

# Package: CompositionalClust (via r-universe)

June 5, 2026

**Type** Package

**Title** Clustering with Compositional Data

**Version** 1.3

**Date** 2026-06-05

**Author** Michail Tsagris [aut, cre], Nikolaos Kontemeniotis [aut]

**Maintainer** Michail Tsagris <mtsagris@uoc.gr>

**Depends** R (>= 4.0)

**Imports** Compositional, factoextra, graphics, lowmemtkmeans, mixture, parallel, Rfast, Rfast2, stats

**Description** Cluster analysis with compositional data using the alpha--transformation. Relevant papers include: Tsagris M. and Kontemeniotis N. (2025). Lobachevskii Journal of Mathematics <doi:10.1134/S1995080225613700>. Tsagris M.T., Preston S. and Wood A.T.A. (2011), <doi:10.48550/arXiv.1106.1451>. Garcia-Escudero Luis A., Gordaliza Alfonso, Matran Carlos, Mayo-Iscar Agustin. (2008), <doi:10.1214/07-AOS515>.

**License** GPL (>= 2)

**NeedsCompilation** no

**Repository** <https://cran.r-universe.dev>

**Date/Publication** 2026-06-05 11:40:20 UTC

**RemoteUrl** <https://github.com/cran/CompositionalClust>

**RemoteRef** HEAD

**RemoteSha** d3c6eb855dcddd104152d417167f4aae8cff8957

## Contents

CompositionalClust-package . . . . .	2
alfa.cikmeans . . . . .	3
alfa.kmeans . . . . .	4
cikmeans . . . . .	6
clust.plot . . . . .	7

Gaussian mixture models for compositional data using the alpha-transformation . . . . .	8
index_max . . . . .	10
index_min . . . . .	11
Mixture model selection with the alpha-transformation using BIC . . . . .	12
The alpha-transformation . . . . .	14
<b>Index</b>	<b>15</b>

---

CompositionalClust-package

*Clustering with Compositional Data*

---

## Description

Cluster analysis with compositional data using the  $\alpha$ -transformation.

## Details

Package: CompositionalClust  
 Type: Package  
 Version: 1.3  
 Date: 2026-06-05  
 License: GPL-2

## Maintainers

Michail Tsagris <mtsagris@uoc.gr>

## Author(s)

Michail Tsagris <mtsagris@uoc.gr> and Nikolaos Kontemeniotis <kontemeniotisn@gmail.com>.

## References

- Tsagris M. and Kontemeniotis N. (2025). Simplicial clustering using the  $\alpha$ -transformation. *Lobachevskii Journal of Mathematics*, 46(12): 6471–6482. <https://arxiv.org/pdf/2509.05945>.
- Alenazi A. (2023). A review of compositional data analysis and recent advances. *Communications in Statistics–Theory and Methods*, 52(16): 5535–5567.
- Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. *Fourth International International Workshop on Compositional Data Analysis*.

---

alfa.cikmeans	<i>Cluster indices for the <math>K</math>-means algorithm for compositional data using the <math>\alpha</math>-transformation</i>
---------------	---

---

### Description

Cluster indices for the  $K$ -means algorithm for compositional data using the  $\alpha$ -transformation.

### Usage

```
alfa.cikmeans(x, ncl = 10, trim = 0, a = seq(-1, 1, by = 0.1), max.iters = 50,
nstart = 10)
```

### Arguments

x	A matrix with the compositional data.
ncl	The maximum number of clusters to try. The minimum number of clusters is 2.
trim	A number in $[0, 1)$ . If $\text{trim} = 0$ , then the classical $K$ -means algorithm is performed. If you chose a number higher than 0 then the trimmed $K$ -means of Garcia-Escudero et al. (2008) is performed.
a	A vector with a grid of values of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $a=0$ , the isometric log-ratio transformation is applied.
max.iters	The maximum number of iterations allowed during the $K$ -means algorithm.
nstart	How many random starts to perform?

### Details

The  $\alpha$ -transformation is applied to the compositional data, and then the  $K$ -means algorithm is performed and a series of cluster validity indices are computed.

