Package 'archetypes'

October 12, 2022

Type Package

Title Archetypal Analysis

Version 2.2-0.1

Date 2014-04-08

Depends methods, stats, modeltools, nnls (>= 1.1)

Suggests MASS, vcd, mlbench, ggplot2, TSP

Description The main function archetypes implements a
framework for archetypal analysis supporting arbitrary
problem solving mechanisms for the different conceptual
parts of the algorithm.

License GPL (>= 2)

Collate 'archetypes-barplot.R' 'generics.R' 'archetypes-class.R'
'archetypes-kit-blocks.R' 'archetypes-kit.R' 'archetypes-map.R'
'archetypes-movie.R' 'archetypes-panorama.R' 'pcplot.R'
'archetypes-pcplot.R' 'archetypes-robust.R'
'archetypes-screeplot.R' 'archetypes-step.R'
'archetypes-weighted.R' 'archetypes-xyplot.R' 'memento.R'
'simplex-pot.R' 'skeletonplot.R'

Author Manuel J. A. Eugster [aut, cre],
Friedrich Leisch [aut],
Sohan Seth [ctb]

Maintainer Manuel J. A. Eugster <manuel@mjae.net>

NeedsCompilation no

Repository CRAN

Date/Publication 2019-04-22 06:55:33 UTC

R topics documented:

---

archetypes   *Perform archetypal analysis on a data matrix.*

Description

Perform archetypal analysis on a data matrix.

Usage

```
archetypes(data, k, weights = NULL, maxIterations = 100,
  minImprovement = sqrt(.Machine$double.eps), maxKappa = 1000,
  verbose = FALSE, saveHistory = TRUE,
  family = archetypesFamily("original"), ...)
```

Arguments

data	A numeric $n \times m$ data matrix.
k	The number of archetypes.
weights	Data weights matrix or vector (used as elements of the diagonal weights matrix).
maxIterations	The maximum number of iterations.
minImprovement	The minimal value of improvement between two iterations.
maxKappa	The limit of kappa to report an ill-ness warning.
verbose	Print some details during execution.
saveHistory	Save each execution step in an environment for further analyses.
family	Blocks defining the underlying problem solving mechanisms; see archetypesFamily.
...	Additional arguments for family blocks.

Value

An object of class archetypes, see as.archetypes.

References

Cutler and Breiman. Archetypal Analysis. Technometrics, 36(4), 1994. 338-348.

See Also

Other archetypes: archetypesFamily; as.archetypes; robustArchetypes; weightedArchetypes

Examples

```
data(toy)
  a <- archetypes(toy, 3)
```

---

archetypesFamily   *Archetypes family constructor*

Description

This function returns a problem solving block for each of the different conceptual parts of the algorithm.

Usage

```
archetypesFamily(which = c("original", "weighted", "robust"), ...)
```

Arguments

which   The kind of archetypes family.

...   Exchange predefined family blocks with self-defined functions.

Value

A list containing a function for each of the different parts.

See Also

Other archetypes: archetypes; as.archetypes; robustArchetypes; weightedArchetypes

---

archmap   *Archetypal maps*

Description

Two-dimensional projection of the observations based on the alpha coefficients into a space spanned by the (projected) archetypes.

Usage

```
archmap(object, projection = simplex_projection, projection_args = list(),
  rotate = 0, cex = 1.5, col = 1, pch = 1, xlab = "", ylab = "",
  axes = FALSE, asp = TRUE, ...)
```

Arguments

`object`	An `archetypes` object
`projection`	Projection function; see `archmap_projections`
`projection_args`	
	Arguments passed to the projection function; see `archmap_projections`
`rotate`	Rotation angle to rotate the projection
`cex`	Character expansion of archetypes
`col`	Color of observations
`pch`	Point character of observations
`xlab`	A label for the x-axis
`ylab`	A label for the y-axis
`axes`	Logical value to draw axes or not
`asp`	The y/x aspect ratio
`...`	Arguments passed to the underlying plot function

