

Package ‘kselection’

May 8, 2026

Type Package

Title Selection of K in K-Means Clustering

Version 0.2.1

Date 2022-05-16

Author Daniel Rodriguez

Maintainer Daniel Rodriguez <daniel.rodriguez.perez@gmail.com>

Description Selection of k in k-means clustering based on Pham et al. paper
``Selection of k in k-means clustering".

License GPL-3

URL <https://github.com/drodriguezperez/kselection>

BugReports <https://github.com/drodriguezperez/kselection/issues>

Imports tools

Suggests amap, FactoClass, foreach, testthat

Encoding UTF-8

RoxygenNote 7.2.0

NeedsCompilation no

Repository CRAN

Date/Publication 2022-05-16 19:50:02 UTC

Contents

kselection-package	2
get_f_k	2
get_k_threshold	3
kselection	4
num_clusters	6
num_clusters_all	7
set_k_threshold	8

Index	9
--------------	----------

kselection-package *Selection of K in K-Means Clustering*

Description

Selection of k in k-means clustering based on Pham et al. paper “Selection of k in k-means clustering”

Details

This package implements the method for selecting the number of clusters for the algorithm K-means introduced in the publication of Pham, Dimov and Nguyen of 2004.

Package: kselection
Version: 0.2.0
License: GPL-3

Author(s)

Daniel Rodriguez <daniel.rodriguez.perez@gmail.com>

References

D T Pham, S S Dimov, and C D Nguyen, "Selection of k in k-means clustering", Mechanical Engineering Science, 2004, pp. 103-119.

get_f_k *Get the $f(K)$ vector*

Description

Get the $f(K)$ vector.

Usage

```
get_f_k(obj)
```

Arguments

obj the output of kselection function.

Value

the vector of $f(K)$ function.

Author(s)

Daniel Rodriguez

See Also

[num_clusters](#), [num_clusters_all](#)

Examples

```
# Create a data set with two clusters
dat <- matrix(c(rnorm(100, 2, .1), rnorm(100, 3, .1),
               rnorm(100, -2, .1), rnorm(100, -3, .1)), 200, 2)

# Get the f(k) vector
sol <- kselection(dat)
f_k <- get_f_k(sol)
```

<code>get_k_threshold</code>	<i>Get the k_threshold</i>
------------------------------	----------------------------

Description

Get the maximum value of $f(K)$ from which can not be considered the existence of more than one cluster.

Usage

```
get_k_threshold(obj)
```

Arguments

`obj` the output of `kselection` function.

Value

the `k_threshold` value.

Author(s)

Daniel Rodriguez

See Also

[set_k_threshold](#)

Description

Selection of k in k -means clustering based on Pham et al. paper.

Usage

```
kselection(
  x,
  fun_cluster = stats::kmeans,
  max_centers = 15,
  k_threshold = 0.85,
  progressBar = FALSE,
  trace = FALSE,
  parallel = FALSE,
  ...
)
```

Arguments

<code>x</code>	numeric matrix of data, or an object that can be coerced to such a matrix.
<code>fun_cluster</code>	function to cluster by (e.g. <code>kmeans</code>). The first parameter of the function must be a numeric matrix and the second the number of clusters. The function must return an object with a named attribute <code>withinss</code> which is a numeric vector with the within.
<code>max_centers</code>	maximum number of clusters for evaluation.
<code>k_threshold</code>	maximum value of $f(K)$ from which can not be considered the existence of more than one cluster in the data set. The default value is 0.85.
<code>progressBar</code>	show a progress bar.
<code>trace</code>	display a trace of the progress.
<code>parallel</code>	If set to true, use <code>parallel foreach</code> to execute the function that implements the <code>kmeans</code> algorithm. Must register <code>parallel</code> before hand, such as <code>doMC</code> or others. Selecting this option the progress bar is disabled.
<code>...</code>	arguments to be passed to the <code>kmeans</code> method.

