Package 'FeatureImpCluster'

May 7, 2026

Title Feature Importance for Partitional Clustering

Version 0.1.5

Description Implements a novel approach for measuring feature importance in k-means clustering. Importance of a feature is measured by the misclassification rate relative to the baseline cluster assignment due to a random permutation of feature values. An explanation of permutation feature importance in general can be found here: <`https://christophm.github.io/interpretable-ml-book/feature-importance.html`>.

License GPL-3

Encoding UTF-8

Suggests flexclust, clustMixType, knitr, rmarkdown, testthat, attempt, ClustImpute, covr

Imports ggplot2

RoxygenNote 7.1.1

Depends data.table

NeedsCompilation no

Author Oliver Pfaffel [aut, cre]

Maintainer Oliver Pfaffel <`opfaffel@gmail.com`>

Repository CRAN

Date/Publication 2021-10-20 16:20:02 UTC

Contents

create_random_data 2
FeatureImpCluster 2
PermMisClassRate 3
plot.featImpCluster 5

Index **6**

`create_random_data` *Create random data set with 4 clusters*

Description

Create random data set with 4 clusters in a 2 dimensional subspace of a nr_other_vars+2 dimensional space

Usage

```
create_random_data(n = 10000, nr_other_vars = 4)
```

Arguments

`n`	number of points
`nr_other_vars`	number of other variables / "noise" dimensions

Value

list containing the random data.table and a vector with the true underlying cluster assignments

Examples

```
create_random_data(n=1e3)
```

`FeatureImpCluster` *Feature importance for k-means clustering*

Description

This function loops through `PermMisClassRate` for each variable of the data. The mean misclassification rate over all iterations is interpreted as variable importance.

Usage

```
FeatureImpCluster(
  clusterObj,
  data,
  basePred = NULL,
  predFUN = NULL,
  sub = 1,
  biter = 10
)
```

Arguments

`clusterObj`	a "typical" cluster object. The only requirement is that there must be a prediction function which maps the data to an integer
`data`	data.table with the same features as the data set used for clustering (or the simply the same data)
`basePred`	should be equal to results of predFUN(clusterObj,newdata=data); this option saves time when data is a very large data set
`predFUN`	predFUN(clusterObj,newdata=data) should provide the cluster assignment as a numeric vector; typically this is a wrapper around a build-in prediction function
`sub`	integer between 0 and 1(=default), indicates that only a subset of the data should be used if <1
`biter`	the permutation is iterated biter(=5, default) times

Value

A list of

misClassRate A matrix of the permutation misclassification rate for each variable and each iteration

featureImp For each row of complete_data, the associated cluster

Examples

```
set.seed(123)
dat <- create_random_data(n=1e3)$data # random data

library(flexclust)
res <- kcca(dat,k=4)
f <- FeatureImpCluster(res,dat)
plot(f)
```

`PermMisClassRate` *Permutation misclassification rate for single variable*

Description

Answers the following question: Using the current partion as a baseline, what is the misclassification rate if a given feature is permuted?

Usage

```
PermMisClassRate(
  clusterObj,
  data,
  varName,
  basePred = NULL,
  predFUN = NULL,
  sub = 1,
  biter = 5,
  seed = 123
)
```

Arguments

`clusterObj`	a "typical" cluster object. The only requirement is that there must be a prediction function which maps the data to an integer
`data`	data.table with the same features as the data set used for clustering (or the simply the same data)
`varName`	character; variable name
`basePred`	should be equal to results of predFUN(clusterObj,newdata=data); this option saves time when data is a very large data set
`predFUN`	predFUN(clusterObj,newdata=data) should provide the cluster assignment as a numeric vector; typically this is a wrapper around a build-in prediction function
`sub`	integer between 0 and 1(=default), indicates that only a subset of the data should be used if <1
`biter`	the permutation is iterated biter(=5, default) times
`seed`	value for random seed

Value

vector of length biter with the misclassification rate

Examples

```
set.seed(123)
dat <- create_random_data(n=1e3)$data # random data

library(flexclust)
res <- kcca(dat,k=4)
PermMisClassRate(res,dat,varName="x")
```

`plot.featImpCluster` *Feature importance box plot*

Description

Feature importance box plot

Usage

```
## S3 method for class 'featImpCluster'
plot(x, dat = NULL, color = "none", showPoints = FALSE, ...)
```

Arguments

`x`	an object returned from FeatureImpCluster
`dat`	same data as used for the computation of the feature importance (only relevant for colored plots)
`color`	If set to "type", the plot will show different variable types with a different color.
`showPoints`	Show points (default is False)
`...`	arguments to be passed to base plot method

Value

Returns a ggplot2 object

Index

`create_random_data`, 2

`FeatureImpCluster`, 2

`PermMisClassRate`, *2*, 3
`plot.featImpCluster`, 5