### Value

A list including:

min_crit	A matrix with 9 columns and at least one row, where each column contains the value of a cluster validity index, whose minimal value is preferred. Each row corresponds to a specific number of clusters, starting from 2 up to ncl.
best_min	The number of clusters selected based upon the minimal valued cluster validity indices.
max_crit	A matrix with 24 columns and at least one row, where each column contains the value of a cluster validity index, whose minimal value is preferred. Each row corresponds to a specific number of clusters, starting from 2 up to ncl.
best_max	The number of clusters selected based upon the maximal valued cluster validity indices.

**cluster** If the argument "all" is TRUE, then the clustering indices of each observation for each number of clusters will be returned in a matrix, where each column corresponds to the clustering of each number of clusters.

### Author(s)

Michail Tsagris.

R implementation and documentation: Michail Tsagris <mtsagris@uoc.gr>.

### References

Tsagris M. and Kontemeniotis N. (2025). Simplicial clustering using the  $\alpha$ -transformation. *Lobachevskii Journal of Mathematics*, 46(12): 6471–6482. <https://arxiv.org/pdf/2509.05945>.

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In *Proceedings of the 4th Compositional Data Analysis Workshop*, Girona, Spain. <https://arxiv.org/pdf/1106.1451.pdf>

Garcia-Escudero Luis A., Gordaliza Alfonso, Matran Carlos, Mayo-Iscar Agustin. (2008). A general trimming approach to robust cluster analysis. *Annals of Statistics* 36(3): 1324–1345.

### See Also

[cikmeans](#)

### Examples

```
y <- as.matrix(iris[, 1:4])
y <- y / rowSums(y)
mod <- alfa.cikmeans( y, ncl = 5, a = c(0, 0.5, 1) )
```

---

alfa.kmeans	<i>The K-means algorithm for compositional data using the <math>\alpha</math>-transformation</i>
-------------	--

---

### Description

The  $K$ -means algorithm for compositional data using the  $\alpha$ -transformation.

### Usage

```
alfa.kmeans(x, ncl = 10, trim = 0, a = seq(-1, 1, by = 0.1), max.iters = 50,
nstart = 10)
```

**Arguments**

<code>x</code>	A matrix with the compositional data.
<code>nc1</code>	The maximum number of clusters to try. The minimum number of clusters is 2.
<code>trim</code>	A number in $[0, 1)$ . If <code>trim = 0</code> , then the classical $K$ -means algorithm is performed. If you chose a number higher than 0 then the trimmed $K$ -means of Garcia-Escudero et al. (2008) is performed.
<code>a</code>	A vector with a grid of values of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If <code>a=0</code> , the isometric log-ratio transformation is applied.
<code>max.iter</code>	The maximum number of iterations allowed during the $K$ -means algorithm.
<code>nstart</code>	How many random starts to perform?

**Details**

The  $\alpha$ -transformation is applied to the compositional data, and then the  $K$ -means algorithm is performed.

**Value**

A list with the results of the `kmeans` function for each value of  $\alpha$ .

**Author(s)**

Michail Tsagris.

R implementation and documentation: Michail Tsagris <[mtsagris@uoc.gr](mailto:mtsagris@uoc.gr)>.

**References**

Tsagris M. and Kontemeniotis N. (2025). Simplicial clustering using the  $\alpha$ -transformation. *Lobachevskii Journal of Mathematics*, 46(12): 6471–6482. <https://arxiv.org/pdf/2509.05945>.

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In *Proceedings of the 4th Compositional Data Analysis Workshop*, Girona, Spain. <https://arxiv.org/pdf/1106.1451.pdf>

Garcia-Escudero Luis A., Gordaliza Alfonso, Matran Carlos, Mayo-Iscar Agustin. (2008). A general trimming approach to robust cluster analysis. *Annals of Statistics* 36(3): 1324–1345.

**See Also**

[cikmeans](#)

**Examples**

```
y <- as.matrix(iris[, 1:4])
y <- y / rowSums(y)
mod <- alfa.kmeans( y, nc1 = 5, a = c(0, 0.5, 1) )
```

---

 cikmeans

*The K-means algorithm with cluster indices computed*


---

### Description

The  $K$ -means algorithm with cluster indices computed.

### Usage

```
cikmeans(y, ncl = 10, trim = 0, max.iters = 50, nstart = 10, all = FALSE)
```

### Arguments

<code>y</code>	A matrix with numerical data.
<code>ncl</code>	The maximum number of clusters to try. The minimum number of clusters is 2.
<code>trim</code>	A number in $[0, 1)$ . If <code>trim = 0</code> , then the classical $K$ -means algorithm is performed. If you chose a number higher than 0 then the trimmed $K$ -means of Garcia-Escudero et al. (2008) is performed.
<code>max.iters</code>	The maximum number of iterations allowed during the $K$ -means algorithm.
<code>nstart</code>	How many random starts to perform?
<code>all</code>	If this is TRUE, then the clustering indices of each observation for each number of clusters will be returned.

