# Chương 2: Kiểu dữ liệu trong R

1. sum

Mục đích: Trả về tổng của tất cả giá trị có trong các đối số của nó.

Công thức: sum(…, na.rm = TRUE)

Ý nghĩa các tham số:

…: vector số, số phức, logic

na.rm: loại bỏ các giá trị NA

Ví dụ: sum(1:100, na.rm = TRUE)

1. median

Mục đích: Tính toán trung vị mẫu

Công thức: median(x, na.rm = TRUE, …)

Ý nghĩa các tham số:

x : một đối tượng mà một phương thức đã được xác định hoặc một vectơ số chứa các giá trị có giá trị trung bình sẽ được tính toán.

na.rm: một giá trị lôgic cho biết liệu các giá trị có nên được loại bỏ trước khi quá trình tính toán được tiến hành hay không.

… : những lập luận có thể tiếp tục cho các phương pháp; không được sử dụng trong phương thức mặc định.

Ví dụ : median(c(1:3,100,1000))

1. max

Mục đích: Trả về giá trị lớn nhất trong một vectơ

Công thức: max(x, na.rm = FALSE)

Ý nghĩa các tham số

x: vectơ hoặc một khung dữ liệu

na.rm: loại bỏ các giá trị NA, nếu nó đề cập đến False nó sẽ coi là NA hoặc nếu nó đề cập đến True, nó sẽ loại bỏ NA khỏi vector hoặc khung dữ liệu.

Ví dụ:

vector = c(1,2,3,4,5)

max(vector)

1. min

Mục đích: Trả về giá trị nhỏ nhất trong một vectơ

Công thức: min(x, na.rm = FALSE)

Ý nghĩa các tham số:

x: vectơ hoặc một khung dữ liệu

na.rm: loại bỏ các giá trị NA, nếu nó đề cập đến False nó sẽ coi là NA hoặc nếu nó đề cập đến True, nó sẽ loại bỏ NA khỏi vector hoặc khung dữ liệu.

Ví dụ:

vector = c(1,2,3,4,5)

min(vector)

1. c()

Mục đích: Đây là một hàm chung kết hợp các đối số của nó.

Phương thức mặc định kết hợp các đối số của nó để tạo thành một vectơ. Tất cả các đối số được ép buộc thành một kiểu chung là kiểu của giá trị trả về và tất cả các thuộc tính ngoại trừ tên đều bị xóa.

Công thức chung: c(…)

Công thức mặc định: c(..., recursive = FALSE, use.names = TRUE)

Ý nghĩa các tham số:

… : các đối tượng được nối

Recursive(đệ quy): hàm giảm xuống một cách đệ quy thông qua các danh sách (và danh sách ghép nối) kết hợp tất cả các phần tử của chúng thành một vectơ.

use.names: lôgic chỉ ra nếu names nên được giữ nguyên

Ví dụ:

c(1:5,10.5, “next”)

1. cbind, rbind

Mục đích: kết hợp R đối tượng theo hàng hoặc cột. Lấy một chuỗi các đối số vectơ, ma trận hoặc khung dữ liệu và kết hợp theo cột hoặc hàng, tương ứng. Đây là các hàm chung với các phương thức cho các lớp R khác.

Công thức:

cbind(..., deparse.level = 1)

rbind(..., deparse.level = 1)

Phương thức S3 cho data.frame

rbind(..., deparse.level =1, make.row.names = TRUE,

StringAsFactors = default.stringAsFactors (), factor.exclude = NA)

...: (tổng quát) vectơ hoặc ma trận. Chúng có thể được đưa ra dưới dạng đối số được đặt tên. Các đối tượng R khác có thể bị cưỡng chế khi thích hợp

deparse.level: số nguyên điều khiển việc xây dựng nhãn trong trường hợp đối số không giống ma trận (đối với phương thức mặc định): deparse.level = 0 cấu trúc không có nhãn; mặc định, deparse.level = 1 hoặc 2 tạo nhãn từ tên đối số, hãy xem phần 'Giá trị' bên dưới.

make.row.names: (chỉ dành cho phương thức khung dữ liệu :) lôgic cho biết nếu duy nhất và hợp lệ row.namesnên được xây dựng từ các đối số.

stringAsFactors: logic, được chuyển đến as.data.frame; chỉ có tác dụng khi các …đối số chứa a (non- data.frame) character.

factor.exclude: nếu các khung dữ liệu chứa các yếu tố, hãy TRUE đảm bảo rằng NA các mức độ của các yếu tố được giữ lại.

Ví dụ:

data\_2 = data.frame(z1 = c(1, 5, 9, 4, 0),

z2 = c(0, 9, 8, 1, 6))

data\_new2 = cbind(data\_1, data\_2)

data\_new2

data\_new3 = rbind(data\_1, data\_2)

data\_new3

1. vector

Mục đích: tạo ra một vectơ có độ dài và chế độ đã cho.

as.vector, một chung chung, cố gắng ép đối số của nó vào một vectơ của chế độ mode(mặc định là buộc chuyển sang chế độ vectơ nào là thuận tiện nhất): nếu kết quả là nguyên tử thì tất cả các thuộc tính sẽ bị xóa.

is.vector trả về TRUE nếu x là một vectơ của chế độ được chỉ định không có thuộc tính nào khác ngoài tên . Nó trả về FALSE ngược lại.

vector(mode = “logical”, length = 0)

as.vector(x, mode = “any”)

is.vector(x,mode = “any”)

mode: chuỗi ký tự đặt tên một chế độ nguyên tử hoặc "list"hoặc "expression"hoặc (ngoại trừ vector) "any"

length: một số nguyên không âm xác định độ dài mong muốn. Đối với một vectơ dài, tức là length .Machine$integer.max, nó phải thuộc loại "double". Việc cung cấp đối số có độ dài khác với đối số là một lỗi.

x: một đối tượng R.

Ví dụ:

x = c(a = 1, b = 2)

is.vector(x)

as.vector(x)

all.equal(x, as.vector(x))

1. matrix

Mục đích: tạo một ma trận từ tập giá trị đã cho.

as.matrix: cố gắng biến đối số của nó thành một ma trận.

is.matrix: kiểm tra xem đối số của nó có phải là một ma trận (nghiêm ngặt) hay không.

Công thức: matrix(data = NA, nrow = 1, ncol = 1, byrow = FALSE,

dimnames = NULL)

as.matrix(x, …)

as.matrix(x, rownames.force = NA, …)

is.matrix(x)

Ý nghĩa các tham số:

data :một vectơ dữ liệu tùy chọn (bao gồm một danh sách hoặc expressionvectơ). Các đối tượng R không được phân loại nguyên tử bị cưỡng chế as.vectorvà tất cả các thuộc tính bị loại bỏ.

nrow: số lượng hàng mong muốn.

ncol: số lượng cột mong muốn.

byrow: Logic. Nếu FALSE(mặc định) thì ma trận được điền bởi các cột, nếu không thì ma trận được lấp đầy bởi các hàng.

Dimnames: Một dimnames thuộc tính cho các ma trận: NULLhoặc một listchiều dài 2 đã nêu tên hàng và cột tương ứng. Một danh sách trống được coi là NULL, và một danh sách có độ dài là tên hàng. Danh sách có thể được đặt tên và tên danh sách sẽ được sử dụng làm tên cho các thứ nguyên.

x: một đối tượng R.

…: các đối số bổ sung được chuyển đến hoặc từ các phương thứ

rownames.force: lôgic chỉ ra nếu ma trận kết quả phải có ký tự (thay vì NULL) rownames. Mặc định, NAsử dụng NULLtên hàng nếu khung dữ liệu có tên hàng 'tự động' hoặc đối với khung dữ liệu không có hàng.

Ví dụ:

A = matrix( c(2, 4, 3, 1, 5, 7),

nrow=2,

ncol=3,

byrow = TRUE)

A

1. data.frame

Mục đích: Tạo ra các khung dữ liệu, các tập hợp các biến được kết hợp chặt chẽ với nhau chia sẻ nhiều thuộc tính của ma trận và danh sách, được sử dụng làm cấu trúc dữ liệu cơ bản bởi hầu hết các phần mềm mô hình của R.

Công thức:

data.frame (…, row.names = NULL, check.rows = FALSE,

check.names = TRUE, fix.empty.names = TRUE,

stringAsFactors = default.stringsAsFactors ())

default.stringsAsFactors ()

Ý nghĩa các tham số:

…:các đối số này có dạng valuehoặc tag = value. Tên thành phần được tạo dựa trên thẻ (nếu có) hoặc chính đối số đã rút gọn.

row.names: NULL hoặc một số nguyên hoặc một chuỗi ký tự chỉ định một cột được sử dụng làm tên hàng hoặc một ký tự hoặc vectơ số nguyên cung cấp tên hàng cho khung dữ liệu.

check.rows: nếu TRUEsau đó các hàng được kiểm tra về tính nhất quán của độ dài và tên.

check.names: Logic. Nếu TRUEsau đó tên của các biến trong khung dữ liệu được kiểm tra để đảm bảo rằng chúng là tên biến hợp lệ về mặt cú pháp và không bị trùng lặp. Nếu cần, chúng được điều chỉnh (bằng cách make.names) sao cho đúng như vậy.

fix.empty.names: lôgic cho biết nếu các đối số "không được đặt tên" (theo nghĩa không được gọi chính thức là someName = arg) nhận được một tên được xây dựng tự động hoặc đúng hơn là tên "". Cần được đặt thành FALSE ngay cả khi check.names sai nếu ""tên nên được giữ nguyên.

stringAsFactors: lôgic: vectơ ký tự có nên được chuyển đổi thành thừa số không? Mặc định 'mới ban đầu' là TRUE, nhưng điều này có thể được thay đổi bằng cách cài đặt options(stringsAsFactors = FALSE).