Value

Invisible matrix with the projected archetypes

See Also

Other archmap: `archmap_projections`, `atypes_projection`, `simplex_projection`, `tspsimplex_projection`

Examples

```
## Not run:
  data("skel", package = "archetypes")
  skel2 <- subset(skel, select = -Gender)

  set.seed(1981)
  a <- archetypes(skel2, k = 5)

  ## Simplex projection:
  archmap(a, col = skel$Gender)

  ## Simplex projection with archetypes arranged according to their
  ## distances:
  archmap(a, col = skel$Gender,
          projection = tspsimplex_projection)
  archmap(a, col = skel$Gender,
          projection = tspsimplex_projection,
          projection_args = list(equidist = TRUE))

  ## MDS projection:
  archmap(a, col = skel$Gender,
          projection = atypes_projection)

## End(Not run)
```

as.archetypes	*Archetypes object constructor*

Description

Archetypes object constructor

Usage

```
as.archetypes(object, k, alphas, rss, iters = NULL, call = NULL,
  history = NULL, kappas = NULL, betas = NULL, zas = NULL,
  family = NULL, familyArgs = NULL, residuals = NULL, weights = NULL,
  reweights = NULL, scaling = NULL)
```

Arguments

object	The archetypes; a $p \times m$ matrix, see [parameters]().
k	The number of archetypes;
alphas	The coefficients; a $n \times p$ matrix, see [coef]().
rss	The residual sum of squares; see [rss.archetypes]().
iters	The number of iterations to the convergence.
call	The call of the [archetypes]() function.
history	If saveHistory set then an environment with the archetypes object for each execution step;
kappas	The kappas for each system of linear equations.
betas	The data coefficients; a $p \times n$ matrix.
zas	The temporary archetypes.
family	The archetypes family.
familyArgs	Additional arguments for family blocks.
residuals	The residuals.
weights	The data weights.
reweights	The data reweights.
scaling	The scaling parameters of the data.

Value

A list with an element for each parameter and class attribute archetypes.

See Also

Other archetypes: [archetypesFamily](); [archetypes](); [robustArchetypes](); [weightedArchetypes]()

barplot.archetypes   *Barplot of archetypes.*

Description

Barplot of archetypes.

Usage

```
## S3 method for class 'archetypes'
barplot(height, data, which = c("below", "beside"),
  which.beside = c("atypes", "variables"), which.below = c("compressed",
  "default"), percentiles = FALSE, below.compressed.height = 0.1,
  below.compressed.srt = 0, col.atypes = NULL, ...)
```

Arguments

height	An [archetypes](#) object.
data	The original data matrix.
which	below creates a barplot for each archetype, beside creates one barplot with bars side by side.
which.beside	Barplot according to atypes or variables.
which.below	compressed plots the labels only once.
percentiles	Show real values or percentile profiles.
below.compressed.height	
	Height of additional tail subplot.
below.compressed.srt	
	Rotations of the x-labels.
col.atypes	Color of archetypes; only used in below.compressed.
...	Passed to the underlying [barplot](#) call.

Value

Undefined.

---

bestModel.stepArchetypes

Return best model

Description

Return best model

Usage

```
## S3 method for class 'stepArchetypes'
bestModel(object, ...)

## S3 method for class 'repArchetypes'
bestModel(object, ...)
```

Arguments

object	An archetypes object.
...	Ignored

---

body   *Exploring relationships in body dimensions*

Description

Body girth measurements and skeletal diameter measurements, as well as age, weight, height and gender, are given for 507 physically active individuals - 247 men and 260 women.

Usage

```
body
```

Format

A data.frame containing 507 observations of 25 variables.

References

Heinz, Peterson, Johnson and Kerk. "Exploring relationships in body dimensions". Journal of Statistics Education, 11(2). http://www.amstat.org/publications/jse/v11n2/datasets.heinz.html

See Also

skel

`coef.archetypes` *Return coefficients*

Description

Return coefficients

Usage

```
## S3 method for class 'archetypes'
coef(object, type = c("alphas", "betas"), ...)
```

Arguments

`object`	An archetypes object.
`type`	Return alpha or beta coefficients.
`...`	Ignored.

