Package 'isopam'

October 7, 2026

Type Package

Title Clustering of Sites with Species Data

Version 3.7

Maintainer Sebastian Schmidtlein <schmidtlein@kit.edu>

Imports vegan, cluster, fastkmedoids, future, future.apply, igraph, methods, ps, RSpectra, grDevices, graphics, stats, utils, proxy, ggplot2, tibble

Description Clustering algorithm developed for use with plot inventories of species. It groups plots by subsets of diagnostic species rather than overall species composition. There is an unsupervised and a supervised mode, the latter accepting suggestions for species with greater weight and cluster medoids.

License GPL (>= 2)

Encoding UTF-8

NeedsCompilation no

Author Sebastian Schmidtlein [aut, cre] (ORCID: <https://orcid.org/0000-0003-1888-1865>), Jason Collison [aut], Robin Pfannendoerfer [aut], Lubomir Tichy [ctb]

Repository CRAN

Date/Publication 2026-10-06 22:00:14 UTC

Depends R (>= 3.5.0)

Contents

andechs 2
clusters 2
isopam 3
isotab 8
plot.isotab 10

Index **12**

andechs *Fen Meadows*

Description

Average cover of vascular plant species in subplots nested within 17 whole-plots from mown fen meadows. This is a subset of the data used in Schmidtlein & Sassin (2004).

Usage

```
data(andechs)
```

Format

A matrix containing 17 plot observations with 110 species.

Source

Schmidtlein, S., Sassin, J. (2004): Mapping of continuous floristic gradients in grasslands using hyperspectral imagery. *Remote Sensing of Environment* **92**, 126–138.

clusters *Returns clusters of an Isopam clustering*

Description

Provides a vector or data frame with cluster affiliations determined using Isopam.

Usage

```
clusters(x, ...)

## S3 method for class 'isopam'
clusters(x, level = NULL, k = NULL,
         style = c("flat", "hierarchical"), ...)
```

Arguments

x	Object of class isopam.
level	An integer scalar or vector with the desired cluster level(s). Level numbers start with 1 for the first division.
k	An integer scalar or vector with the desired number(s) of groups
style	Whether the labels of the clusters are consecutive ('flat') or nested ('hierarchical', i.e. 1.1, 1.2 etc.).
...	Additional arguments (currently unused).

Value

Factor vector or data frame with the cluster assignments.

Author(s)

Sebastian Schmidtlein

See Also

isopam, isotab

Examples

```
## load data to the current environment
data(andechs)

## call isopam with the standard options
ip <-isopam(andechs)

## return clusters
clusters(ip)

## clusters of level 2, with labels reflecting the hierarchy
clusters(ip, 2, style = "hier")

## cluster solution with 3 classes
clusters(ip, k = 3)
```

---

isopam *Isopam (Clustering)*

Description

Isopam classification is performed either as a hierarchical, divisive method or as non-hierarchical partitioning. Isopam is designed for matrices representing species abundances in plots and with a diagnostic species approach in mind. It optimises clusters and cluster numbers for concentration of indicative species in groups. Predefined indicative species and cluster medoids can optionally be added for fully or semi-supervised classification.

Usage

```
isopam(dat, c.fix = NULL, c.max = NULL, l.max = FALSE, stopat = c(1,7),
       sieve = TRUE, Gs = 3.5, ind = NULL, centers = NULL,
       distance = 'bray', k.max = 100, d.max = 7, juice = FALSE,
       polishing = c('strict', 'relaxed'), ...)

## S3 method for class 'isopam'
identify(x, ...)
```

```
     ## S3 method for class 'isopam'
plot(x, ...)
     ## S3 method for class 'isopam'
summary(object, ...)
     ## S3 method for class 'isopam'
print(x, ...)
```