Ví dụ:

emp.data = data.frame(emp\_id = c (1:5),

emp\_name = c("Rick","Dan","Michelle","Ryan","Gary"),

salary = c(623.3,515.2,611.0,729.0,843.25),

start\_date = as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11", "2015-03-27")),

stringsAsFactors = FALSE)

print(emp.data)

1. list

Mục đích: Các chức năng xây dựng, cưỡng chế và kiểm tra cả hai loại danh sách R.

list(…)

pairlist(…)

as.list(x, …)

as.list(x, all.names = FALSE, sorted = FALSE, …)

as.pairlist(x)

is.list(x)

is.pairlist(x)

alist(…)

Ý nghĩa các tham số

…: các đối tượng, có thể được đặt tên.

X: đối tượng bị cưỡng chế hoặc kiểm tra.

all.names: lôgic cho biết có nên sao chép tất cả các giá trị hay (mặc định) chỉ những giá trị có tên không bắt đầu bằng dấu chấm.

Sorted: một lôgic cho biết liệu namesdanh sách kết quả có được sắp xếp hay không (ngày càng nhiều). Lưu ý rằng điều này hơi tốn kém, nhưng có thể hữu ích để so sánh giữa các môi trường.

Ví dụ:

list\_data <- list(c("Jan","Feb","Mar"), matrix(c(3,9,5,1,-2,8), nrow = 2),

list("green",12.3))

names(list\_data) <- c("1st Quarter", "A\_Matrix", "A Inner list")

print(list\_data)

1. rep

Mục đích: sao chép các giá trị trong x. Nó là một hàm chung và phương thức mặc định (nội bộ) được mô tả ở đây.

Rep.int và rep\_len là các phiên bản đơn giản hóa nhanh hơn cho hai trường hợp phổ biến. Bên trong, chúng là chung, vì vậy các phương pháp có thể được xác định cho chúng.

rep(x, …)

rep.int(x, times)

rep\_len(x, length.out)

Ý nghĩa các tham số

X: một vectơ (của bất kỳ chế độ nào bao gồm a list) hoặc một yếu tố hoặc ( repchỉ cho ) a POSIXcthoặc POSIXlthoặc Dateđối tượng; hoặc một đối tượng S4 có chứa một đối tượng như vậy.

…: các đối số khác được chuyển đến hoặc từ các phương thức khác. Đối với phương pháp mặc định nội bộ, chúng có thể bao gồm:

Times: một vectơ có giá trị nguyên cho số lần (không âm) để lặp lại từng phần tử nếu có độ dài length(x)hoặc lặp lại toàn bộ vectơ nếu có độ dài 1. Âm hoặc NAcác giá trị là một lỗi. Một doublevectơ được chấp nhận, các đầu vào khác bị ép buộc thành một số nguyên hoặc vectơ kép.

length.out: số nguyên không âm. Chiều dài mong muốn của vectơ đầu ra. Các đầu vào khác sẽ bị ép buộc thành một vectơ kép và phần tử đầu tiên được lấy. Bỏ qua nếu NAhoặc không hợp lệ.

each:số nguyên không âm. Mỗi phần tử của xđược lặp lại nhiều eachlần. Các đầu vào khác sẽ bị ép buộc thành một số nguyên hoặc vectơ kép và phần tử đầu tiên được lấy. Đối xử như 1thể NAhoặc không hợp lệ.

Ví dụ:

x = rep(1:5,2)

x

1. seq

Mục đích: Tạo chuỗi thường xuyên. seqlà một chung tiêu chuẩn với một phương thức mặc định. seq.intlà một nguyên thủy có thể nhanh hơn nhiều nhưng có một vài hạn chế.

seq(…)

seq(from = 1, to = 1, by = ((to - from)/(length.out - 1)),

length.out = NULL, along.with = NULL, …)

seq.int(from, to, by, length.out, along.with, …)

seq\_along(along.with)

seq\_len(length.out)

Ý nghĩa các tham số:

…: các đối số được truyền đến hoặc từ các phương thức.

From,to: giá trị bắt đầu và giá trị cuối (lớn nhất) của dãy. Có độ dài 1trừ khi chỉ fromđược cung cấp dưới dạng đối số không tên.

By: number: dãy số.

length.out: độ dài mong muốn của trình tự. Một số không âm, cho seqvà seq.intsẽ được làm tròn nếu là phân số.

Along.with: lấy độ dài từ độ dài của đối số này.

Ví dụ:

seq(3, 10, length.out = 5)

seq(3, 10, along.with = 1:5)

1. dim

Mục đích: Lấy hoặc thiết lập kích thước của một đối tượng.

dim(x)

dim(x) <- value

x: một đối tượng R , ví dụ như ma trận, mảng hoặc khung dữ liệu.

Value: Đối với phương thức mặc định, một NULLhoặc một vectơ số, được ép buộc thành số nguyên (bằng cách cắt bớt).

set.seed(62626)

N <- 500

x1 <- round(rnorm(N, 0, 10))

x2 <- round(runif(N, 5, 10))

x3 <- round(runif(N, 1, 3), 1)

x4 <- round(runif(N, 10, 20))

x5 <- rpois(N, 5)

data <- data.frame(x1, x2, x3, x4, x5)

dim(data)

1. colnames, rownames

Chức năng trợ giúp nội bộ.

rownames(x, do.NULL = TRUE, prefix = "row")

rownames(x) = value

colnames(x, do.NULL = TRUE, prefix = "col")

colnames(x) = value

Ý nghĩa các tham số

x: một đối tượng R giống như ma trận , có ít nhất hai chiều cho colnames.

do.NULL: logic. Nếu FALSEvà tên là NULL, tên được tạo.

prefix: cho các tên đã tạo.

value: một giá trị hợp lệ cho thành phần đó của dimnames(x). Đối với ma trận hoặc mảng, đây là một trong hai NULLhoặc một vectơ ký tự có độ dài khác 0 bằng kích thước thích hợp.

Ví dụ:

m0 = matrix(NA, 4, 0)

rownames(m0)

m2 = cbind(1, 1:4)

colnames(m2, do.NULL = FALSE)

colnames(m2) = c("x","Y")

rownames(m2) = rownames(m2, do.NULL = FALSE, prefix = "Obs.")

m2

1. setwd

Mục đích: Trả về một đường dẫn tệp tuyệt đối đại diện cho thư mục làm việc hiện tại của tiến trình R ; setwd(dir)được sử dụng để đặt thư mục làm việc thành dir.

getwd ()

setwd (dir)

Ý nghĩa các tham số:

dir: Một chuỗi ký tự: mở rộng sẽ được thực hiện.

Ví dụ:

setwd("/my/new/path)

1. read.table: Đầu vào dữ liệu

Mục đích: Đọc tệp ở định dạng bảng và tạo khung dữ liệu từ đó, với các trường hợp tương ứng với các dòng và biến cho các trường trong tệp.

read.table(file, header = FALSE, sep = "", quote = "\"'",

dec = ".", numerals = c("allow.loss", "warn.loss", "no.loss"),

row.names, col.names, as.is = !stringsAsFactors,

na.strings = "NA", colClasses = NA, nrows = -1,

skip = 0, check.names = TRUE, fill = !blank.lines.skip,

strip.white = FALSE, blank.lines.skip = TRUE,

comment.char = "#",

allowEscapes = FALSE, flush = FALSE,

stringsAsFactors = default.stringsAsFactors(),

fileEncoding = "", encoding = "unknown", text, skipNul = FALSE)

File: tên của tệp mà dữ liệu sẽ được đọc từ đó. Mỗi hàng của bảng xuất hiện dưới dạng một dòng của tệp. Nếu nó không chứa một đường dẫn tuyệt đối , tên tệp có liên quan đến thư mục làm việc hiện tại getwd(),. Mở rộng dấu nghiêng được thực hiện nếu được hỗ trợ. Đây có thể là một tệp nén (xem file).

Ngoài ra, filecó thể là một kết nối ở chế độ văn bản có thể đọc được (sẽ được mở để đọc nếu cần, và nếu như vậy closed (và do đó bị hủy) khi kết thúc lệnh gọi hàm). (Nếu stdin()được sử dụng, lời nhắc cho các dòng có thể hơi khó hiểu. Kết thúc đầu vào bằng một dòng trống hoặc tín hiệu EOF, Ctrl-Dtrên Unix và Ctrl-Ztrên Windows. Mọi phản hồi trên stdin()sẽ bị xóa trước khi quay lại.)

filecũng có thể là một URL hoàn chỉnh. (Đối với các lược đồ URL được hỗ trợ, hãy xem phần 'URL' của trợ giúp cho url.)

Header: một giá trị logic cho biết liệu tệp có chứa tên của các biến như dòng đầu tiên của nó hay không. Nếu thiếu, giá trị được xác định từ định dạng tệp: headerđược đặt thành TRUEnếu và chỉ khi hàng đầu tiên chứa một trường ít hơn số cột.