Value

Coefficient matrix.

`fitted.archetypes` *Return fitted data*

Description

Returns the approximated data.

Usage

```
## S3 method for class 'archetypes'
fitted(object, ...)
```

Arguments

`object`	An archetypes object.
`...`	Ignored.

Value

Matrix with approximated data.

---

kappa.archetypes   *Return kappa*

Description

Return kappa

Usage

```
## S3 method for class 'archetypes'
kappa(z, ...)
```

Arguments

z	An archetypes object.
...	Ignored.

Value

A vector of kappas.

---

lines.pcplot   *Add lines to an existing parallel coordinates plot.*

Description

Add lines to an existing parallel coordinates plot.

Usage

```
## S3 method for class 'pcplot'
lines(x, data, col = 1, lty = 1, ...)
```

Arguments

x	A matrix or data frame containing the additional data.
data	The data of the existing parallel coordinates plot.
col	Line colors.
lty	Line types.
...	Passed to underlying matlines.

Value

Undefined.

---

movieplot *Archetypes movies.*

Description

Archetypes movies.

Shows the intermediate steps of the algorithm;

Archetypes parallel coordinates plot movie.

Usage

```
movieplot(zs, data, show = c("atypes", "adata", "rwdata"), ssleep = 0,
  bsleep = 0, postfn = function(iter) { }, rwdata.col1 = gray(0.7),
  rwdata.col2 = 2, ...)

movieplot2(zs, data, show = "atypes", ssleep = 0, bsleep = 0,
  zas.col = 2, zas.pch = 13, old.col = rgb(1, 0.5, 0.5), ...)

moviepcplot(zs, data, show = c("atypes", "adata"), ssleep = 0, bsleep = 0,
  ...)
```

Arguments

zs	An [archetypes](#) object.
data	The data matrix.
show	Show archetypes or approximated data.
ssleep	Seconds to sleep before start.
bsleep	Seconds to sleep between each plot.
postfn	Post plot function; is called in each iteration after the plot call.
rwdata.col1	If show = 'rwdata': color of base data set.
rwdata.col2	If show = 'rwdata': color of weighted data set.
...	Passed to underlying plot functions.
zas.col	Color of the intermediate archetypes.
zas.pch	Type of the intermediate archetypes points.
old.col	Color of the archetypes on step further.

Value

Undefined.

Undefined.

Undefined.

nparameters.archetypes
Return number of archetypes

Description

Return number of archetypes

Usage

```
## S3 method for class 'archetypes'
nparameters(object, ...)

## S3 method for class 'stepArchetypes'
nparameters(object, ...)

## S3 method for class 'repArchetypes'
nparameters(object, ...)
```

Arguments

object	An archetypes object.
...	Ignored.

Value

Number of archetypes.

panorama.archetypes *Panorma plot for archetypes.*

Description

Panorma plot for archetypes.

Usage

```
## S3 method for class 'archetypes'
panorama(object, data, distfn = distEuclidean,
  xlab = "Index", ylab = "Distance", order = TRUE, col = 1, pch = 1,
  cex = 1, atypes.col = (seq(length = nparameters(object)) + 1),
  atypes.pch = rep(19, nparameters(object)), atypes.cex = rep(1,
  nparameters(object)), ylim = NULL, ...)
```

Arguments

`object`	An [archetypes](#)-related object.
`data`	A matrix or data frame.
`distfn`	Distance function.
`xlab`	Label of xaxis.
`ylab`	Label of yaxis.
`order`	Order the distances.
`col`	Color of distances.
`pch`	Plot character of distances.
`cex`	magnification of the distances.
`atypes.col`	Color of archetype distances.
`atypes.pch`	Plot character of archetype distances.
`atypes.cex`	Magnification of the archetype distances.
`ylim`	The y limits of the plot.
`...`	Passed to the underlying `plot` call.