### Details

The  $K$ -means algorithm is performed and a series of cluster validity indices are computed.

### Value

A list including:

<code>min_crit</code>	A matrix with 9 columns and at least one row, where each column contains the value of a cluster validity index, whose minimal value is preferred. Each row corresponds to a specific number of clusters, starting from 2 up to <code>ncl</code> .
<code>best_min</code>	The number of clusters selected based upon the minimal valued cluster validity indices.
<code>max_crit</code>	A matrix with 24 columns and at least one row, where each column contains the value of a cluster validity index, whose minimal value is preferred. Each row corresponds to a specific number of clusters, starting from 2 up to <code>ncl</code> .
<code>best_max</code>	The number of clusters selected based upon the maximal valued cluster validity indices.
<code>cluster</code>	If the argument "all" is TRUE, then the clustering indices of each observation for each number of clusters will be returned in a matrix, where each column corresponds to the clustering of each number of clusters.

**Author(s)**

Michail Tsagris and Nikolaos Kontemeniotis.

R implementation and documentation: Michail Tsagris <mtsagris@uoc.gr> and Nikolaos Kontemeniotis <kontemeniotisn@gmail.com>..

**References**

Garcia-Escudero Luis A., Gordaliza Alfonso, Matran Carlos, Mayo-Iscar Agustin. (2008). A general trimming approach to robust cluster analysis. *Annals of Statistics* 36(3): 1324–1345.

**See Also**

[index\\_min](#), [index\\_max](#), [alfa.cikmeans](#)

**Examples**

```
y <- as.matrix(iris[, 1:4])
y <- y / rowSums(y)
mod <- cikmeans(y, ncl = 5)
```

---

clust.plot

*Visualization of the K-means algorithm results*

---

**Description**

Visualization of the  $K$ -means algorithm results.

**Usage**

```
clust.plot(mod, x)
```

**Arguments**

mod	The output of the kmeans() function.
x	A matrix with the data.

**Details**

The function performs PCA and plots the data onto the first two dimensions, constructs the convex hull of the groups and plots them with different colours.

**Value**

A plot.

**Author(s)**

Michail Tsagris.

R implementation and documentation: Michail Tsagris <mtsagris@uoc.gr>.

**References**

Tsagris M. and Kontemeniotis N. (2025). Simplicial clustering using the  $\alpha$ -transformation. Lobachevskii Journal of Mathematics (Accepted for publication). <https://arxiv.org/pdf/2509.05945>.

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. <https://arxiv.org/pdf/1106.1451.pdf>

Garcia-Escudero Luis A., Gordaliza Alfonso, Matran Carlos, Mayo-Iscar Agustin. (2008). A general trimming approach to robust cluster analysis. Annals of Statistics 36(3): 1324–1345.

**See Also**

[cikmeans](#)

**Examples**

```
y <- as.matrix(iris[, 1:4])
y <- y / rowSums(y)
z <- CompositionalClust::alfa(y, 1)
mod <- alfa.kmeans( z, ncl = 3, a = 1 )
clust.plot(mod[[ 1 ]], z )
```

---

Gaussian mixture models for compositional data using the alpha-transformation

*Gaussian mixture models for compositional data using the  $\alpha$ -transformation*

---

**Description**

Gaussian mixture models for compositional data using the  $\alpha$ -transformation.

**Usage**

```
alfa.mix.norm(x, g, a, model, veo = FALSE)
```

**Arguments**

x	A matrix with the compositional data.
g	How many clusters to create.
a	The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$ the isometric log-ratio transformation is applied.