Sep: ký tự phân cách trường. Các giá trị trên mỗi dòng của tệp được phân tách bằng ký tự này. Nếu sep = ""(mặc định cho read.table) dấu phân tách là 'khoảng trắng', đó là một hoặc nhiều dấu cách, tab, dòng mới hoặc dấu xuống dòng.

Quote: tập hợp các ký tự trích dẫn. Để tắt hoàn toàn việc trích dẫn, hãy sử dụng quote = "". Xem scancác hành vi trên dấu ngoặc kép được nhúng trong dấu ngoặc kép. Trích dẫn chỉ được coi là đối với các cột được đọc dưới dạng ký tự, đó là tất cả chúng trừ khi colClassesđược chỉ định.

Dec: ký tự được sử dụng trong tệp cho dấu thập phân.

Numerals: chuỗi chỉ ra cách chuyển đổi các số có chuyển đổi thành độ chính xác kép sẽ làm mất độ chính xác, hãy xem type.convert. Có thể viết tắt. (Cũng áp dụng cho các đầu vào số phức.)

row.names: một vectơ tên hàng. Đây có thể là một vectơ cung cấp tên hàng thực hoặc một số duy nhất cung cấp cho cột của bảng chứa tên hàng hoặc chuỗi ký tự cung cấp tên của cột bảng chứa tên hàng.

Nếu có tiêu đề và hàng đầu tiên chứa một trường ít hơn số cột, thì cột đầu tiên trong dữ liệu đầu vào được sử dụng cho tên hàng. Ngược lại, nếu row.namesbị thiếu, các hàng sẽ được đánh số.

Sử dụng row.names = NULLlực lượng đánh số hàng. Thiếu hoặc NULL row.namestạo tên hàng được coi là 'tự động' (và không được giữ nguyên bởi as.matrix).

col.names: một vectơ tên tùy chọn cho các biến. Mặc định là sử dụng "V"theo sau là số cột.

as.is: hành vi mặc định của read.tablelà chuyển đổi các biến ký tự (không được chuyển đổi thành logic, số hoặc phức) thành các yếu tố. Biến as.iskiểm soát việc chuyển đổi các cột không được chỉ định bởi colClasses. Giá trị của nó là một vectơ logic (các giá trị được tái chế nếu cần) hoặc một vectơ của các chỉ số số hoặc ký tự chỉ định cột nào không nên được chuyển đổi thành thừa số.

Lưu ý: để loại bỏ tất cả các chuyển đổi bao gồm cả chuyển đổi của các cột số, hãy đặt colClasses = "character".

Lưu ý rằng nó as.isđược chỉ định cho mỗi cột (không phải cho mỗi biến) và do đó, bao gồm cột tên hàng (nếu có) và bất kỳ cột nào bị bỏ qua.

na.strinngs: một vectơ ký tự của các chuỗi được hiểu là NAcác giá trị. Các trường trống cũng được coi là các giá trị bị thiếu trong các trường logic, số nguyên, số và trường phức. Lưu ý rằng thử nghiệm xảy ra sau khi khoảng trắng bị xóa khỏi đầu vào, vì vậy na.stringscác giá trị có thể cần xóa khoảng trắng riêng của chúng trước.

colClasses: Một vectơ của các lớp được giả định cho các cột. Nếu không có tên, hãy tái chế khi cần thiết. Nếu được đặt tên, các tên được khớp với các giá trị không xác định sẽ được sử dụng NA.

Các giá trị có thể có là NA(mặc định, khi nào type.convertđược sử dụng), "NULL"(khi cột bị bỏ qua), một trong các lớp vectơ nguyên tử (lôgic, số nguyên, số, phức, ký tự, nguyên) hoặc "factor", "Date"hoặc "POSIXct". Nếu không, cần phải có một asphương thức (từ các phương thức gói ) để chuyển đổi từ "character"sang lớp chính thức được chỉ định.

Lưu ý rằng nó colClassesđược chỉ định cho mỗi cột (không phải cho mỗi biến) và do đó bao gồm cột tên hàng (nếu có).

Nrows: số nguyên: số hàng tối đa để đọc. Các giá trị âm và không hợp lệ khác bị bỏ qua.

Skip: số nguyên: số dòng của tệp dữ liệu cần bỏ qua trước khi bắt đầu đọc dữ liệu.

check.names: Nếu TRUEsau đó tên của các biến trong khung dữ liệu được kiểm tra để đảm bảo rằng chúng là tên biến hợp lệ về mặt cú pháp. Nếu cần, chúng được điều chỉnh (bởi make.names) sao cho đúng như vậy, và cũng để đảm bảo rằng không có bản sao.

Fill: Nếu TRUEsau đó, trong trường hợp các hàng có độ dài không bằng nhau, các trường trống sẽ được thêm vào một cách ngầm định.

Strip.white: Chỉ được sử dụng khi sep đã được chỉ định và cho phép loại bỏ khoảng trắng ở đầu và cuối khỏi các charactertrường không được trích dẫn ( numericcác trường luôn bị loại bỏ). Xem scanđể biết thêm chi tiết (bao gồm cả ý nghĩa chính xác của 'khoảng trắng'), nhớ rằng các cột có thể bao gồm tên hàng.

blank.lines.skip: logic: nếu TRUE các dòng trống trong đầu vào bị bỏ qua.

comment.char: ký tự: một vectơ ký tự có độ dài một chứa một ký tự hoặc một chuỗi rỗng. Sử dụng ""để tắt hoàn toàn việc giải thích các nhận xét.

allowEscapes: Nếu kiểu C thoát ra, chẳng hạn như\ nđược xử lý hoặc đọc nguyên văn (mặc định)? Lưu ý rằng nếu không nằm trong dấu ngoặc kép, chúng có thể được hiểu là dấu phân cách (nhưng không phải là ký tự chú thích). Để biết thêm chi tiết xem scan.

Flush: lôgic: nếu TRUE, scansẽ chảy xuống cuối dòng sau khi đọc phần cuối cùng của các trường được yêu cầu. Điều này cho phép đặt các bình luận sau trường cuối cùng.

stringAsFactors: lôgic: vectơ ký tự có nên được chuyển đổi thành thừa số không? Lưu ý rằng điều này được ghi đè bởi as.isvà colClasses, cả hai đều cho phép kiểm soát tốt hơn.

fileEncoding: chuỗi ký tự: nếu không rỗng khai báo mã hóa được sử dụng trên tệp (không phải kết nối) để dữ liệu ký tự có thể được mã hóa lại. Xem phần 'Mã hóa' của trợ giúp cho file'Sổ tay Nhập / Xuất dữ liệu R' và 'Lưu ý'.

Encoding: mã hóa được giả định cho các chuỗi đầu vào. Nó được sử dụng để đánh dấu các chuỗi ký tự được biết là bằng tiếng Latinh-1 hoặc UTF-8 (xem Encoding): nó không được sử dụng để mã hóa lại đầu vào, nhưng cho phép R xử lý các chuỗi được mã hóa trong mã hóa gốc của chúng (nếu một trong những hai). Xem "Giá trị" và "Lưu ý".

Text: chuỗi ký tự: nếu filekhông được cung cấp và đây là, thì dữ liệu được đọc từ giá trị của textthông qua kết nối văn bản. Lưu ý rằng một chuỗi ký tự có thể được sử dụng để bao gồm các tập dữ liệu (nhỏ) trong mã R.

SkipNul: logic: nên bỏ qua nuls?

…: Các đối số khác sẽ được chuyển đến read.table.

Ví dụ:

x = read.table("tp.csv",header=T,sep="\t",strip.white=T);

x

1. scan

Mục đích: Đọc dữ liệu thành vectơ hoặc danh sách từ bảng điều khiển hoặc tệp.

scan(file = "")

Ý nghĩa các tham số:

File: tên của một tệp để đọc các giá trị dữ liệu từ đó. Nếu là tệp được chỉ định "", thì đầu vào sẽ được lấy từ bàn phím (hoặc bất kỳ thứ gì stdin()đọc nếu dữ liệu nhập được chuyển hướng hoặc R được nhúng). (Trong trường hợp này, đầu vào có thể được kết thúc bằng một dòng trống hoặc tín hiệu EOF,Ctrl-D trên Unix và Ctrl-Z trên Windows.)

Nếu không, tên tệp được diễn giải liên quan đến thư mục làm việc hiện tại (được cung cấp bởi getwd()), trừ khi nó chỉ định một đường dẫn tuyệt đối . Mở rộng dấu nghiêng được thực hiện nếu được hỗ trợ. Khi chạy R từ một tập lệnh, file = "stdin"có thể được sử dụng để tham chiếu đến stdinluồng tệp của quy trình .

Đây có thể là một tệp nén (xem file).

Ngoài ra, filecó thể là a connection, sẽ được mở nếu cần, và nếu được đóng khi kết thúc lệnh gọi hàm. Dù kết nối được mở ở chế độ nào, bất kỳ LF, CRLF hoặc CR nào sẽ được chấp nhận làm điểm đánh dấu EOL cho một đường và như vậy sẽ khớp sep = "\n".

filecũng có thể là một URL hoàn chỉnh. (Đối với các lược đồ URL được hỗ trợ, hãy xem phần 'URL' của trợ giúp cho url.)

