

Package ‘aperçu’

October 12, 2022

Type Package

Date 2019-12-05

Title Quick Look at your Data

Version 0.2.4

Author Aurelien Chateigner <aurelien.chateigner@gmail.com>

Maintainer Aurelien Chateigner <aurelien.chateigner@gmail.com>

Description The goal is to print an “aperçu”, a short view of a vector, a matrix, a data.frame, a list or an array. By default, it prints the first 5 elements of each dimension. By default, the number of columns is equal to the number of lines. If you want to control the selection of the elements, you can pass a list, with each element being a vector giving the selection for each dimension.

License CC BY-SA 4.0

Encoding UTF-8

LazyData true

RoxygenNote 7.0.2

Imports pls

Suggests testthat

NeedsCompilation no

Repository CRAN

Date/Publication 2019-12-06 08:10:02 UTC

R topics documented:

ap	2
Index	5

ap *Aperçu of your object*

Description

Some says that size doesn't matter, but when it comes to data it does. The goal is to print an "aperçu", a short view of an object.

Usage

```
ap(object, limitsList = limitsLister(object), pA = FALSE)
```

Arguments

object	A vector, matrix, data frame, list or array
limitsList	A list with each element being the selection of each dimension
pA	Boolean, to only print the aperçu (FALSE) or to also print the dimensions and classes (TRUE)

Details

The goal is to print an "aperçu", a short view of a vector, a matrix, a data.frame, a list or an array. By default, it prints the first 5 elements of each dimension. By default, the number of columns is equal to the number of lines. If you want to control the selection of the elements, you can pass a list, with each element being a vector giving the selection for each dimension. This function provides a simpler way of using the '[' function, with the same speed and flexibility.

Value

A quick look or a selection of the data

Examples

```
# Creation of a vector, a matrix, a data frame, a list and 3 arrays of 3, 4
# and 5 dimensions:
v <- c(1:20)
names(v) <- letters[1:20]
m <- matrix(1:100, 10, 10)
dimnames(m) <- list(letters[1:10], letters[1:10])
df <- as.data.frame(m)
li <- lapply(1:10, function(x) {
  u <- matrix((1:100)*x, 10,10)
  dimnames(u) <- list(letters[1:10], letters[1:10])
  return(u)
})
names(li) <- letters[1:10]
a <- array(c(1:1000), c(10,10,10))
dimnames(a) <- list(letters[1:10], letters[1:10], letters[1:10])
```

```

a2 <- array(1:10000, c(10,10,10,10))
dimnames(a2) <- list(letters[1:10], letters[1:10], letters[1:10],
  letters[1:10])
a3 <- array(1:100000, c(10,10,10,10,10))
dimnames(a3) <- list(letters[1:10], letters[1:10], letters[1:10],
  letters[1:10], letters[1:10])

# Automatic apply of the objects
ap(v)
ap(m)
ap(df)
ap(li)
ap(a)
ap(a2)
ap(a3)

# To print also the dimensions and classes
ap(v, pA = TRUE)
ap(m, pA = TRUE)
ap(df, pA = TRUE)
ap(li, pA = TRUE)
ap(a, pA = TRUE)
ap(a2, pA = TRUE)
ap(a3, pA = TRUE)
# if the size of the object is very small :
sm <- matrix(1:4, 2, 2)
sDf <- as.data.frame(sm)
ap(sm)
ap(sDf)

# Specific apply of the objects
ap(v, list(1:2))
ap(m, list(c(1,3), 1:5))
## outputs a vector as only one dimension is given
ap(m, list(c(1,3,1:5)))
## outputs lines 1, 3 and 5, and columns 1 to 10
ap(df, list(c(1,3,5), 1:10))
## the result is different between a list and an array
## as the dimensions of both are not in the same order
ap(li, list(c(1:3,5),c(4,6,9), c(3,6)))
ap(a, list(c(1:3,5),c(4,6,9), c(3,6)))
## outputs the element a[1,3,5]
ap(a, list(1,3,5))
## outputs the element a[c(1,3,5),1,1]
ap(a, list(c(1,3,5)))
ap(a2, list(1:4,3:5,2:8,3:4))
ap(a3, list(1:4,3:5,2:3,3:4,7:10))

# It also works with data frames with a matrix in it:
library(pls)
data("gasoline")
ap(gasoline)

```

```
ap(gasoline, list(1:10, list(1:10,1:10)))
```

Index

ap, [2](#)