Examples

```
## Not run:
  data(toy)
  a <- archetypes(toy, 3)
  panorama(a, toy)

  ## See demo(robust-ozone).

## End(Not run)
```

```
parameters,archetypes-method
```
Return fitted archetypes

Description

Return fitted archetypes

Usage

```
## S4 method for signature 'archetypes'
parameters(object, ...)

## S4 method for signature 'stepArchetypes'
parameters(object, ...)

## S4 method for signature 'repArchetypes'
parameters(object, ...)
```

Arguments

object	An archetypes object.
...	Ignored.

Value

Matrix with k archetypes.

---

pcplot.archetypes   *Parallel coordinates of data and archetypes.*

Description

Parallel coordinates of data and archetypes.

Usage

```
## S3 method for class 'archetypes'
pcplot(x, data, data.col = gray(0.7), data.lwd = 1,
  atypes.col = 2, atypes.lwd = 2, atypes.lty = 1, chull = NULL,
  chull.col = 1, chull.lwd = 2, chull.lty = 1, ...)
```

Arguments

x	An [archetypes](#) object.
data	A matrix or data frame.
data.col	Color of data lines.
data.lwd	Width of data lines.
atypes.col	Color of archetypes lines.
atypes.lwd	Width of archetypes lines.
atypes.lty	Type of archetypes lines.
chull	An integer vector giving the indices of the points from data lying on the convex hull.
chull.col	Color of convex hull lines.
chull.lwd	Width of convex hull lines.
chull.lty	Type of convex hull lines.
...	Passed to [pcplot](#) and [lines.pcplot](#).

Value

Undefined.

---

pcplot.default   *Default parallel coordinates plot.*

Description

Code copied from function [parcoord](#) of package MASS to simply play arround with the visualization of archetypes. At a later date, when it is clear which visualisation is the best, the functionality is probabibly merged with the original function or it is possible with parallel coordinate plots which are available et all.

Usage

```
## Default S3 method:
pcplot(x, col = gray(0.7), lty = 1, var.label = TRUE,
  rx = NULL, ...)
```

Arguments

x	A $n \times m$ matrix or data frame who columns represent variables. Missing values are allowed.
col	Line color.
lty	Line type.
var.label	Axes labels.
rx	A $2 \times m$ matrix with ranges for each dimension.
...	Passed to the underlying [matplot](#) function.

Value

Undefined.

---

predict.archetypes   *Predict method for archetypal analysis fits*

Description

This method produces predicted alpha coefficients for new data.

Usage

```
## S3 method for class 'archetypes'
predict(object, newdata, ...)
```

Arguments

`object`	An archetypes object; currently only [original](#)-family objects.
`newdata`	A data frame with data for which to predict the alpha coefficients.
`...`	Ignored.

Value

The predict alpha coefficients.

`residuals.archetypes`   *Return residuals*

Description

Return residuals

Usage

```
## S3 method for class 'archetypes'
residuals(object, ...)
```

Arguments

`object`	An archetypes object.
`...`	Ignored.

Value

Matrix with residuals.

`robustArchetypes`   *Robust archetypes*

Description

Robust archetypes

Usage

```
robustArchetypes(data, k, familyBlocks = list(), ...)
```

Arguments

familyBlocks	Exchange predefined family blocks; see `archetypesFamily`.
data	A numeric $n \times m$ data matrix.
k	The number of archetypes.
...	Additional arguments for family blocks.

Value

An object of class robustArchetypes and `as.archetypes`.

See Also

Other archetypes: `archetypesFamily`; `archetypes`; `as.archetypes`; `weightedArchetypes`

---

rss   *Defined generics*

Description

Generics defined by the archetypes package.

Return number of parameters

Return best model

Panorama

Parallel coordinates plot

Usage

```
rss(object, ...)

nparameters(object, ...)

bestModel(object, ...)

panorama(object, ...)

pcplot(x, ...)
```

Arguments

object	An object
...	Futher arguments
x	An object.