Details

This function implements the method proposed by Pham, Dimov and Nguyen for selecting the number of clusters for the K -means algorithm. In this method a function $f(K)$ is used to evaluate the quality of the resulting clustering and help decide on the optimal value of K for each data set. The $f(K)$ function is defined as

$$f(K) = \begin{cases} 1 & \text{if } K = 1 \\ \frac{S_K}{\alpha_K S_{K-1}} & \text{if } S_{K-1} \neq 0, \forall K > 1 \\ 1 & \text{if } S_{K-1} = 0, \forall K > 1 \end{cases}$$

where S_K is the sum of the distortion of all cluster and α_K is a weight factor which is defined as

$$\alpha_K = \begin{cases} 1 - \frac{3}{4N_d} & \text{if } K = 1 \text{ and } N_d > 1 \\ \alpha_{K-1} + \frac{1-\alpha_{K-1}}{6} & \text{if } K > 2 \text{ and } N_d > 1 \end{cases}$$

where N_d is the number of dimensions of the data set.

In this definition $f(K)$ is the ratio of the real distortion to the estimated distortion and decreases when there are areas of concentration in the data distribution.

The values of K that yield $f(K) < 0.85$ can be recommended for clustering. If there is not a value of K which $f(K) < 0.85$, it cannot be considered the existence of clusters in the data set.

Value

an object with the $f(K)$ results.

Author(s)

Daniel Rodriguez

References

D T Pham, S S Dimov, and C D Nguyen, "Selection of k in k-means clustering", Mechanical Engineering Science, 2004, pp. 103-119.

See Also

[num_clusters](#), [get_f_k](#)

Examples

```
# Create a data set with two clusters
dat <- matrix(c(rnorm(100, 2, .1), rnorm(100, 3, .1),
               rnorm(100, -2, .1), rnorm(100, -3, .1)), 200, 2)

# Execute the method
sol <- kselection(dat)

# Get the results
k <- num_clusters(sol) # optimal number of clusters
f_k <- get_f_k(sol)    # the f(K) vector

# Plot the results
plot(sol)

## Not run:
# Parallel
require(doMC)
registerDoMC(cores = 4)

system.time(kselection(dat, max_centers = 50 , nstart = 25))
system.time(kselection(dat, max_centers = 50 , nstart = 25, parallel = TRUE))
```

```
## End(Not run)
```

num_clusters	<i>Get the number of clusters.</i>
--------------	------------------------------------

Description

The optimal number of clusters proposed by the method.

Usage

```
num_clusters(obj)
```

Arguments

obj the output of kselection function.

Value

the number of clusters proposed.

Author(s)

Daniel Rodriguez

See Also

[num_clusters_all](#), [get_f_k](#)

Examples

```
# Create a data set with two clusters
dat <- matrix(c(rnorm(100, 2, .1), rnorm(100, 3, .1),
               rnorm(100, -2, .1), rnorm(100, -3, .1)), 200, 2)

# Get the optimal number of clusters
sol <- kselection(dat)
k <- num_clusters(sol)
```

num_clusters_all	<i>Get all recommended numbers of clusters</i>
------------------	--

Description

The number of cluster which could be recommender according the method threshold.

Usage

```
num_clusters_all(obj)
```

Arguments

obj the output of kselection function.

Value

an array of number of clusters that could be recommended.

Author(s)

Daniel Rodriguez

See Also

[num_clusters](#), [get_f_k](#)

Examples

```
# Create a data set with two clusters
dat <- matrix(c(rnorm(100, 2, .1), rnorm(100, 3, .1),
               rnorm(100, -2, .1), rnorm(100, -3, .1)), 200, 2)

# Get the optimal number of clustes
sol <- kselection(dat)
k   <- num_clusters(sol)
```

set_k_threshold	<i>Set the k_threshold</i>
-----------------	----------------------------

Description

Set the maximum value of $f(K)$ from which can not be considered the existence of more than one cluster.

Usage

```
set_k_threshold(obj, k_threshold)
```

Arguments

obj	the output of kselection function.
k_threshold	maximum value of $f(K)$ from which can not be considered the existence of more than one cluster in the data set.

Value

the output of kselection function with new k_threshold.

Author(s)

Daniel Rodriguez

See Also

[get_k_threshold](#)

Index

`get_f_k`, [2](#), [5–7](#)

`get_k_threshold`, [3](#), [8](#)

`kselection`, [4](#)

`kselection-package`, [2](#)

`num_clusters`, [3](#), [5](#), [6](#), [7](#)

`num_clusters_all`, [3](#), [6](#), [7](#)

`set_k_threshold`, [3](#), [8](#)