Package: portion (via r-universe)

October 28, 2026

Type Package

Title Extracting a Data Portion

Version 0.1.3

Description Provides simple methods to extract data portions from various objects. The relative portion size and the way the portion is selected can be chosen.

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Imports stats

URL https://github.com/loelschlaeger/portion

BugReports https://github.com/loelschlaeger/portion/issues

Config/roxygen2/version 8.0.0

Repository https://loelschlaeger.r-universe.dev

Date/Publication 2026-06-30 15:38:15 UTC

RemoteUrl https://github.com/loelschlaeger/portion

RemoteRef HEAD

RemoteSha a7b0ba46d973c10b4a8b51b38f83b82dc180d79d

Contents

portion 2

Index **5**

portion *Extract a data portion*

Description

Extract a relative portion from vectors, matrices, data frames, or lists of these objects.

Usage

```
portion(x, proportion, how = "random", centers = 2L, ...)

## Default S3 method:
portion(x, ...)

## S3 method for class 'numeric'
portion(x, proportion, how = "random", centers = 2L, ...)

## S3 method for class 'character'
portion(x, proportion, how = "random", ...)

## S3 method for class 'logical'
portion(x, proportion, how = "random", centers = 2L, ...)

## S3 method for class 'matrix'
portion(
  x,
  proportion,
  how = "random",
  centers = 2L,
  byrow = TRUE,
  ignore = integer(),
  ...
)

## S3 method for class 'data.frame'
portion(
  x,
  proportion,
  how = "random",
  centers = 2L,
  byrow = TRUE,
  ignore = integer(),
  ...
)

## S3 method for class 'list'
portion(x, proportion, how = "random", centers = 2L, ...)
```

Arguments

x	An object to be portioned.
proportion	[numeric(1)] The relative portion size as a number between 0 and 1. The absolute size is rounded up with ceiling().
how	[character(1)] Exactly one of: • "random" (default), select at random • "first", select the first elements, rows, or columns • "last", select the last elements, rows, or columns • "similar", select similar elements, rows, or columns • "dissimilar", select dissimilar elements, rows, or columns Options "similar" and "dissimilar" are based on clustering via kmeans and require numeric clustering data without NA, NaN, or infinite values.
centers	[integer(1)] Only relevant if how = "similar" or how = "dissimilar". A positive whole number passed to kmeans for clustering. If centers exceeds the number of distinct values or rows, it is reduced automatically.
...	Further arguments to be passed to or from other methods.
byrow	[logical(1)] Only relevant if x has two dimensions (rows and columns). In this case, set to TRUE to portion row-wise (default) or FALSE to portion column-wise.
ignore	[integer()] Only relevant for two-dimensional x and how = "similar" or how = "dissimilar". Indices to exclude from the clustering data, but not from the returned object. With byrow = TRUE, these are column indices. With byrow = FALSE, these are row indices. This is useful, for example, to ignore identifier or non-numeric columns while selecting similar rows.

Value

A portion of x, preserving attributes where possible. Vectors return selected elements, matrices and data frames return selected rows or columns, and lists return a list with each element portioned. The selected indices are stored in the "indices" attribute of each returned object.

Examples

```
# can portion vectors, matrices, data.frames, and lists of such types
portion(
  list(
    1:10,
    matrix(LETTERS[1:12], nrow = 3, ncol = 4),
    data.frame(a = 1:6, b = -6:-1)
  ),
```

```
  proportion = 0.5,
  how = "first"
)

# can portion similar and dissimilar elements (based on kmeans clustering)
set.seed(1)
x <- c(1, 1, 2, 2)
portion(x, proportion = 0.5, how = "similar")
portion(x, proportion = 0.5, how = "dissimilar")

# ignore non-numeric columns when clustering data frame rows
x <- data.frame(value = c(1, 1, 5, 5), group = c("a", "a", "b", "b"))
portion(x, proportion = 0.5, how = "similar", ignore = 2)

# object attributes are preserved
x <- structure(1:10, "test_attribute" = "test")
x[1:5]
portion(x, proportion = 0.5, how = "first")
```

Index

kmeans, *3*

portion, 2