Để đọc tệp dữ liệu không có trong bảng mã hiện tại (ví dụ: tệp Latin-1 trong ngôn ngữ UTF-8 hoặc ngược lại), hãy sử dụng filecài đặt kết nối encodingđối số của nó (hoặc đối số scancủa nó file)

# Chương 3: Truy cập biến, tạo subset data

1. write.table

Mục đích: in đối số bắt buộc của nó x(sau khi chuyển đổi nó thành khung dữ liệu nếu nó không phải là một hoặc không phải là ma trận) vào một tệp

write.table (x, file = "", append = FALSE, quote = TRUE, sep = "",

eol = "\ n", na = "NA", dec = ".", row.names = TRUE,

col.names = TRUE, qmethod = c ("Escape", "double"),

fileEncoding = "")

Ý nghĩa các tham số

x: đối tượng được ghi, tốt nhất là ma trận hoặc khung dữ liệu. Nếu không, nó được cố gắng ép buộc xvào một khung dữ liệu.

File: hoặc một chuỗi ký tự đặt tên cho một tệp hoặc một kết nối mở để ghi. ""chỉ ra đầu ra cho bảng điều khiển.

Append: Chỉ có liên quan nếu filelà một chuỗi ký tự. Nếu TRUE, đầu ra được nối vào tệp. Nếu FALSE, bất kỳ tệp hiện có nào của tên bị phá hủy.

Quote: một giá trị logic ( TRUEhoặc FALSE) hoặc một vectơ số. Nếu TRUE, bất kỳ cột ký tự hoặc yếu tố nào sẽ được bao quanh bởi dấu ngoặc kép. Nếu một vectơ số, các phần tử của nó được lấy làm chỉ số của các cột để trích dẫn. Trong cả hai trường hợp, tên hàng và cột được trích dẫn nếu chúng được viết. Nếu FALSE, không có gì được trích dẫn.

Sep: chuỗi phân tách trường. Các giá trị trong mỗi hàng x được phân tách bằng chuỗi

Eol: (các) ký tự cần in ở cuối mỗi dòng (hàng). Ví dụ: eol = "\r\n"sẽ tạo ra các kết thúc dòng của Windows trên hệ điều hành Unix giống nhau và eol = "\r"sẽ tạo ra các tệp như mong đợi của Excel: mac 2004.

NA: chuỗi để sử dụng cho các giá trị bị thiếu trong dữ liệu.

Dec: chuỗi để sử dụng cho dấu thập phân trong cột số hoặc cột phức tạp: phải là một ký tự duy nhất.

row.names: hoặc là một giá trị logic cho biết liệu các tên hàng xsẽ được viết cùng với xhoặc một vectơ ký tự của các tên hàng sẽ được viết.

col.names: hoặc là một giá trị lôgic cho biết liệu các tên cột xsẽ được viết cùng với xhay một vectơ ký tự của các tên cột sẽ được viết. Xem phần 'tệp CSV' để biết ý nghĩa của col.names = NA.

qmethod: một chuỗi ký tự chỉ định cách xử lý các ký tự dấu ngoặc kép được nhúng khi trích dẫn chuỗi. Phải là một trong số "escape"(mặc định cho write.table), trong trường hợp này ký tự trích dẫn được thoát trong kiểu C bởi dấu gạch chéo ngược hoặc "double"(mặc định cho write.csvvà write.csv2), trong trường hợp này, ký tự trích dẫn được nhân đôi. Bạn có thể chỉ định chữ cái đầu tiên.

fileEncoding: chuỗi ký tự: nếu không rỗng khai báo mã hóa được sử dụng trên tệp (không phải kết nối) để dữ liệu ký tự có thể được mã hóa lại khi chúng được viết. Thấy chưa file.

…: đối số write.table: append, col.names, sep, decvà qmethodkhông thể thay đổi.

Ví dụ:

write.table(mtcars, file = "mtcars.txt", sep = "\t", row.names = TRUE, col.names = NA)

1. order

Mục đích: Trả về một hoán vị sắp xếp lại đối số đầu tiên của nó thành thứ tự tăng dần hoặc giảm dần, phá vỡ quan hệ bằng các đối số khác. sort.listgiống nhau, chỉ sử dụng một đối số. Xem các ví dụ để biết cách sử dụng các hàm này để sắp xếp các khung dữ liệu, v.v.

order(…, na.last = TRUE, decreasing = FALSE,

method = c("auto", "shell", "radix"))

Ý nghĩa các tham số:

…: một chuỗi các vectơ số, phức, ký tự hoặc lôgic, tất cả đều có cùng độ dài hoặc một đối tượng R được phân loại .

x: một vectơ nguyên tử.

partial: vectơ của các chỉ số để sắp xếp từng phần. (Các NULLgiá trị không được thực hiện.)

decreasing: Thứ tự sắp xếp nên tăng hay giảm? Đối với "radix"phương thức, đây có thể là một vectơ có độ dài bằng số đối số trong ….

Ví dụ:

a = c(4, 3, 2, NA, 1)

b = c(4, NA, 2, 7, 1)

z = cbind(a, b)

o = order(a, b)); z[o, ]

o = order(a, b, na.last = FALSE)); z[o, ]

o = order(a, b, na.last = NA)); z[o, ]

1. merge

Mục đích: Hợp nhất hai khung dữ liệu theo các cột hoặc tên hàng chung, hoặc thực hiện các phiên bản khác của hoạt động kết hợp cơ sở dữ liệu .

merge(x, y, …)

Ý nghĩa các tham số:

x, y: khung dữ liệu hoặc các đối tượng được ép buộc vào một.

Ví dụ:

x = data.frame(k1 = c(NA,NA,3,4,5), k2 = c(1,NA,NA,4,5), data = 1:5)

y = data.frame(k1 = c(NA,2,NA,4,5), k2 = c(NA,NA,3,4,5), data = 1:5)

merge(x, y, by = c("k1","k2"))

1. attach

Mục đích: Cơ sở dữ liệu được đính kèm với đường dẫn tìm kiếm R. Điều này có nghĩa là cơ sở dữ liệu được tìm kiếm bởi R khi đánh giá một biến, vì vậy các đối tượng trong cơ sở dữ liệu có thể được truy cập bằng cách chỉ cần đặt tên của chúng.

attach(what, pos = 2L, name = deparse(substitute(what), backtick=FALSE),

warn.conflicts = TRUE)

what: 'cơ sở dữ liệu'. Đây có thể là tệp dữ liệu a data.framehoặc a listhoặc R được tạo bằng savehoặc NULLhoặc môi trường. Xem thêm 'Chi tiết'.

pos: số nguyên xác định vị trí ở search()nơi cần đính kèm.

name: tên để sử dụng cho cơ sở dữ liệu đính kèm. Tên bắt đầu bằng package

warn.conflicts: Nếu TRUE, cảnh báo được in conflictsra từ việc đính kèm cơ sở dữ liệu, trừ khi cơ sở dữ liệu đó chứa một đối tượng .conflicts.OK. Xung đột là một chức năng che một chức năng hoặc một phi chức năng che một phi chức năng

Ví dụ:

data = data.frame(x1 = c(9, 8, 3, 4, 8),

x2 = c(5, 4, 7, 1, 1),

x3 = c(1, 2, 3, 4, 5))

attach(data)

1. str

Mục đích: Hiển thị nhỏ gọn hình ảnh str bên trong của một đối tượng R , một chức năng chẩn đoán và một sự thay thế cho summary(và ở một mức độ nào đó, dput). Tốt nhất, chỉ một dòng cho mỗi cấu trúc 'cơ bản' được hiển thị. Nó đặc biệt phù hợp để hiển thị nhỏ gọn nội dung (viết tắt) của danh sách (có thể lồng nhau). Ý tưởng là cung cấp đầu ra hợp lý cho bất kỳ đối tượng R nào . Nó gọi argscác đối tượng hàm (không nguyên thủy).

str(object, …)

Object: bất kỳ đối tượng R nào mà bạn muốn có một số thông tin.

Ví dụ

a = list(2, 4, 5, 6, 7, 9, 13, 15, 3, 1)

str(a)

1. factor

Mục đích: Hàm factor được sử dụng để mã hóa một vectơ dưới dạng một yếu tố (các thuật ngữ 'danh mục' và 'kiểu liệt kê' cũng được sử dụng cho các yếu tố). Nếu orderedlà đối số TRUE, các mức yếu tố được giả định là có thứ tự. Để tương thích với S cũng có một chức năng ordered.

factor(x = character(), levels, labels = levels,

exclude = NA, ordered = is.ordered(x), nmax = NA)

x: một vectơ dữ liệu, thường nhận một số lượng nhỏ các giá trị riêng biệt.

Levels: một vectơ tùy chọn của các giá trị duy nhất (dưới dạng chuỗi ký tự) xcó thể đã nhận. Giá trị mặc định là bộ duy nhất của các giá trị được thực hiện bằng cách as.character(x), sắp xếp vào thứ tự tăng dần củax . Lưu ý rằng tập hợp này có thể được chỉ định là nhỏ hơn sort(unique(x)).

Labels: hoặc là một vectơ ký tự tùy chọn của các nhãn cho các cấp (theo thứ tự như levelssau khi loại bỏ các cấp đó exclude) hoặc một chuỗi ký tự có độ dài 1. Các giá trị trùng lặp trong labelscó thể được sử dụng để ánh xạ các giá trị khác nhau của xcùng một cấp yếu tố.

Exclude: một vectơ của các giá trị được loại trừ khi hình thành tập các mức. Đây có thể là yếu tố Ordered: lôgic để xác định xem các mức có được coi là có thứ tự hay không (theo thứ tự đã cho).