model	<p>The type of model to be used.</p> <ol style="list-style-type: none"> <li>"EII": All groups have the same diagonal covariance matrix, with the same variance for all variables.</li> <li>"VII": Different diagonal covariance matrices, with the same variance for all variables within each group.</li> <li>"EEI": All groups have the same diagonal covariance matrix.</li> <li>"VEI": Different diagonal covariance matrices. If we make all covariance matrices have determinant 1, (divide the matrix with the <math>p</math>-th root of its determinant) then all covariance matrices will be the same.</li> <li>"EVI": Different diagonal covariance matrices with the same determinant.</li> <li>"VVI": Different diagonal covariance matrices, with nothing in common.</li> <li>"EEE": All covariance matrices are the same.</li> <li>"EEV": Different covariance matrices, but with the same determinant and in addition, if we make them have determinant 1, they will have the same trace.</li> <li>"VEV": Different covariance matrices but if we make the matrices have determinant 1, then they will have the same trace.</li> <li>"VVV": Different covariance matrices with nothing in common.</li> <li>"EVE": Different covariance matrices, but with the same determinant. In addition, calculate the eigenvectors for each covariance matrix and you will see the extra similarities.</li> <li>"VVE": Different covariance matrices, but they have something in common with their directions. Calculate the eigenvectors of each covariance matrix and you will see the similarities.</li> <li>"VEE": Different covariance matrices, but if we make the matrices have determinant 1, then they will have the same trace. In addition, calculate the eigenvectors for each covariance matrix and you will see the extra similarities.</li> <li>"EVV": Different covariance matrices, but with the same determinant.</li> </ol>
veo	<p>Stands for "Variables exceed observations". If TRUE then if the number variables in the model exceeds the number of observations, but the model is still fitted.</p>

### Details

A log-ratio transformation is applied and then a Gaussian mixture model is constructed.

### Value

A list including:

mu	A matrix where each row corresponds to the mean vector of each cluster.
su	An array containing the covariance matrix of each cluster.
prob	The estimated mixing probabilities.
est	The estimated cluster membership values.

**Author(s)**

Michail Tsagris.

R implementation and documentation: Michail Tsagris <mtsagris@uoc.gr>.

**References**

Tsagris M. and Kontemeniotis N. (2025). Simplicial clustering using the  $\alpha$ -transformation. *Lobachevskii Journal of Mathematics*, 46(12): 6471–6482. <https://arxiv.org/pdf/2509.05945>.

Ryan P. Browne, Aisha ElSherbiny and Paul D. McNicholas (2015). R package mixture: Mixture Models for Clustering and Classification.

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In *Proceedings of the 4th Compositional Data Analysis Workshop*, Girona, Spain. <https://arxiv.org/pdf/1106.1451.pdf>

**See Also**

[bic.alfamixnorm](#)

**Examples**

```
x <- as.matrix(iris[, 1:4])
x <- x/ rowSums(x)
mod1 <- alfa.mix.norm(x, 3, 0.4, model = "EII" )
mod2 <- alfa.mix.norm(x, 4, 0.7, model = "VII")
```

---

index\_max

*Cluster indices (maximal valued) for the  $K$ -means algorithm*

---

**Description**

Cluster indices (maximal valued) for the  $K$ -means algorithm.

**Usage**

```
index_max(y, mod)
```

**Arguments**

**y** A matrix with numerical data.  
**mod** An object with the result of the [kmeans](#) function.

**Details**

A series of cluster validity indices (maximal valued) are computed.

**Value**

A vector with 24 cluster validity indices.

**Author(s)**

Michail Tsagris.

R implementation and documentation: Michail Tsagris <mtsagris@uoc.gr>.

**See Also**

[index\\_min](#), [cikmeans](#), [alfa.cikmeans](#)

**Examples**

```
y <- as.matrix(iris[, 1:4])
mod <- kmeans(y, 3)
mod <- index_max(y, mod)
```

---

index\_min

*Cluster indices (minimal valued) for the K-means algorithm*

---

**Description**

Cluster indices (minimal valued) for the  $K$ -means algorithm.

**Usage**

```
index_min(y, mod)
```

**Arguments**

**y** A matrix with numerical data.  
**mod** An object with the result of the [kmeans](#) function.

**Details**

A series of cluster validity indices (minimal valued) are computed.

**Value**

A vector with 9 cluster validity indices.

**Author(s)**

Michail Tsagris.

R implementation and documentation: Michail Tsagris <mtsagris@uoc.gr>.

**See Also**

[index\\_max](#), [cikmeans](#), [alfa.cikmeans](#)

**Examples**

```
y <- as.matrix(iris[, 1:4])
mod <- kmeans(y, 3)
mod <- index_min(y, mod)
```

---

Mixture model selection with the alpha-transformation using BIC

*Mixture model selection with the  $\alpha$ -transformation using BIC*

---

**Description**

Mixture model selection with the  $\alpha$ -transformation using BIC.

