

# Package ‘archetypes’

February 14, 2012

**Type** Package

**Title** Archetypal Analysis

**Version** 2.0-2

**Date** 2010-08-24

**Depends** methods, stats, modeltools, nnl (>= 1.1)

**Suggests** MASS, vcd, mlbench, ggplot2

**Author** Manuel J. A. Eugster <manuel.eugster@stat.uni-muenchen.de>

**Maintainer** Manuel J. A. Eugster <manuel.eugster@stat.uni-muenchen.de>

**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)

**Revision** 44

**Collate** ‘archetypes-barplot.R’ ‘archetypes-class.R’ ‘archetypes-deprecated.R’ ‘archetypes-kit-blocks.R’ ‘archetypes-kit.R’ ‘archetypes-movie.R’ ‘archetypes-panorama.R’ ‘pcplot.R’ ‘archetypes-pcplot.R’ ‘archetypes-robust.R’ ‘archetypes-screepplot.R’ ‘archetypes-step.R’ ‘archetypes-weighted.R’ ‘archetypes-xyplot.R’ ‘memento.R’ ‘skeletonplot.R’

**Repository** CRAN

**Date/Publication** 2010-08-24 10:47:57

## R topics documented:

archetypes . . . . .	2
archetypes-class . . . . .	3
archetypes-deprecated . . . . .	5
archetypes-generics . . . . .	6

archetypesFamily	7
barplot.archetypes	7
body	8
memento	9
movieplot	9
panorama.archetypes	10
pcplot	11
screeplot.stepArchetypes	12
skel	13
skeletonplot	14
stepArchetypes	15
toy	16
xyplot	17

## Index 19

---

archetypes	<i>Perform archetypal analysis on a data matrix.</i>
------------	--

---

### Description

Perform archetypal analysis on a data matrix.

### Usage

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

### 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 <a href="#">archetypesFamily</a> .
...	Additional arguments for family blocks.
familyBlocks	Exchange predefined family blocks.

**Value**

archetypes: An object of class `archetypes`, see [archetypes-class](#).

robustArchetypes: An object of class `robustArchetypes` and [archetypes-class](#).

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

**Note**

Please see the vignette for a detailed explanation!

**References**

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

**See Also**

[stepArchetypes](#), [archetypes-class](#)

**Examples**

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

---

<code>archetypes-class</code>	<i>Archetypes object constructor and methods.</i>
-------------------------------	---

---

**Description**

Archetypes object constructor and methods.

**Usage**

```
as.archetypes(archetypes, k, alphas, rss, iters, call, history, kappas, betas, zas,
  family, familyArgs, residuals, weights, reweights)
## S3 method for class 'archetypes'
fitted(object, ...)
## S3 method for class 'archetypes'
coef(object, type=c("alphas", "betas"), ...)
## S3 method for class 'archetypes'
residuals(object, ...)
## S3 method for class 'archetypes'
rss(object, type=c("scaled", "single", "global"), ...)
## S3 method for class 'archetypes'
weights(object, type=c("weights", "reweights"), ...)
## S3 method for class 'archetypes'
kappa(z, ...)
## S3 method for class 'archetypes'
nparameters(object, ...)
```

**Arguments**

archetypes	The archetypes; a $p \times m$ matrix, see <a href="#">atypes</a> .
k	The number of archetypes;
alphas	The coefficients; a $n \times p$ matrix, see <a href="#">alphas</a> .
rss	The residual sum of squares; see <a href="#">rss</a> .
iters	The number of iterations to the convergence.
call	The call of the <a href="#">archetypes</a> 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.
object	An archetypes-related object.
...	Ignored.
type	Return alphas or betas.
z	An archetypes object.

**Value**

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

fitted.archetypes: Matrix with approximated data.

parameters,archetypes-method: Matrix with  $k$  archetypes.

coef.archetypes: Coefficient matrix.

residuals.archetypes: Matrix with residuals.

rss.archetypes: Residual sum of squares.

weights.archetypes: Vector of weights.

kappa.archetypes: A vector of kappas.

nparameters.archetypes: Number of archetypes.