Nmax: giới hạn trên về số cấp độ;

Ví dụ:

sex <- factor(c("male", "female", "female", "male"))

levels(sex)

# Chương 4: Hàm đơn giản (function)

1. tapply

Mục đích: Áp dụng một hàm cho mỗi ô của một mảng bị rách, nghĩa là cho mỗi nhóm giá trị (không trống) được cung cấp bởi sự kết hợp duy nhất của các cấp độ của các yếu tố nhất định.

tapply (X, INDEX, FUN = NULL,…, default = NA, simple = TRUE)

Ý nghĩa các tham số:

x: một đối tượng R mà một splitphương thức tồn tại. Điển hình giống như vectơ.

index: một listtrong số một hoặc nhiều factors, mỗi s có cùng độ dài X. Các yếu tố được ép buộc với các yếu tố bởi as.factor.

fun: một chức năng (hoặc tên của một chức năng) được áp dụng, hoặc NULL. Trong trường hợp các hàm như +, %\*%v.v., tên hàm phải được trích dẫn lại hoặc trích dẫn. Nếu FUNcó NULL, tapply trả về một vectơ có thể được sử dụng để chỉ số dưới mảng nhiều chiều tapplythường tạo ra.

…: đối số tùy chọn cho FUN: phần Ghi chú.

default: (chỉ trong trường hợp đơn giản hóa thành một mảng) giá trị mà mảng được khởi tạo bằng array(default, dim = ..). Trước R 3.4.0, điều này được mã hóa cứng array()thành mặc định NA. Nếu là NA(mặc định), giá trị còn thiếu của loại câu trả lời, ví dụ NA\_real\_, sẽ được chọn ( as.raw(0)cho "raw"). Trong trường hợp số, nó có thể được đặt, ví dụ, thành FUN(integer(0)), ví dụ, trong trường hợp FUN = sumto 0hoặc 0L.

Simplify: if FALSE, tapplyluôn trả về một mảng chế độ "list"; nói cách khác, a listvới một dimthuộc tính. Nếu TRUE(mặc định), thì if FUNluôn trả về một đại lượng vô hướng, tapplytrả về một mảng có chế độ vô hướng.

Ví dụ:

tdata = as.data.frame(cbind(c(1,1,1,1,1,2,2,2,2,2), my.matrix))

colnames(tdata)

tapply(tdata$V2, tdata$V1, mean)

1. Sapply

Mục đích: Sapply tương đương với tapply, ngoại trừ việc nó giữ nguyên tên thứ nguyên và thứ nguyên của đối số X. Nó cũng bảo toàn thứ nguyên của kết quả của hàm FUN. Nó được thiết kế để áp dụng cho các kết quả, ví dụ như một cuộc gọi đến by. Lapplylà một tương tự với lapplytrong chừng mực vì nó không cố gắng đơn giản hóa kết quả listcủa các kết quả của FUN.

Công thức:

sapply(X, FUN, ..., simplify = TRUE, USE.NAMES = TRUE)

Ý nghĩa các tham số:

x: một vectơ hoặc danh sách thích hợp cho một cuộc gọi đến sapply.

Fun: một chức năng.

...: đối số tùy chọn cho FUN.

Simplify: một giá trị logic; kết quả có nên được đơn giản hóa thành một vectơ hoặc ma trận nếu có thể

USE.NAMES: if TRUEvà if Xlà ký tự, sử dụng Xlàm tên cho kết quả trừ khi nó đã có tên.

Ví dụ:

berkeley <- Aggregate(Table(Admit,Freq)~.,data=UCBAdmissions)

berktest1 <- By(~Dept+Gender,

glm(cbind(Admitted,Rejected)~1,family="binomial"),

data=berkeley)

berktest2 <- By(~Dept,

glm(cbind(Admitted,Rejected)~Gender,family="binomial"),

data=berkeley)

sapply(berktest1,coef)

Sapply(berktest1,coef)

1. lapply

Mục đích: lapply trả về một danh sách có cùng độ dài X, mỗi phần tử là kết quả của việc áp dụng FUNcho phần tử tương ứng của X.

lapply (X, FUN,…)

Ý nghĩa các tham số:

x: một vectơ (nguyên tử hoặc danh sách) hoặc một expressionđối tượng. Các đối tượng khác (bao gồm các đối tượng được phân loại) sẽ bị cưỡng chế bởi .base::as.list

Fun: chức năng được áp dụng cho từng phần tử của X: xem 'Chi tiết'. Trong trường hợp các hàm như +,, %\*% tên hàm phải được trích dẫn lại hoặc trích dẫn.

…: đối số tùy chọn cho FUN.

A = c(1:9)

B = c(1:12)

C = c(1:15)

my.lst = list(A,B,C)

lapply(my.lst, sum)

1. sd

Mục đích: Hàm này tính toán độ lệch chuẩn của các giá trị trong x. Nếu na.rmlà TRUEthì các giá trị bị thiếu sẽ bị loại bỏ trước khi tính toán tiếp tục.

Công thức:

sd (x, na.rm = FALSE)

Ý nghĩa các tham số:

x: một vectơ số hoặc một đối tượng R nhưng không phải là một đối tượng factorbắt buộc đối với số bởi as.double(x).

na.rm: logic. Có nên xóa các giá trị bị thiếu không?

Ví dụ:

sd(1:2) ^ 2

1. length

Mục đích: Lấy hoặc đặt độ dài của vectơ (bao gồm danh sách) và các thừa số, và của bất kỳ đối tượng R nào khác mà phương thức đã được xác định.

length(x)

length(x) <- value

x: một đối tượng R. Để thay thế, một vectơ hoặc nhân tố.

Values: một số nguyên không âm hoặc kép (sẽ được làm tròn xuống).

length(diag(4))

length(options())

1. summary

Mục đích: Tóm tắt một đối tượng

Công thức:

summary(object, …)

Ý nghĩa các tham số:

Object: một đối tượng mà bản tóm tắt được mong muốn.

Ví dụ:

summary(attenu, digits = 4)

summary(attenu $ station, maxsum = 20)

1. table

Mục đích: table sử dụng các yếu tố phân loại chéo để xây dựng một bảng dự phòng về số lượng ở mỗi sự kết hợp của các cấp độ yếu tố.

Công thức:

table(…,

exclude = if (useNA == "no") c(NA, NaN),

useNA = c("no", "ifany", "always"),

dnn = list.names(…), deparse.level = 1)

Ý nghĩa các tham số:

…: một hoặc nhiều đối tượng có thể được hiểu là các yếu tố (bao gồm các chuỗi ký tự) hoặc một danh sách (hoặc khung dữ liệu) có các thành phần có thể được hiểu như vậy. (Đối với as.table, các đối số được chuyển đến các phương thức cụ thể; đối với as.data.frame, không sử dụng.)

Exclude: các cấp độ để loại bỏ cho tất cả các yếu tố trong …. Nếu nó không chứa NAvà useNAkhông được chỉ định, nó ngụ ý useNA = "ifany". Xem 'Chi tiết' để biết cách diễn giải cho các đối số không thuộc yếu tố.

useNA: có bao gồm NAcác giá trị trong bảng hay không. Xem chi tiết'. Có thể viết tắt.

dnn: tên sẽ được cung cấp cho các thứ nguyên trong kết quả ( tên dimnames ).

deparse.level: kiểm soát cách dnncấu trúc mặc định . Xem chi tiết'.

Ví dụ:

a = letters[1:3]

table(a, sample(a)) # dnn is c("a", "")

table(a, sample(a), deparse.level = 0) # dnn is c("", "")

table(a, sample(a), deparse.level = 2) # dnn is c("a", "sample(a)")

# Chương 5: Plot

1. plot

Mục đích: Hàm chung để vẽ các đối tượng R

Công thức:

plot(x, y, …)

Ý nghĩa các tham số:

x: tọa độ của các điểm trong âm mưu. Ngoài ra, có thể cung cấp một cấu trúc biểu đồ đơn lẻ, hàm hoặc bất kỳ đối tượng R nào có plotphương thức .

y: tọa độ y của các điểm trong biểu đồ, tùy chọn nếu xlà một cấu trúc thích hợp.

…: Các đối số được chuyển tới các phương thức, chẳng hạn như các tham số đồ họa (xem par). Nhiều phương thức sẽ chấp nhận các đối số

Ví dụ:

plot(table(rpois(100, 5)), type = "h", col = "red", lwd = 10,

main = "rpois(100, lambda = 5)")

1. lines

Mục đích: Một hàm tổng quát lấy các tọa độ được đưa ra theo nhiều cách khác nhau và nối các điểm tương ứng với các đoạn thẳng.

Công thức:

lines(x, …)

Ý nghĩa các tham số:

x: tọa độ các vectơ của các điểm tham gia.

…: Thông số đồ họa

Ví dụ:

plot(cars, main = "Stopping Distance versus Speed")

lines(stats::lowess(cars))

1. loess

Mục đích: Phù hợp với một bề mặt đa thức được xác định bởi một hoặc nhiều dự đoán số, sử dụng khớp cục bộ.