**Usage**

```
bic.alfamixnorm(x, G, a = seq(-1, 1, by = 0.1), veo = FALSE, graph = TRUE)
```

**Arguments**

x	A matrix with compositional data.
G	A numeric vector with the number of components, clusters, to be considered, e.g. 1:3.
a	A vector with a grid of values of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$ the isometric log-ratio transformation is applied.
veo	Stands for "Variables exceed observations". If TRUE then if the number variables in the model exceeds the number of observations, but the model is still fitted.
graph	A boolean variable, TRUE or FALSE specifying whether a graph should be drawn or not.

**Details**

The  $\alpha$ -transformation is applied to the compositional data first and then mixtures of multivariate Gaussian distributions are fitted. BIC is used to decide on the optimal model and number of components.

**Value**

A list including:

<code>abic</code>	A list that contains the matrices of all BIC values for all values of $\alpha$ .
<code>optalpha</code>	The value of $\alpha$ that leads to the highest BIC.
<code>optG</code>	The number of components with the highest BIC.
<code>optmodel</code>	The type of model corresponding to the highest BIC.

If `graph` is set equal to `TRUE` a plot with the BIC of the best model for each number of components versus the number of components and a list with the results of the Gaussian mixture model for each value of  $\alpha$ .

**Author(s)**

Michail Tsagris.

R implementation and documentation: Michail Tsagris <[mtsagris@uoc.gr](mailto:mtsagris@uoc.gr)>.

**References**

- Tsagris M. and Kontemeniotis N. (2025). Simplicial clustering using the  $\alpha$ -transformation. *Lobachevskii Journal of Mathematics*, 46(12): 6471–6482. <https://arxiv.org/pdf/2509.05945>.
- Ryan P. Browne, Aisha ElSherbiny and Paul D. McNicholas (2018). `mixture`: Mixture Models for Clustering and Classification. R package version 1.5.
- Ryan P. Browne and Paul D. McNicholas (2014). Estimating Common Principal Components in High Dimensions. *Advances in Data Analysis and Classification*, 8(2), 217-226.
- Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In *Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain*. <https://arxiv.org/pdf/1106.1451.pdf>

**See Also**

[alfa.mix.norm](#)

**Examples**

```
x <- as.matrix( iris[, 1:4] )
x <- x/ rowSums(x)
bic.alfamixnorm(x, 1:3, a = c(0.4, 0.5, 0.6), graph = FALSE)
```

---

The alpha-transformation

*The  $\alpha$ -transformation*

---

### Description

The  $\alpha$ -transformation.

### Usage

```
alfa(x, a)
```

### Arguments

**x** A matrix with the compositional data.

**a** The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If  $\alpha = 0$  the isometric log-ratio transformation is applied.

### Details

The  $\alpha$ -transformation is applied to the compositional data.

### Value

The  $\alpha$ -transformed data.

### Author(s)

Michail Tsagris.

R implementation and documentation: Michail Tsagris <mtsagris@uoc.gr>.

### References

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. <https://arxiv.org/pdf/1106.1451.pdf>

### See Also

[alfa.kmeans](#)

### Examples

```
y <- as.matrix(iris[, 1:4])
y <- y / rowSums(y)
ya <- alfa(y, 0.2)
```

# Index

alfa (The alpha-transformation), [14](#)  
alfa.cikmeans, [3](#), [7](#), [11](#), [12](#)  
alfa.kmeans, [4](#), [14](#)  
alfa.mix.norm, [13](#)  
alfa.mix.norm (Gaussian mixture models  
for compositional data using  
the alpha-transformation), [8](#)

bic.alfamixnorm, [10](#)  
bic.alfamixnorm (Mixture model  
selection with the  
alpha-transformation using  
BIC), [12](#)

cikmeans, [4](#), [5](#), [6](#), [8](#), [11](#), [12](#)  
clust.plot, [7](#)  
CompositionalClust-package, [2](#)

Gaussian mixture models for  
compositional data using the  
alpha-transformation, [8](#)

index\_max, [7](#), [10](#), [12](#)  
index\_min, [7](#), [11](#), [11](#)

kmeans, [10](#), [11](#)

Mixture model selection with the  
alpha-transformation using  
BIC, [12](#)

The alpha-transformation, [14](#)