithm_by_metric, 15](#)
- [result_internal_algorithm_by_metric, 16](#)

- [sort.clustering, 17](#)
- [stock, 18](#)
- [stulong, 19](#)

- [weather, 19](#)