**See Also**

[archetypes](#)

---

archetypes-deprecated *Deprecated functions in archetypes package*

---

## Description

These functions are provided for compatibility with older versions of archetypes only, and may be defunct as soon as the next release.

## Usage

```
atypes(zs, ...)  
ntypes(zs, ...)  
adata(zs, ...)  
alphas(zs, ...)  
betas(zs, ...)  
iters(zs, ...)  
ahistory(zs, ...)  
nhistory(zs, ...)
```

## Arguments

<code>zs</code>	An archetypes-related object.
<code>...</code>	Further arguments.

## Details

`atypes`: replaced by [parameters](#).  
`ntypes`: replaced by [nparameters](#).  
`adata`: replaced by [fitted](#).  
`alphas`: replaced by [coef](#).  
`betas`: replaced by [coef](#).  
`iters`: removed.  
`ahistory`: removed; see [memento](#).  
`nhistory`: removed; see [memento](#).

## Value

`atypes`: Archetypes matrix.  
`ntypes`: Number of archetypes.  
`adata`: Approximated data matrix.  
`alphas`: Alpha matrix.  
`betas`: Beta matrix.  
`iters`: Number of iterations.

ahistory: The archetypes object of the requested step.

nhistory: The number of history steps available.

---

archetypes-generics    *Generic functions in archetypes package*

---

### Description

These generic functions are defined in the package archetypes.

### Usage

```
rss(object, ...)  
nparameters(object, ...)  
panorama(object, ...)  
bestModel(object, ...)  
xyplot(x, ...)  
pcplot(x, ...)
```

### Arguments

object	An object.
...	Ignored.
x	An object.

### Value

rss: Residual sum of squares.

nparameters: Number of parameters.

panorama: Undefined.

bestModel: The best models.

xyplot: Undefined.

---

archetypesFamily      *Archetypes family constructor.*

---

**Description**

Archetypes family constructor.

**Usage**

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

**Arguments**

which                    The kind of archetypes family.  
 ...                      Exchange predefined family blocks with self-defined functions.

**Details**

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

**Value**

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

**See Also**

[archetypes](#)

---

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, ...)
```

**Arguments**

height	An <code>archetypes</code> object.
data	The original data matrix.
which	<code>below</code> creates a barplot for each archetype, <code>beside</code> 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 <code>below.compressed</code> .
...	Passed to the underlying <code>barplot</code> call.

**Value**

Undefined.

---

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

---

memento	<i>Memento environment.</i>
---------	-----------------------------

---

**Description**

Memento environment.

**Arguments**

i	The number of the state.
state	The state to save.

**Details**

Simple implementation of the 'Memento' design pattern.

**Value**

Memento environment.

**Examples**

```
## Not run:  
m <- new.memento()  
m$save(i, state)  
m$states()  
m$get(i)  
  
## End(Not run)
```

---

movieplot	<i>Archetypes movies.</i>
-----------	---------------------------

---

**Description**

Archetypes movies.

**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 <code>archetypes</code> 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 <code>show = 'rwdata'</code> : color of base data set.
rwdata.col2	If <code>show = 'rwdata'</code> : 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.

**Details**

movieplot2: Shows the intermediate steps of the algorithm;

**Value**

movieplot: Undefined.  
 movieplot2: Undefined.  
 moviepcplot: Undefined.

---

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, ...)
```

**Arguments**

object	An <code>archetypes</code> -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)
```

---

pcplot

*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, chull.col=1, chull.lwd=2, chull.lty=1, ...)
## Default S3 method:
pcplot(x, col=gray(0.7), lty=1, var.label=TRUE, rx, ...)
lines.pcplot(x, data, col=1, lty=1, ...)
```