---

rss.archetypes   *Return residual sum of squares*

Description

Return residual sum of squares

Usage

```
## S3 method for class 'archetypes'
rss(object, type = c("scaled", "single", "global"), ...)

## S3 method for class 'stepArchetypes'
rss(object, ...)

## S3 method for class 'repArchetypes'
rss(object, ...)
```

Arguments

object	An archetypes object.
type	Return scaled, single or global RSS.
...	Ignored.

Value

Residual sum of squares.

---

screeplot.stepArchetypes

Screeplot of stepArchetypes.

Description

Screeplot draws the residual sum of square curve based on the best model of each step.

Usage

```
## S3 method for class 'stepArchetypes'
screeplot(x, type = c("lines", "barplot"), ...)
```

Arguments

x	A [stepArchetypes](#) object.
type	Draw lines or a barplot.
...	Passed to underlying plot functions.

Value

Undefined.

---

simplexplot   *Simplex visualization*

Description

The stochastic nature of the alpha coefficients implies that they exist on a standard (K-1)-simplex with the K archetypes Z as the corners, and the coefficients as the coordinate with respect to these corners. A standard simplex can be projected to two dimensions via a skew orthogonal projection, where all the vertices of the simplex are shown on a circle connected by edges. The individual alpha coefficients can be then projected into this circle.

Usage

```
simplexplot(object, radius = 10, order = NULL, labels_cex = 1,
  labels = NULL, show_labels = TRUE, points_col = "#00000044",
  points_pch = 19, points_cex = 1, projection = simplex_projection,
  show_points = TRUE, show_circle = TRUE, circle_col = "lightgray",
  show_edges = TRUE, edges_col = "lightgray", show_direction = FALSE,
  direction_length = 1, directions_col = points_col, ...)
```

Arguments

object	An [archetypes](#) object
radius	Radius of the projection
order	Order of the archetypes
labels_cex	Label expansion
labels	Labels
show_labels	Show labels
points_col	Color of the points
points_pch	Plot character of the points
points_cex	Character expansion of the points
projection	Projection function; see [archmap_projections](#)
show_points	Show the points
show_circle	Show the circle
circle_col	Color of the circle
show_edges	Show the edges
edges_col	Color of the edges
direction_length	
	Expansion of the direction pointers
directions_col	Color of the direction pointers
show_direction	Show direction pointers
...	Additional arguments; currently ignored

Value

Invisible list of all computed components needed for the simplex visualization.

References

See Section 6 in "Probabilistic Archetypal Analysis" by Seth and Eugster (2014), http://arxiv.org/abs/1312.7604.

Examples

```
### This example reproduces parts of the Figure 7 shown in
  ### "Probabilistic Archetypal Analysis" by Seth and Eugster (2014)

  data("toy", package = "archetypes")

  suppressWarnings(RNGversion("3.5.0"))
  set.seed(1234); a3 <- archetypes(toy, k = 3)
  set.seed(1237); a4 <- archetypes(toy, k = 4)
  set.seed(1238); a5 <- archetypes(toy, k = 5)

  simplexplot(a3)
  simplexplot(a3, show_direction = TRUE, show_points = FALSE)
  simplexplot(a4, projection = tspsimplex_projection)
  simplexplot(a5, show_direction = TRUE, show_points = FALSE,
    direction_length = 2, directions_col = "black")
```

---

simplex_projection   *Archetypal map projections*

Description

Archetypal map projections

Usage

```
simplex_projection(x, r = 10)

tspsimplex_projection(x, r = 10, equidist = FALSE, ...)

atypes_projection(x)
```

Arguments

x	Archetypes matrix
r	Radius of the simplex projection
equidist	Arrange archetypes equidistantly or in relation to their distance
...	Parameters for the solve_TSP function

Value

Matrix with the projected archetypes

See Also

Other archmap: archmap

---

skel   *Exploring relationships in body dimensions, skeletal measurements*

Description

Skeletal diameter measurements, as well as height and gender, are given for 507 physically active individuals - 247 men and 260 women.

