

# Package ‘fastkmedoids’

May 8, 2026

**Type** Package

**Title** Faster K-Medoids Clustering Algorithms: FastPAM, FastCLARA, FastCLARANS

**Version** 1.6

**Date** 2026-03-25

**Description** R wrappers of C++ implementation of Faster K-Medoids clustering algorithms (FastPAM, FastCLARA and FastCLARANS) proposed in Erich Schubert, Peter J. Rousseeuw 2019 <[doi:10.1007/978-3-030-32047-8\\_16](https://doi.org/10.1007/978-3-030-32047-8_16)>.

**Depends** methods

**License** GPL (>= 2)

**Collate** fastkmedoids-package.R RcppExports.R KmedoidsResult.R

**Encoding** UTF-8

**Imports** Rcpp (>= 1.0.1)

**LinkingTo** Rcpp

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**RoxygenNote** 7.3.2

**NeedsCompilation** yes

**Author** Xun Li [aut, cre]

**Maintainer** Xun Li <[lixun910@gmail.com](mailto:lixun910@gmail.com)>

**Repository** CRAN

**Date/Publication** 2026-03-26 13:50:02 UTC

## Contents

fastclara . . . . .	2
fastclarans . . . . .	3
fastpam . . . . .	4
KmedoidsResult-class . . . . .	5
pam . . . . .	5

<b>Index</b>	<b>6</b>
--------------	----------

fastclara

*FastCLARA***Description**

Clustering Large Applications (CLARA) with the improvements, to increase scalability in the number of clusters. This variant will also default to twice the sample size, to improve quality. (Schubert and Rousseeuw, 2019)

**Usage**

```
fastclara(
  rdist,
  n,
  k,
  maxiter = 0L,
  initializer = "LAB",
  fasttol = 1,
  numsamples = 5L,
  sampling = 0.25,
  independent = FALSE,
  seed = 123456789L
)
```

**Arguments**

<code>rdist</code>	The distance matrix (lower triangular matrix, column wise storage)
<code>n</code>	The number of observations
<code>k</code>	The number of clusters to produce
<code>maxiter</code>	The maximum number of iterations (default: 0)
<code>initializer</code>	Initializer: either "BUILD" (used in classic PAM) or "LAB" (linear approximative BUILD)
<code>fasttol</code>	Tolerance for fast swapping behavior (may perform worse swaps). Default: 1.0, which means to perform any additional swap that gives an improvement. When set to 0, it will only execute an additional swap if it appears to be independent (i.e., the improvements resulting from the swap have not decreased when the first swap was executed).
<code>numsamples</code>	Number of samples to draw (i.e. iterations). Default: 5
<code>sampling</code>	Sampling rate. Default value: $80 + 4 * k$ . (see Schubert and Rousseeuw, 2019) If less than 1, it is considered to be a relative value. e.g. $N * 0.10$
<code>independent</code>	NOT Keep the previous medoids in the next sample. Default: FALSE
<code>seed</code>	Seed for random number generator. Default: 123456789

**Value**

KMedoids S4 class

**References**

Erich Schubert, Peter J. Rousseeuw "Faster k-Medoids Clustering: Improving the PAM, CLARA, and CLARANS Algorithms" 2019 <https://arxiv.org/abs/1810.05691>

---

fastclarans

*FastCLARANS*

---

**Description**

A faster variation of CLARANS, that can explore  $O(k)$  as many swaps at a similar cost by considering all medoids for each candidate non-medoid. Since this means sampling fewer non-medoids, we suggest to increase the subsampling rate slightly to get higher quality than CLARANS, at better runtime. (Schubert and Rousseeuw, 2019)

**Usage**

```
fastclarans(rdist, n, k, numlocal = 2L, maxneighbor = 0.025, seed = 123456789L)
```

**Arguments**

rdist	The distance matrix (lower triangular matrix, column wise storage)
n	The number of observations
k	The number of clusters to produce.
numlocal	Number of samples to draw (i.e. restarts). Default: 2
maxneighbor	Sampling rate. If less than 1, it is considered to be a relative value. Default: 2 * 0.0125, larger sampling rate than CLARANS (see Schubert and Rousseeuw, 2019)
seed	Seed for random number generator. Default: 123456789

**Value**

KMedoids S4 class

**References**

Erich Schubert, Peter J. Rousseeuw "Faster k-Medoids Clustering: Improving the PAM, CLARA, and CLARANS Algorithms" 2019 <https://arxiv.org/abs/1810.05691>

---

fastpam	<i>FastPAM</i>
---------	----------------

---

### Description

FastPAM: An improved version of PAM, that is usually  $O(k)$  times faster. Because of the speed benefits, we also suggest to use a linear-time initialization, such as the k-means++ initialization or the proposed LAB (linear approximative BUILD, the third component of FastPAM) initialization, and try multiple times if the runtime permits. (Schubert and Rousseeuw, 2019)

### Usage

```
fastpam(
  rdist,
  n,
  k,
  maxiter = 0L,
  initializer = "LAB",
  fasttol = 1,
  seed = 123456789L
)
```

### Arguments

<code>rdist</code>	The distance matrix (lower triangular matrix, column wise storage)
<code>n</code>	The number of observations
<code>k</code>	The number of clusters to produce.
<code>maxiter</code>	The maximum number of iterations (default: 0)
<code>initializer</code>	Initializer: either "BUILD" (used in classic PAM) or "LAB" (linear approximative BUILD) Because of the speed benefits, "LAB" is suggested, and one can try multiple times if the runtime permits.
<code>fasttol</code>	Tolerance for fast swapping behavior (may perform worse swaps). Default: 1.0, which means to perform any additional swap that gives an improvement. When set to 0, it will only execute an additional swap if it appears to be independent (i.e., the improvements resulting from the swap have not decreased when the first swap was executed).
<code>seed</code>	Seed for random number generator. Default: 123456789

### Value

KMedoids S4 class

### References

Erich Schubert, Peter J. Rousseeuw "Faster k-Medoids Clustering: Improving the PAM, CLARA, and CLARANS Algorithms" 2019 <https://arxiv.org/abs/1810.05691>

---

KmedoidsResult-class *An S4 class to represent the result of kmedoids clustering*

---

### Description

An S4 class to represent the result of kmedoids clustering

### Slots

cost The cost value of kmedoids clustering

medoids The medoids of kmedoids clustering

assignment The assignment of which cluster each observation belongs to of kmedoids clustering

---

pam *PAM (Partitioning Around Medoids)*

---

### Description

The original Partitioning Around Medoids (PAM) algorithm or k-medoids clustering, as proposed by Kaufman and Rousseeuw; a largely equivalent method was also proposed by Whitaker in the operations research domain, and is well known by the name "fast interchange" there. (Schubert and Rousseeuw, 2019)

### Usage

```
pam(rdist, n, k, maxiter = 0L)
```

### Arguments

rdist The distance matrix (lower triangular matrix, column wise storage)

n The number of observations

k The number of clusters to produce

maxiter The maximum number of iterations (default: 0)

### Value

KMedoids S4 class

### References

L. Kaufman, P. J. Rousseeuw "Clustering by means of Medoids" Information Systems and Operational Research 21(2)

# Index

fastclara, [2](#)

fastclarans, [3](#)

fastpam, [4](#)

KmedoidsResult-class, [5](#)

pam, [5](#)