**Arguments**

<code>x</code>	An <code>archetypes</code> object.
<code>data</code>	A matrix or data frame.
<code>data.col</code>	Color of data lines.
<code>data.lwd</code>	Width of data lines.
<code>atypes.col</code>	Color of archetypes lines.
<code>atypes.lwd</code>	Width of archetypes lines.
<code>atypes.lty</code>	Type of archetypes lines.
<code>chull</code>	An integer vector giving the indices of the points from <code>data</code> lying on the convex hull.
<code>chull.col</code>	Color of convex hull lines.
<code>chull.lwd</code>	Width of convex hull lines.
<code>chull.lty</code>	Type of convex hull lines.
<code>...</code>	Passed to <code>pcplot</code> and <code>lines.pcplot</code> .
<code>col</code>	Line color.
<code>lty</code>	Line type.
<code>var.label</code>	Axes labels.
<code>rx</code>	A $2 \times m$ matrix with ranges for each dimension.

**Details**

`pcplot.default`: Code copied from function `parcoord` of package `MASS` to simply play around with the visualization of archetypes. At a later date, when it is clear which visualisation is the best, the functionality is probably merged with the original function or it is possible with parallel coordinate plots which are available et all.

**Value**

`pcplot.archetypes`: Undefined.

`pcplot.default`: Undefined.

`lines.pcplot`: Undefined.

---

`screepLOT.stepArchetypes`

*ScreepLOT of stepArchetypes.*

---

**Description**

ScreepLOT of stepArchetypes.

**Usage**

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

**Arguments**

x	A <a href="#">stepArchetypes</a> object.
type	Draw lines or a barplot.
...	Passed to underlying plot functions.

**Details**

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

**Value**

Undefined.

---

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

Skeleton plot.

### 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=c(hipbase = 1, hip = 1, shoulderbase = 1, shoulder = 1, head =
                 1, elbow = 2, wrist = 3, knee = 4, ankle = 5, chest = "purple1",
                 pelvis = 6), mtext=TRUE, skel.lwd=1, ...)
jd()
```

### Arguments

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

### Details

`skeletonplot`: Displays a schematic representation of skeleton data as available in dataset [skel](#).

`jd`: Displays a generic skeleton with annotations explaining the measurements available in data set [skel](#).

### Value

`skeletonplot`: List of skeleton instances.

`jd`: Generic skeleton instance.

### See Also

[skel](#)

---

stepArchetypes	<i>Runs archetypes algorithm repeatedly.</i>
----------------	--

---

### Description

Runs archetypes algorithm repeatedly.

### Usage

```
stepArchetypes(..., k, nrep=3, method=archetypes, verbose=TRUE)
## S3 method for class 'stepArchetypes'
x[i]
## S3 method for class 'stepArchetypes'
summary(object, ...)
## S3 method for class 'stepArchetypes'
nparameters(object, ...)
## S3 method for class 'stepArchetypes'
rss(object, ...)
## S3 method for class 'stepArchetypes'
bestModel(object, ...)
```

### Arguments

...	Passed to <a href="#">archetypes</a> function.
k	A vector of integers passed in turn to the k argument of <a href="#">archetypes</a> .
nrep	For each value of k run <a href="#">archetypes</a> nrep times.
method	Archetypes function to use, typically <a href="#">archetypes</a> , <a href="#">weightedArchetypes</a> or <a href="#">robustArchetypes</a> ,
verbose	Show progress during execution.
x	A stepArchetypes object.
i	The indices to extract.
object	A stepArchetypes object.

### Details

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

### Value

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

[.stepArchetypes: A stepArchetypes object containing only the parts defined in i.

summary.stepArchetypes: Undefined.

`nparameters.stepArchetypes`: Vector of numbers of archetypes.  
`rss.stepArchetypes`: A vector of residual sum of squares.  
`bestModel.stepArchetypes`: A list of length `k` of best models.

**Note**

Please see the vignette for a detailed explanation!

**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:
screepplot(as)

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

## End(Not run)
```

---

toy

*Toy data set*

---

**Description**

A simple artificial two-dimensional data set.