This is a subset of the body data set.

Usage

```
skel
```

Format

A data.frame containing 507 observations of 11 variables.

References

Heinz, Peterson, Johnson and Kerk. "Exploring relationships in body dimensions". Journal of Statistics Education, 11(2). http://www.amstat.org/publications/jse/v11n2/datasets.heinz.html

See Also

body

---

skeletonplot   *Skeleton plot.*

Description

Displays a schematic representation of skeleton data as available in dataset skel.

Displays a generic skeleton with annotations explaining the measurements available in data set skel.

Usage

```
skeletonplot(x, skel.width = 100, skel.height = 200, ylab = "Height (cm)",
  base.radius = 2, xlab = "", xlim = (nrow(x) * c(0, skel.width)),
  ylim = c(0, skel.height), col = NULL, mtext = TRUE, skel.lwd = 1, ...)

jd()
```

Arguments

x	Matrix or data.frame of skeleton data.
skel.width	Reference width for instance calculation.
skel.height	Reference height for instance calculation.
base.radius	Base radius for points.
xlab	The x label of the plot.
ylab	The y label of the plot.
xlim	Numeric of length 2 giving the x limits for the plot.
ylim	Numeric of length 2 giving the y limits for the plot.
col	Color of the different parts of the skeleton.
mtext	Label archetypes.
skel.lwd	Line width of skeleton.
...	Passed to underlying canvas plot function.

Value

List of skeleton instances.

Generic skeleton instance.

See Also

[skel](#)

---

stepArchetypes	*Run archetypes algorithm repeatedly*

Description

Run archetypes algorithm repeatedly

Usage

```
stepArchetypes(..., k, nrep = 3, method = archetypes, verbose = TRUE)
```

Arguments

...	Passed to the specific archetype function.
k	A vector of integers passed in turn to the k argument of [archetypes]().
nrep	For each value of k run [archetypes]() nrep times.
method	Archetypes function to use, typically [archetypes](), [weightedArchetypes]() or [robustArchetypes](),
verbose	Show progress during exection.

Value

A list with k elements and class attribute `stepArchetypes`. Each element is a list of class `repArchetypes` with `nrep` elements; only for internal usage.

See Also

[archetypes]()

Examples

```
## Not run:
  data(skel)
  skel2 <- subset(skel, select=-Gender)
  as <- stepArchetypes(skel2, k=1:5, verbose=FALSE)

  ## Residual sum of squares curve:
  screeplot(as)

  ## Select three archetypes and from that the best
  ## recurrence:
  a3 <- bestModel(as[[3]])

## End(Not run)
```

---

summary.stepArchetypes

Summary method for stepArchetypes object

Description

Summary method for stepArchetypes object

Usage

```
## S3 method for class 'stepArchetypes'
summary(object, ...)
```

Arguments

object	A `stepArchetypes` object.
...	Ignored.

Value

Undefined.

---

toy	*Toy data set*

Description

A simple artificial two-dimensional data set.

Usage

```
toy
```

Format

A data.frame containing 250 observations of 2 variables.

---

weightedArchetypes	*Weighted archetypes*

Description

Weighted archetypes

Usage

```
weightedArchetypes(data, k, weights = NULL, familyBlocks = list(), ...)
```

Arguments

weights	Data weights matrix.
familyBlocks	Exchange predefined family blocks; see [archetypesFamily](#).
data	A numeric $n \times m$ data matrix.
k	The number of archetypes.
...	Additional arguments for family blocks.

Value

An object of class `weightedArchetypes` and [as.archetypes](#).

See Also

Other archetypes: archetypesFamily; archetypes; as.archetypes; robustArchetypes

---

weights.archetypes | *Return weights*

Description

Return weights

Usage

```
## S3 method for class 'archetypes'
weights(object, type = c("weights", "reweights"), ...)
```

Arguments

object   An archetypes object.

type   Return global weights (weighted archetypes) or weights calculated during the iterations (robust archetypes).

...   Ignored.

Value

Vector of weights.