Arguments

dat	data matrix: each row corresponds to an object (typically a plot), each column corresponds to a descriptor (typically a species). All variables must be numeric. Missing values (NAs) are not allowed. At least 3 rows (plots) are required.
c.fix	number of clusters (defaults to NULL). If a number is given, non-hierarchical partitioning is performed, c.max is ignored and l.max is set to one. In combination with centers, if c.fix exceeds the number of provided centers, semi-supervised mode is activated: the provided centers remain fixed while additional clusters are formed to reach the specified total (see Details).
c.max	maximum number of clusters per partition. Applies to all splits. Defaults to 6 unless centers is provided, in which case it defaults to the number of centers. If explicitly set to a value greater than the number of provided centers (with c.fix = NULL), semi-supervised search mode is activated: the algorithm searches for the optimal number of additional clusters (see Details).
l.max	maximum number of hierarchy levels. Defaults to FALSE (no maximum number). Note that divisions may stop well before this number is reached (see stopat). Use l.max = 1 for non-hierarchical partitioning (or use c.fix).
stopat	vector with stopping rules for hierarchical clustering. Two values define if a partition should be retained in hierarchical clustering: the first determines how many indicator species must be present per cluster, the second defines the standardized G-value that must be reached by these indicators. stopat is not effective at the first hierarchy level or in non-hierarchical partitioning.
sieve	logical. If TRUE (the deafult), only species exceeding a threshold defined by Gs are used in the search for a good clustering solution. Their number is multiplied with their mean standardized G-value. The product is used as optimality criterion. If FALSE all species are used for optimization.
Gs	threshold (standardized G value) for species to be considered in the search for a good clustering solution. Effective with sieve = TRUE.
ind	optional vector of column names from dat defining species used as indicators. This turns Isopam in an expert system. Replaces the automated selection of indicators with sieve = TRUE (ind overrules sieve).
centers	optional vector with indices (numeric) or names (character) of observations used as cluster cores. With centers alone (fully supervised mode), exactly as many clusters as centers are created. Combine with c.fix or c.max for semi-supervised modes (see Details).
distance	name of a dissimilarity index for the distance matrix used as a starting point for Isomap. Any distance measure implemented in packages **vegan** (predefined or using a designdist equation) or **proxy** can be used (see details).

k.max	maximum Isomap *k*.
d.max	maximum number of Isomap dimensions.
juice	logical. If TRUE input files for Juice are generated.
polishing	treatment of rare or invariant species and plots with few species. In the case of polishing = "strict" (default), species with only one occurrence or no variance and plots with only one species are omitted during clustering. If "relaxed" is used, only missing and invariant species and empty plots are removed.
...	other arguments used by juice or passed to S3 functions plot and identify (see dendrogram and hclust).
x	isopam result object in methods plot, print, and identify.
object	isopam result object in method summary.

Details

Isopam is described in Schmidtlein et al. (2010). It consists of dimensionality reduction (Isomap: Tenenbaum et al. 2000; isomap in **vegan**) and partitioning of the resulting ordination space (PAM: Kaufman & Rousseeuw 1990; pam in **cluster**). The classification is performed either as a hierarchical, divisive method, or as non-hierarchical partitioning. It has the following features: partitions are optimized for the occurrence of species with high fidelity to groups; it optionally selects the number of clusters per division; the shapes of groups in feature space are not restricted to spherical or other regular geometric shapes (thanks to the underlying Isomap algorithm); the distance measure used for the initial distance matrix can be freely defined.

Three supervised modes are available when centers are provided:

- Fully supervised mode (default with centers): Only the specified centers are used, creating exactly as many clusters as centers provided. No optimization of cluster number.
- Semi-supervised mode with fixed number of clusters (centers + c.fix): When c.fix exceeds the number of centers, additional clusters are formed to reach the total specified by c.fix. For example, with 2 centers and c.fix = 4, you get 2 fixed + 2 free clusters. The initial medoids for swapping are placed using a greedy maximin approach.
- Semi-supervised mode with optimization of cluster number (centers + c.max): When c.max is explicitly set greater than the number of centers (and c.fix = NULL), the algorithm searches for the optimal total number of clusters up to c.max. The provided centers remain fixed while the number and placement of additional clusters is optimized. If no additional clusters improve the result, only the provided centers are used.

Pre-defined indicator species are not as constraining as centers, even if preference is given to cluster solutions in which their fidelity is maximized. It depends on the data how much they affect the result.

Using polishing = "strict" reduces noise introduced by rare species and random outcomes due to species-poor plots, which consequently are not allocated. If you have the feeling that species with only one occurrence and plots with only one species should also contribute to the clustering, work with polishing = "relaxed", where only empty plots and missing species are excluded. This comes at the risk of noise and unstable results caused by coincidental species occurrences.

The preset distance measure is Bray-Curtis (Odum 1950). Distance measures are passed to vegdist or to designdist in **vegan**. If this does not work it is passed to dist in **proxy**. Measures available

in vegan are listed in vegdist. Isopam does not accept distance matrices as a replacement for the original data matrix because it operates on individual descriptors (species).

Isopam is slow with large data sets. It switches to a slow mode when an internally used lookup array does not fit into RAM. It is used for the results of the search for an optimal parameterisation (selection of Isomap dimensions and -*k*, optionally selection of cluster numbers) does not fit into RAM.

`plot` creates (and silently returns) an object of class `dendrogram` and calls the S3 plot method for that class. `identify` works just like `identify.hclust`.