**Usage**

```
toy
```

**Format**

A `data.frame` containing 250 observations of 2 variables.

xyplot

*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, atypes.col=2, atypes.pch=19,
       ahull.show=TRUE, ahull.col=atypes.col, chull, chull.col=gray(0.7),
       chull.pch=19, adata.show=FALSE, adata.col=3, adata.pch=13,
       link.col=data.col, link.lty=1, ...)
## S3 method for class 'weightedArchetypes'
xyplot(x, y, data.col=1, data.pch=21, data.bg=gray, link.col, link.lty,
       weights.type="weights", ...)
## S3 method for class 'robustArchetypes'
xyplot(x, y, ...)
## 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 <a href="#">archetypes</a> 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.

<code>link.col</code>	Color of link between approximated and original data points.
<code>link.lty</code>	Line type of link between approximated and original data points.
<code>...</code>	Passed to the underlying plot functions.
<code>weights.type</code>	Weights to display; see <a href="#">weights.archetypes</a> .

**Value**

`xyplot.archetypes`: Undefined.

`xyplot.stepArchetypes`: Undefined.

**Note**

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

# Index

## \*Topic **datasets**

- body, 8
- skel, 13
- toy, 16
- [.stepArchetypes (stepArchetypes), 15
- adata (archetypes-deprecated), 5
- ahistory (archetypes-deprecated), 5
- alphas, 4
- alphas (archetypes-deprecated), 5
- archetypes, 2, 4, 7, 8, 10–12, 15–17
- archetypes-class, 3
- archetypes-class, 3
- archetypes-deprecated, 5
- archetypes-generics, 6
- archetypesFamily, 2, 7
- as.archetypes (archetypes-class), 3
- atypes, 4
- atypes (archetypes-deprecated), 5
- barplot, 8
- barplot.archetypes, 7
- bestModel (archetypes-generics), 6
- bestModel.stepArchetypes (stepArchetypes), 15
- betas (archetypes-deprecated), 5
- body, 8, 13
- coef, 5
- coef.archetypes (archetypes-class), 3
- fitted, 5
- fitted.archetypes (archetypes-class), 3
- iters (archetypes-deprecated), 5
- jd (skeletonplot), 14
- kappa.archetypes (archetypes-class), 3
- lines.pcplot, 12

- lines.pcplot (pcplot), 11
- memento, 5, 9
- moviepcplot (movieplot), 9
- movieplot, 9
- movieplot2 (movieplot), 9
- new.memento (memento), 9
- nhistory (archetypes-deprecated), 5
- nparameters, 5
- nparameters (archetypes-generics), 6
- nparameters.archetypes (archetypes-class), 3
- nparameters.stepArchetypes (stepArchetypes), 15
- ntypes (archetypes-deprecated), 5
- panorama (archetypes-generics), 6
- panorama.archetypes, 10
- parameters, 5
- parameters (archetypes-class), 3
- parameters.archetypes-method (archetypes-class), 3
- parcoord, 12
- pcplot, 11, 12
- pcplot (archetypes-generics), 6
- pcplot.archetypes (pcplot), 11
- pcplot.default (pcplot), 11
- residuals.archetypes (archetypes-class), 3
- robustArchetypes, 15
- robustArchetypes (archetypes), 2
- rss, 4
- rss (archetypes-generics), 6
- rss.archetypes (archetypes-class), 3
- rss.stepArchetypes (stepArchetypes), 15
- screepplot.stepArchetypes, 12
- skel, 13, 14
- skeletonplot, 14

`stepArchetypes`, 3, 13, 15  
`summary.stepArchetypes`  
  (`stepArchetypes`), 15

`toy`, 16

`weightedArchetypes`, 15  
`weightedArchetypes (archetypes)`, 2  
`weights.archetypes`, 18  
`weights.archetypes (archetypes-class)`, 3

`xyplot`, 17  
`xyplot (archetypes-generics)`, 6