Công thức:

loess(formula, data, weights, subset, na.action, model = FALSE,

span = 0.75, enp.target, degree = 2,

parametric = FALSE, drop.square = FALSE, normalize = TRUE,

family = c("gaussian", "symmetric"),

method = c("loess", "model.frame"),

control = loess.control(…), …)

Ý nghĩa các tham số:

formula: một công thức chỉ định phản hồi số và một đến bốn yếu tố dự đoán số (được chỉ định tốt nhất thông qua một tương tác, nhưng cũng có thể được chỉ định một cách cộng thêm). Sẽ bị ép buộc vào một công thức nếu cần thiết.

data: một khung dữ liệu, danh sách hoặc môi trường tùy chọn (hoặc đối tượng bị cưỡng chế bởi as.data.framekhung dữ liệu) có chứa các biến trong mô hình. Nếu không được tìm thấy trong data, các biến sẽ được lấy từ environment(formula), thường là môi trường mà từ đó loessđược gọi

weights: trọng lượng tùy chọn cho từng trường hợp

subset: một đặc tả tùy chọn của một tập hợp con dữ liệu sẽ được sử dụng.

na.action : hành động cần thực hiện với các giá trị bị thiếu trong phản hồi hoặc các yếu tố dự đoán. Mặc định được đưa ra bởi getOption

Model: khung mô hình có nên được trả lại không?

Span: tham số \ (\ alpha \) kiểm soát mức độ làm mịn.

enp.target: một cách thay thế để chỉ định span, làm số lượng tham số tương đương gần đúng sẽ được sử dụng.

Degree : bậc của đa thức sẽ được sử dụng, thường là 1 hoặc 2. (Độ 0 cũng được phép, nhưng hãy xem phần 'Lưu ý'.)

Parametric: có nên áp dụng bất kỳ điều khoản nào trên toàn cầu thay vì cục bộ không? Các thuật ngữ có thể được xác định bằng tên, số hoặc dưới dạng vectơ logic có cùng độ dài với số lượng các yếu tố dự đoán.

drop.square: để phù hợp với nhiều hơn một dự đoán và degree = 2, có nên bỏ thuật ngữ bậc hai cho các dự đoán cụ thể không? Các điều khoản được chỉ định theo cách tương tự như đối với parametric

Normalize: Các yếu tố dự đoán có nên được chuẩn hóa thành một thang đo chung nếu có nhiều hơn một? Chuẩn hóa được sử dụng là đặt độ lệch chuẩn được cắt bớt 10% thành một. Đặt thành false cho các công cụ dự đoán tọa độ không gian và các công cụ khác được biết là nằm trên một tỷ lệ chung.

Family: nếu sự "gaussian"phù hợp là bình phương nhỏ nhất và nếu công "symmetric"cụ ước lượng M giảm dần được sử dụng với hàm số hai của Tukey. Có thể viết tắt

Method: phù hợp với mô hình hoặc chỉ trích xuất khung mô hình. Có thể viết tắt.

Control: thông số điều khiển:

…: Các thông số điều khiển cũng có thể được cung cấp trực tiếp ( nếu control không được chỉ định.

Ví dụ:

cars.lo = loess(dist ~ speed, cars)

predict(cars.lo, data.frame(speed = seq(5, 30, 1)), se = TRUE)

cars.lo2 = loess(dist ~ speed, cars,

control = loess.control(surface = "direct"))

predict(cars.lo2, data.frame(speed = seq(5, 30, 1)), se = TRUE)

1. fitted

Mục đích: fitted là một hàm chung trích xuất các giá trị phù hợp từ các đối tượng được trả về bởi các hàm mô hình hóa. fitted.valueslà một bí danh cho nó.

Tất cả các lớp đối tượng được trả về bởi các hàm phù hợp với mô hình phải cung cấp một fittedphương thức. (Lưu ý rằng chung chung là fittedvà không fitted.values.)

Các phương thức có thể sử dụng các napredictphương pháp để bù đắp cho sự thiếu sót của các giá trị bị thiếu. Mặc định và nlscác phương thức làm.

fitted(object, …)

Object: một đối tượng mà việc trích xuất các giá trị phù hợp với mô hình là có ý nghĩa.

…: các đối số khác.

# Chương 6: Học loop function

1. jpeg

Mục đích: Đây là một trình bao bọc cho hàm jpeg của R

Công thức:

jpeg(file, Rcss = "default", Rcssclass = NULL, ...)

Ý nghĩa các tham số:

file: chuỗi ký tự với tên tệp

Rcss: đối tượng bảng kiểu. Để "mặc định" để sử dụng một kiểu được xác định qua RcssSetDefaultStyle ()

Rcssclass: nhân vật, lớp phong cách

...: Các thông số khác

Ví dụ:

jpeg(file="example-file.jpg")

barplot(1:5)

1. dev.off

Mục đích: Hàm này đóng ô được chỉ định (theo mặc định là thiết bị hiện tại) và nếu đó là thiết bị imguR, hãy tải các ô lên để lưu trữ web

Công thức:

dev.off (which = dev.cur (), justLink = TRUE)

Ý nghĩa các tham số:

which: Một số nguyên chỉ định một số thiết bị

justlink: A booleancho biết liệu chỉ trả về một liên kết.

Ví dụ:

imguR()

plot(x=1:10, y=-1:-10, col=1:10, pch=19, main="oh hai dere")

cat(dev.off())

1. function

Mục đích: cung cấp các cơ chế cơ bản để xác định các hàm mới trong ngôn ngữ R.

function( arglist ) expr

return(value)

arglisr: Rỗng hoặc một hoặc nhiều tên hoặc tên = các thuật ngữ biểu thức.

expr: Một biểu thức.

value: Một biểu thức.

Ví dụ:

(function(x, y){ z <- x^2 + y^2; x+y+z })(0:7, 1)

1. paste

Mục đích: Nối các vectơ sau khi chuyển đổi thành ký tự.

Công thức:

paste (…, sep = " ", collapse = NULL)

Ý nghĩa các tham số:

…: một hoặc nhiều đối tượng R , được chuyển đổi thành vectơ ký tự.

Sep: một chuỗi ký tự để phân tách các điều khoản. Không NA\_character\_.

Collapse: một chuỗi ký tự tùy chọn để phân tách các kết quả. Không NA\_character\_.

Ví dụ:

paste(1:12)

as.character(1:12)

1. if

Mục đích: Hàm IF là một trong những hàm logic đầu tiên có 3 đối số, kiểm tra logic, giá trị nếu đúng và giá trị nếu sai. Nếu kiểm tra logic vượt qua (nghĩa là nó trả về true) thì hàm chuyển sang giá trị nếu đúng, ngược lại nó chuyển sang giá trị nếu đối số sai.

IF (logic\_test, valueifTrue = 0, valueifFalse = 0)

Logic\_test: Đây là bài kiểm tra thông thường chúng tôi chạy trong excel trả về giá trị TRUE hoặc FALSE. Sử dụng dấu bằng kép để kiểm tra logic nếu bạn muốn bằng nhau.

valueifTrue: Nếu logic\_test đánh giá là TRUE thì hàm sẽ trả về giá trị bạn nhập vào đây

valueifFalse: Nếu logic\_test đánh giá là FALSE thì hàm sẽ trả về giá trị bạn nhập vào đây

Ví dụ:

IF(iris$Species == "virginica","Yes","No")

1. ifelse

Mục đích: ifelse trả về một giá trị có cùng hình dạng testđược lấp đầy bởi các phần tử được chọn từ một trong hai yeshoặc notùy thuộc vào việc phần tử của testlà TRUEhoặc FALSE.

ifelse(test, yes, no)

test : một đối tượng có thể bị ép buộc sang chế độ logic.

yes: trả về giá trị cho các phần tử true của test.

no: trả về giá trị cho các phần tử sai của test.

Ví dụ:

x = c(6:-4)

sqrt(x)

sqrt(ifelse(x >= 0, x, NA))

ifelse(x >= 0, sqrt(x), NA)

1. ff\_else

Mục đích: So với cơ sở ifelse(), chức năng này nghiêm ngặt hơn. Nó kiểm tra cái đó truevà falsecùng loại. Sự nghiêm ngặt này làm cho loại đầu ra dễ đoán hơn và làm cho nó nhanh hơn một chút.

Công thức:

if\_else(condition, true, false, missing = NULL)

Ý nghĩa các tham số:

condition: Vectơ logic

True, false: Giá trị để sử dụng TRUEvà FALSEgiá trị của condition. Chúng phải có cùng độ dài conditionhoặc bằng 1. Chúng cũng phải cùng loại: if\_else()kiểm tra xem chúng có cùng kiểu và cùng lớp hay không. Tất cả các thuộc tính khác được lấy từ true.

Missing: Nếu không NULL, sẽ được sử dụng để thay thế các giá trị bị thiếu.

Ví dụ:

x = c(-5:5, NA)

if\_else(x < 0, NA\_integer\_, x)

if\_else(x < 0, "negative", "positive", "missing")

# Chương 7:

1. pie

Mục đích: Vẽ biểu đồ hình tròn.

Công thức:

pie(x, labels = names(x), edges = 200, radius = 0.8,

clockwise = FALSE, init.angle = if(clockwise) 90 else 0,

density = NULL, angle = 45, col = NULL, border = NULL,

lty = NULL, main = NULL, …)

Ý nghĩa các tham số:

x: một vectơ của các đại lượng số không âm. Các giá trị trong xđược hiển thị dưới dạng diện tích của lát bánh.

Labels: một hoặc nhiều biểu thức hoặc chuỗi ký tự cung cấp tên cho các lát. Các đối tượng khác bị cưỡng chế bởi as.graphicsAnnot. Đối với các nhãn trống hoặc NA(sau khi ép buộc thành ký tự), không có nhãn hoặc đường trỏ nào được vẽ.