---

xyplot | *Two-dimensional plot.*

Description

Two-dimensional plot.

Usage

```
xyplot(x, ...)
```

Arguments

x   An object.

...   Further arguments.

Value

Undefined.

xyplot.archetypes   *Plot of two-dimensional data and archetypes.*

Description

Plot of two-dimensional data and archetypes.

Usage

```
## S3 method for class 'archetypes'
xyplot(x, y, data.col = 1, data.pch = 19,
  data.bg = NULL, atypes.col = 2, atypes.pch = 19, ahull.show = TRUE,
  ahull.col = atypes.col, chull = NULL, chull.col = gray(0.7),
  chull.pch = 19, adata.show = FALSE, adata.col = 3, adata.pch = 13,
  link.col = data.col, link.lty = 1, ...)
```

Arguments

x	An [archetypes](#) object.
y	A matrix or data frame.
data.col	Color of data points.
data.pch	Type of data points.
data.bg	Background of data points.
atypes.col	Color of archetypes points.
atypes.pch	Type of archetypes points.
ahull.show	Show approximated convex hull.
ahull.col	Color of approximated convex hull line.
chull	An integer vector giving the indices of the points from data lying on the convex hull.
chull.col	Color of convex hull points.
chull.pch	Type of convex hull points.
adata.show	Show approximated data with link to the original data.
adata.col	Color of approximated data points.
adata.pch	Type of approximated data points.
link.col	Color of link between approximated and original data points.
link.lty	Line type of link between approximated and original data points.
...	Passed to the underlying plot functions.

Value

Undefined.

Note

The link between approximated and original data is based on an idea and Matlab source code of Bernard Pailthorpe.

---

xyplot.robustArchetypes

Plot of two-dimensional data and robust archetypes.

Description

Plot of two-dimensional data and robust archetypes.

Usage

```
## S3 method for class 'robustArchetypes'
xyplot(x, y, ...)
```

Arguments

x	An archetypes object.
y	A matrix or data frame.
...	Arguments of xyplot.weightedArchetypes and xyplot.robustArchetypes

---

xyplot.stepArchetypes *Plot of two-dimensional data and stepArchetypes.*

Description

Plot of two-dimensional data and stepArchetypes.

Usage

```
## S3 method for class 'stepArchetypes'
xyplot(x, y, data.col = gray(0.7), data.pch = 19,
  atypes.col = (seq_len(length(x) * length(x[[1]])) + 1), atypes.pch = 19,
  ahull.show = TRUE, ahull.col = atypes.col, ...)
```

Arguments

x	An [stepArchetypes](#) object.
y	A matrix or data frame.
data.col	Color of data points.
data.pch	Type of data points.
atypes.col	Color of archetypes points.
atypes.pch	Type of archetypes points.
ahull.show	Show approximated convex hull.
ahull.col	Color of approximated convex hull line.
...	Passed to the underlying plot functions.

Value

Undefined.

---

xyplot.weightedArchetypes

Plot of two-dimensional data and weighted archetypes.

Description

Plot of two-dimensional data and weighted archetypes.

Usage

```
## S3 method for class 'weightedArchetypes'
xyplot(x, y, data.col = 1, data.pch = 21,
  data.bg = gray, link.col = NULL, link.lty = NULL,
  weights.type = "weights", ...)
```

Arguments

x	An [archetypes](#) object.
y	A matrix or data frame.
data.col	Color of data points.
data.pch	Type of data points.
data.bg	Background of data points.
link.col	Color of link between approximated and original data points.
link.lty	Line type of link between approximated and original data points.
weights.type	Weights to display; see [weights.archetypes](#).
...	Arguments of [xyplot.archetypes](#).

[.stepArchetypes   *Extract method*

Description

An extraction on a stepArchetypes object returns again a stepArchetypes object.

Usage

```
## S3 method for class 'stepArchetypes'
x[i]
```

Arguments

x   A stepArchetypes object.

i   The indizes to extract.

Value

A stepArchetypes object containing only the parts defined in i.

Index