Value

`call`	generating call
`distance`	distance measure used by Isomap
`flat`	observations (plots) with group affiliation. Running group numbers for each level of the hierarchy.
`hier`	observations (plots) with group affiliation. Group identifiers reflect the cluster hierarchy. Not present with only one level of partitioning.
`medoids`	observations (plots) representing the medoids of the resulting groups.
`analytics`	table summarizing parameter settings for the partitioning steps. `Name`: name of the respective parent cluster (0 in case of the first partition); `Subgroups`: number of subgroups; `Isomap.dim`: Isomap dimensions used; `Isomap.k.min`: minimum possible Isomap *k*; `Isomap.k`: Isomap *k* used; `Isomap.k.max`: maximum possible Isomap *k*; `Ind.N`: number of indicators reaching or exceeding `Gs`; `Ind.Gs`: the average standardized G value of these indicators; and `Global.Gs`: the average standardized G value of all descriptors (species).
`centers_usr`	Cluster centers suggested by user.
`ind_usr`	Indicators suggested by user.
`supervised`	character string describing the type of supervision used, or `NULL` if unsupervised.
`indicators`	Indicators used.
`dendro`	an object of class `hclust` representing the clustering (as used by `plot`). Not present with only one level of partitioning.
`dat`	data used

Note

With very small datasets, the indicator based optimization may fail. In such cases consider using `sieve = FALSE` instead of the default method.

Author(s)

Sebastian Schmidtlein with contributions from Jason Collison, Robin Pfannendoerfer and Lubomir Tichý

References

Odum, E.P. (1950): Bird populations in the Highlands (North Carolina) plateau in relation to plant succession and avian invasion. *Ecology* **31**: 587–605.

Kaufman, L., Rousseeuw, P.J. (1990): *Finding groups in data*. Wiley.

Schmidtlein, S., Tichý, L., Feilhauer, H., Faude, U. (2010): A brute force approach to vegetation classification. *Journal of Vegetation Science* **21**: 1162–1171.

Tenenbaum, J.B., de Silva, V., Langford, J.C. (2000): A global geometric framework for nonlinear dimensionality reduction. *Science* **290**, 2319–2323.

See Also

`clusters` for extracting cluster assignments from the result object.

`isotab` for a table of descriptor (species) frequencies in clusters and fidelity measures. There is a plot method associated to `isotab` objects that visualizes species fidelities to clusters.

Examples

```
## load data to the current environment
data(andechs)

## call isopam with the standard options
ip <- isopam(andechs)

## print function
ip

## examine cluster hierarchy
plot(ip)

## retrieve cluster vector (second hierarchy level)
clusters(ip, 2)

## frequency table
it <- isotab(ip)
it

## plot with species fidelities (equalized phi)
plot(it)

## non-hierarchical partitioning with (forced) three clusters
ip <- isopam(andechs, c.fix = 3)
ip

## limiting the set of species used in cluster search
ip <- isopam(andechs, ind = c("Car_pan", "Sch_fer"))
ip

## supervised mode with fixed cluster medoids
ip <- isopam(andechs, centers = c("p3", "p19"))
ip
```

```
## semi-supervised: one fixed medoid + two free clusters
ip <- isopam(andechs, centers = "p3", c.fix = 3)
ip

## semi-supervised search: one fixed medoid and optimized number of additional clusters
ip <- isopam(andechs, centers = "p3", c.max = 4)
ip
```

`isotab` *Fidelity and frequency of species in clusters*

Description

Calculates the fidelity of species to clusters. Returns equalized phi coefficients of association, an ordered frequency table and Fisher‘s exact test for the probability of obtaining the observed frequencies. Isopam objects as well as other combinations of tables and cluster vectors are accepted as input data. An associated plotting method visualises how closely individual species are associated with clusters.

Usage

```
isotab(x, level = NULL, clusters = NULL, phi.min = "isotab", p.max = .05)
## S3 method for class 'isotab'
print(x, n = NA, ...)
```

Arguments

`x`	Object either of class `isopam` or a dataframe or matrix with rownames (plot names) and column names (species names) that is accompanied by a cluster vector (`clusters`) with named elements corresponding to the rows in `x`. Tibbles need a column with plot names (<chr>), while the other columns are of class <dbl> or <int>. In method `print`, `x` is an object of class `isotab`.
`clusters`	Vector with assignments of clusters to plots, only needed if `x` is not an isopam object. The names of the elements need to be identical to the rownames of `x`.
`level`	Level in cluster hierarchy starting with 1 = first division.
`phi.min`	Threshold of equalized *phi* determining which species are listed in the upper part of the table. Applies only to species passing the criterion defined by `p.max`. If `phi.min = "isopam"` (the default) isotab suggests a value based on the numbers of observations.
`p.max`	Threshold of Fisher's *p* determining which species are listed in the upper part of the table. Applies only to species passing the criterion defined by `phi.min`.
`n`	number of lines used by `print`. If `NA` (the default), n is oriented on the number of diagnosic species. Use `n = Inf` to print all rows.
`...`	other arguments used by `print`.