Edges: đường viền hình tròn của chiếc bánh được xấp xỉ bởi một đa giác có nhiều cạnh này.

Radius: chiếc bánh được vẽ ở giữa trong một hộp hình vuông có các cạnh nằm trong khoảng từ \ (- 1 \) đến \ (1 \). Nếu các chuỗi ký tự ghi nhãn các lát dài thì có thể cần sử dụng bán kính nhỏ hơn.

Clockwise: lôgic cho biết nếu các lát cắt được vẽ theo chiều kim đồng hồ hoặc ngược chiều kim đồng hồ (tức là theo chiều dương về mặt toán học), thì giá trị sau là mặc định.

init.angle: số chỉ định góc bắt đầu (tính bằng độ) cho các lát cắt. Mặc định là 0 (tức là '3 giờ') trừ khi clockwiseđúng Ý nghĩa các tham số giá trị init.anglemặc định là 90 (độ), (tức là '12 giờ').

Density: mật độ của các đường tô bóng, tính bằng dòng trên inch. Giá trị mặc định của NULLnghĩa là không có đường tô bóng nào được vẽ. Các giá trị không tích cực của densitycũng ngăn cản việc vẽ các đường

Angle: độ dốc của các đường tô bóng, được cho dưới dạng góc tính bằng độ (ngược chiều kim đồng hồ).

col: một vector màu được sử dụng để tô hoặc tô bóng cho các lát cắt. Nếu thiếu bộ 6 màu phấn được sử dụng, trừ khi densityđược chỉ định khi nào par("fg")được sử dụng.

Border, lty: (có thể là vectơ) đối số được truyền tới để polygonvẽ từng lát cắt.

Main: một tiêu đề tổng thể cho cốt truyện.

…: các tham số đồ họa có thể được cung cấp dưới dạng đối số cho pie. Chúng sẽ chỉ ảnh hưởng đến tiêu đề và nhãn chính.

Ví dụ:

require(grDevices)

pie(rep(1, 24), col = rainbow(24), radius = 0.9)

1. pie3D

Mục đích: Hiển thị biểu đồ hình tròn 3D với các nhãn tùy chọn.

Công thức:

pie3D (x, edge = NA, radius = 1, height = 0.1, theta = pi / 6, start = 0, border = par ("fg"),

col = NULL, label = NULL, labelpos = NULL, labelcol = par ("fg"), labelcex = 1.5,

sector.order = NULL, boom = 0, shadow = 0.8, mar = c (4,4,4,4), pty = "s", ...)

Ý nghĩa các tham số:

x: một vectơ số mà mỗi giá trị sẽ là một khu vực

edges: số đường tạo thành một hình elip

eadius: bán kính của chiếc bánh tính bằng đơn vị người dùng

height: chiều cao của chiếc bánh tính bằng đơn vị người dùng

theta: Góc xem tính bằng radian

start: Góc bắt đầu vẽ các cung.

border: Màu của các đường viền khu vực

col: Màu sắc của các ngành

labels: Các nhãn tùy chọn cho từng lĩnh vực

labelpos: Các vị trí tùy chọn cho các nhãn (xem ví dụ)

labelcol: Màu sắc của nhãn

labelcex: Hệ số mở rộng ký tự cho các nhãn

sector.order: Cho phép nhà điều hành chỉ định thứ tự mà các cung được vẽ.

explode: Số lượng để làm "nổ" chiếc bánh tính theo đơn vị người dùng

shade: Nếu> 0 và <1, tỷ lệ giảm độ sáng của màu khu vực để có được hiệu ứng 3D tốt hơn.

mar: Lề xung quanh chiếc bánh.

pty: Có buộc một vùng ô vuông hay không.

...: thông số đồ họa được chuyển tới Plot

Ví dụ:

pie3D(pieval,radius=0.9,labels=pielabels,explode=0.1,main="3D PIE OPINIONS",

col=c("brown","#ddaa00","pink","#dd00dd"),labelpos=lp)

1. par

Mục đích: par có thể được sử dụng để thiết lập hoặc truy vấn các tham số đồ họa. Các tham số có thể được đặt bằng cách chỉ định chúng dưới dạng đối số par trong tag = valuebiểu mẫu hoặc bằng cách chuyển chúng dưới dạng danh sách các giá trị được gắn thẻ.

Công thức:

par(…, no.readonly = FALSE)

(..., = )

Ý nghĩa các tham số:

…: đối số trong tag = valuebiểu mẫu hoặc danh sách các giá trị được gắn thẻ. Các thẻ phải đến từ tên của các tham số đồ họa được mô tả trong phần 'Tham số đồ họa'.

no.readonly: nếu TRUEvà không có đối số nào khác, chỉ các tham số được trả về có thể được thiết lập bởi một par()cuộc gọi tiếp theo trên cùng một thiết bị .

op = par(mfrow = c(2, 2), # 2 x 2 pictures on one plot

pty = "s")

par(op)

op = par(no.readonly = TRUE)

1. barplot

Mục đích: Tạo một biểu đồ thanh với các thanh dọc hoặc ngang.

Công thức:

barplot(height, …)

Ý nghĩa các tham số:

height: hoặc là một vectơ hoặc ma trận các giá trị mô tả các thanh tạo nên biểu đồ. Nếu heightlà một vectơ, thì biểu đồ bao gồm một dãy các thanh hình chữ nhật với chiều cao được cho bởi các giá trị trong vectơ. Nếu heightlà một ma trận và besidelà FALSEsau đó mỗi thanh của tương ứng với âm mưu một cột height, với các giá trị trong cột cho chiều cao của chồng tiểu thanh chiếm quầy bar. Nếu heightlà một ma trận và besidelà TRUE, thì các giá trị trong mỗi cột được đặt cạnh nhau thay vì xếp chồng lên nhau.

Ví dụ:

barplot(GNP ~ Year, data = longley)

barplot(cbind(Employed, Unemployed) ~ Year, data = longley)

op = par(mfrow = 2:1, mgp = c(3,1,0)/2, mar = .1+c(3,3:1))

summary(d.Titanic <- as.data.frame(Titanic))

barplot(Freq ~ Class + Survived, data = d.Titanic,

subset = Age == "Adult" & Sex == "Male",

main = "barplot(Freq ~ Class + Survived, \*)", ylab = "# {passengers}", legend = TRUE)

1. arrows

Mục đích: Vẽ mũi tên giữa các cặp điểm.

Công thức:

arrows(x0, y0, x1 = x0, y1 = y0, length = 0.25, angle = 30,

code = 2, col = par("fg"), lty = par("lty"),

lwd = par("lwd"), …)

Ý nghĩa các tham số:

x0, y0: tọa độ của các điểm để từ đó vẽ.

x1, y1: tọa độ của điểm đến để vẽ. Ít nhất một phải được cung cấp

Length: chiều dài của các cạnh của đầu mũi tên (tính bằng inch).

Góc: góc từ trục của mũi tên đến mép của đầu mũi tên.

Mã: mã số nguyên, xác định loại mũi tên sẽ được vẽ.

col, lty, lwd: các tham số đồ họa , các vectơ có thể. NAgiá trị colkhiến mũi tên bị bỏ qua.

…: các tham số đồ họa chẳng hạn như xpdvà các đặc tính của đường lend, ljoinvà lmitre: par

Ví dụ:

x <- stats::runif(12); y <- stats::rnorm(12)

i <- order(x, y); x <- x[i]; y <- y[i]

plot(x,y, main = "arrows(.) and segments(.)")

1. box

Mục đích: Hàm này vẽ một hộp xung quanh biểu đồ hiện tại với màu sắc và kiểu đường đã cho. Các btytham số xác định loại hộp rút ra.

Công thức:

box (which = "plot", lty = "solid",…)

Ý nghĩa các tham số:

which: nhân vật, một trong những "plot", "figure", "inner"và "outer".

Lty: loại dòng của hộp.

…: thêm các thông số đồ họa , chẳng hạn như bty, colhoặc lwd, xem par. Lưu ý rằng điều đó xpdkhông được chấp nhận vì phần cắt luôn dành cho vùng thiết bị.

Ví dụ:

plot(1:7, abs(stats::rnorm(7)), type = "h", axes = FALSE)

axis(1, at = 1:7, labels = letters[1:7])

box(lty = '1373', col = 'red')

1. Boxplot

Mục đích: Tạo (các) biểu đồ hình hộp của các giá trị đã cho (được nhóm lại).

Công thức:

boxplot(x, …)

Ý nghĩa các tham số:

x: để xác định dữ liệu mà từ đó các ô hộp sẽ được tạo ra. Một vectơ số hoặc một danh sách chứa các vectơ như vậy. Các đối số không được đặt tên bổ sung chỉ định thêm dữ liệu dưới dạng vectơ riêng biệt (mỗi vectơ tương ứng với một ô hộp thành phần). NAs được cho phép trong dữ liệu.

…: Đối với formulaphương thức, các đối số được đặt tên sẽ được chuyển cho phương thức mặc định.

Đối với phương thức mặc định, các đối số không được đặt tên là các vectơ dữ liệu bổ sung (trừ khi xlà một danh sách khi chúng bị bỏ qua) và các đối số được đặt tên là các đối số và tham số đồ họa sẽ được chuyển bxpvào ngoài các đối số được cung cấp bởi đối số pars(và ghi đè các đối số Ý nghĩa các tham số pars). Lưu ý rằng bxpcó thể sử dụng hoặc không sử dụng các tham số đồ họa mà nó được truyền: xem tài liệu của nó.