Details

phi.min is based on the 'equalized *phi*' value according to Tichý & Chitrý 2006. The threshold proposed if phi.min is set to "isotab" should be adjusted to local conditions. The significance (Fisher's *p*) refers to the probability that the observed frequency is reached. The test is two-tailed, which means that exceptionally low frequencies can result as highly significant as well as exceptionally high frequencies. This allows positive and negative characterisation of a cluster by species.

Value

call	generating call
depth	Number of levels in the cluster hierarchy from the original clustering procedure.
level	Level chosen for isotab.
tab	Ordered species by cluster table with frequencies and their significance. The latter is derived from Fisher's exact test (see fisher_p and details, $p <= 0.05$: *, $p <= 0.01$: **, $p <= 0.001$: ***).
phi	Dataframe with equalized phi values (see details).
fisher_p	Numerical results from Fisher's exact test (see details)
n	Matrix with cluster sizes.
thresholds	phi.min and p.max used for table sorting.
typical	Text with items (often species) typically found in clusters (according to thresholds).
typical_vector	typical as a single character vector.
sorted_table	Ordered species by plot table.

Author(s)

Sebastian Schmidtlein

References

Tichý, L., Chytrý, M. (2006): Statistical determination of diagnostic species for site groups of unequal size. *Journal of Vegetation Science* **17**: 809-–818.

Schmidtlein, S., Tichý, L., Feilhauer, H., Faude, U. (2010): A brute force approach to vegetation classification. *Journal of Vegetation Science* **21**: 1162–1171.

See Also

isopam, plot.isotab

Examples

```
## load data to the current environment
data(andechs)

## call isopam with the standard options
ip <-isopam(andechs)
```

```
## build table
it <- isotab(ip)
it

## change phi threshold
it <- isotab(ip, phi.min = 0.8)

## switch cluster level
it <- isotab(ip, level = 1)
it
```

`plot.isotab` *Plot species fidelities to clusters*

Description

Function to plot `isotab` results. Based on equalised phi values according to Tichý & Chitrý (2006), the method visualises how closely how many species are associated with clusters.

Usage

```
## S3 method for class 'isotab'
plot(x, labels = FALSE, text.size = 15, title = NULL,
       phi.min = "isotab", p.max = "isotab", ...)
```

Arguments

`x`	Object of class `isotab`.
`labels`	Logical. Whether the bars should be labeled with species names. You may need to enlarge the figure height to accommodate these names (or decrease `text.size`).
`text.size`	Text size
`title`	Optional text string with title
`phi.min`	Threshold of equalized *phi* determining which species are shown. Applies only to species passing the criterion defined by `p.max`. If `phi.min = "isotab"` (the default) the threshold used by isotab is applied. Use `phi.min = 0` to remove the filter.
`p.max`	Threshold of Fisher's *p* determining which species are shown. Applies only to species passing the criterion defined by `phi.min`. Note that this value relates to frequencies rather than phi. If `p.max = "isotab"` (the default) the threshold used by isotab is applied. Use `p.max = 1` to remove the filter.
`...`	Other arguments (ignored)

Details

The thresholds are explained in `isotab`.

Value

Prints and returns (invisibly) an object of class `ggplot`.

Author(s)

Sebastian Schmidtlein

References

Tichý, L., Chytrý, M. (2006): Statistical determination of diagnostic species for site groups of unequal size. *Journal of Vegetation Science* **17**: 809-–818.

See Also

`isopam`, `isotab`

Examples

```
## load data to the current environment
data(andechs)

## call isopam with the default options
ip <- isopam(andechs)

## calculate fidelities
it <- isotab(ip)

## plotting
plot(it)

## show species labels
plot(it, labels = TRUE)

## show all species
plot(it, phi.min = 0)
```

Index

∗ **cluster**
    isopam, 3
∗ **datasets**
    andechs, 2

andechs, 2

clusters, 2, *7*

dendrogram, *5*
designdist, *5*
dist, *5*

hclust, *5*

identify.isopam *(*isopam*)*, 3
isomap, *5*
isopam, *3*, 3, *9*, *11*
isotab, *3*, *7*, 8, *10*, *11*

pam, *5*
plot.isopam *(*isopam*)*, 3
plot.isotab, *9*, 10
print.isopam *(*isopam*)*, 3
print.isotab *(*isotab*)*, 8

summary.isopam *(*isopam*)*, 3

vegdist, *5*, *6*