Ví dụ:

boxplot(count ~ spray, data = InsectSprays,

notch = TRUE, add = TRUE, col = "blue")

1. text

Mục đích: vẽ các chuỗi được cho trong vectơ labelstại các tọa độ cho bởi xvà y. ycó thể bị thiếu vì xy.coords(x, y)được sử dụng để xây dựng các tọa độ.

Công thức:

text(x, …)

Ý nghĩa các tham số:

x, y: vectơ số của tọa độ nơi văn bản labelssẽ được viết. Nếu chiều dài của xvà ykhác nhau, cái ngắn hơn sẽ được tái chế.

Ví dụ:

plot(1:10) # Basic plot in R

text(x = 2, y = 3, # Add text element

1. points

Mục đích: Là một hàm tổng quát để vẽ một chuỗi các điểm tại các tọa độ xác định. (Các) ký tự được chỉ định được vẽ, căn giữa tại các tọa độ.

Công thức:

points(x, y = NULL, type = "p", …)

Ý nghĩa các tham số:

x, y: tọa độ các vectơ của điểm để vẽ đồ thị.

Type: ký tự chỉ kiểu âm mưu; thực sự là bất kỳ types như trong plot.default.

…: Các tham số đồ họa khác cũng có thể được cung cấp dưới dạng đối số.

Ví dụ:

plot(-4:4, -4:4, type = "n")

points(rnorm(200), rnorm(200), col = "red")

points(rnorm(100)/2, rnorm(100)/2, col = "blue", cex = 1.5)

1. legend

Mục đích: Thêm chú giải vào các ô. Lưu ý rằng một lệnh gọi hàm locator(1)có thể được sử dụng thay cho các đối số xvà y.

Công thức:

legend(x, y = NULL,...)

Ý nghĩa các tham số:

x, y: tọa độ x và y được sử dụng để định vị chú giải. Chúng có thể được chỉ định theo từ khóa hoặc theo bất kỳ cách nào được chấp nhận bởi xy.coords:

Ví dụ:

x = seq(-pi, pi, len = 65)

plot(x, sin(x), type = "l", ylim = c(-1.2, 1.8), col = 3, lty = 2)

points(x, cos(x), pch = 3, col = 4)

lines(x, tan(x), type = "b", lty = 1, pch = 4, col = 6)

title("legend(..., lty = c(2, -1, 1), pch = c(NA, 3, 4), merge = TRUE)",

cex.main = 1.1)

legend(-1, 1.9, c("sin", "cos", "tan"), col = c(3, 4, 6),

text.col = "green4", lty = c(2, -1, 1), pch = c(NA, 3, 4),

merge = TRUE, bg = "gray90")

1. title

Mục đích: Chức năng này có thể được sử dụng để thêm nhãn vào một biểu đồ.

title(main = NULL, sub = NULL, xlab = NULL, ylab = NULL,

line = NA, outer = FALSE, …)

Main: Tiêu đề chính (trên cùng) sử dụng phông chữ, kích thước (mở rộng ký tự) và màu sắc par(c("font.main", "cex.main", "col.main"

Sub: Tiêu đề phụ (ở dưới cùng) sử dụng phông chữ, kích thước và màu sắc par(c("font.sub", "cex.sub", "col.sub")).

Xlab: Nhãn trục X sử dụng phông chữ, kích thước và màu sắc par(c("font.lab", "cex.lab", "col.lab")).

Ylab: Nhãn trục Y, các thuộc tính phông chữ giống như xlab.

Line: chỉ định một giá trị để lineghi đè vị trí mặc định của các nhãn và đặt chúng nhiều dòng này ra ngoài từ cạnh của biểu đồ.

Outer: một giá trị logic. Nếu TRUE, tiêu đề được đặt ở lề ngoài của cốt truyện.

…: các thông số đồ họa khác từ par. Sử dụng ví dụ, col.mainhoặc cex.subthay vì chỉ colhoặc cex. adjkiểm soát sự biện minh của các tiêu đề. xpdcó thể được sử dụng để đặt vùng cắt: vùng này mặc định là vùng hình trừ khi outer = TRUE, nếu không thì vùng thiết bị và chỉ có thể được tăng lên. mgpkiểm soát vị trí mặc định của tiêu đề trục.

Ví dụ:

plot(cars, main = "") # here, could use main directly

title(main = "Stopping Distance versus Speed")

plot(cars, main = "")

title(main = list("Stopping Distance versus Speed", cex = 1.5,

col = "red", font = 3))

1. expression

Mục đích: Tạo hoặc thử nghiệm các đối tượng của chế độ "expression".

expression(…)

…: expression: R các đối tượng, thường là lời gọi, ký hiệu hoặc hằng số. as.expression: các đối số được truyền cho các phương thức.

x: một đối tượng R tùy ý .

Ví dụ

length(ex1 <- expression(1 + 0:9))

ex1

eval(ex1) # 1:10

1. pairs

Mục đích: Một ma trận các biểu đồ phân tán được tạo ra.

pairs(x, …)

x: tọa độ của các điểm được cho dưới dạng cột số của ma trận hoặc khung dữ liệu. Các cột logic và cột yếu tố được chuyển đổi thành số theo cách tương tự data.matrix.

…: các đối số được chuyển đến hoặc từ các phương thức.

Ngoài ra, các tham số đồ họa có thể được cung cấp như các đối số có thể plotchẳng hạn main. par("oma")sẽ được thiết lập thích hợp trừ khi được chỉ định.

Ví dụ

pairs(iris[1:4], main = "Anderson's Iris Data -- 3 species",

pch = 21, bg = c("red", "green3", "blue")[unclass(iris$Species)])

1. coplot

Mục đích: Hàm này tạo ra hai biến thể của các biểu đồ đồng điều hòa

coplot(formula, data, given.values, panel = points, rows, columns,

show.given = TRUE, col = par("fg"), pch = par("pch"),

bar.bg = c(num = gray(0.8), fac = gray(0.95)),

xlab = c(x.name, paste("Given :", a.name)),

ylab = c(y.name, paste("Given :", b.name)),

subscripts = FALSE,

axlabels = function(f) abbreviate(levels(f)),

number = 6, overlap = 0.5, xlim, ylim, …)

Formula: một công thức mô tả hình thức của điều kiện âm mưu. Một công thức của biểu mẫu y ~ x | achỉ ra rằng các ô yso với xphải được tạo ra có điều kiện đối với biến a. Một công thức của biểu mẫu y ~ x| a \* bchỉ ra rằng các ô y so với x phải được tạo ra có điều kiện đối với hai biến a và b.

Tất cả ba hoặc bốn biến có thể là số hoặc nhân tố. Khi xhoặc ylà các yếu tố, kết quả gần như as.numeric()được áp dụng, trong khi đối với yếu tố ahoặc b, điều kiện (và đồ họa của nó nếu show.givenlà đúng) được điều chỉnh.

Data: một khung dữ liệu chứa các giá trị cho bất kỳ biến nào trong công thức. Theo mặc định, môi trường nơi coplotđược gọi từ được sử dụng.

Given.values: một giá trị hoặc danh sách hai giá trị xác định cách điều chỉnh đang avà bsẽ diễn ra.

Khi không có b(nghĩa là chỉ điều kiện bật a), thường thì đây là một ma trận có hai cột mỗi hàng cho một khoảng, được điều chỉnh trên, nhưng cũng có thể là một vectơ duy nhất của các số hoặc một tập hợp các mức nhân tố ( nếu biến được điều kiện hóa là một thừa số). Trong trường hợp này (không b), kết quả của co.intervalscó thể được sử dụng trực tiếp làm given.valuesđối số.

Panel: a function(x, y, col, pch, …)cung cấp hành động được thực hiện trong mỗi bảng điều khiển của màn hình. Mặc định là points.

Rows: các tấm của âm mưu được sắp xếp rowstheo từng columnsmảng. rowscho biết số hàng trong mảng.

Columns: số lượng cột trong mảng bố cục bảng điều khiển.

show.given: lôgic (có thể có độ dài 2 cho 2 biến điều hòa): đồ thị điều hòa có được hiển thị cho các biến điều hòa tương ứng không (mặc định TRUE).

Col: một vector màu sẽ được sử dụng để vẽ các điểm. Nếu quá ngắn, các giá trị được tái chế.

Pch: một vector biểu tượng hoặc ký tự. Nếu quá ngắn, các giá trị được tái chế.

bar.bg: một vectơ được đặt tên với các thành phần "num"và "fac"cung cấp màu nền cho các thanh (shingle), cho các biến điều hòa num eric và fac tor tương ứng.

xlab: tính cách; nhãn để sử dụng cho trục x và biến điều kiện đầu tiên. Nếu chỉ có một nhãn được đưa ra, nó được sử dụng cho trục x và nhãn mặc định được sử dụng cho biến điều hòa.

Ylab: tính cách; nhãn để sử dụng cho trục y và bất kỳ biến điều kiện thứ hai nào.

Subscripts: lôgic: nếu đúng, chức năng bảng điều khiển được cung cấp một đối số bổ sung (thứ ba) subscriptscung cấp các chỉ số con của dữ liệu được chuyển đến bảng điều khiển đó.

Axlabels: chức năng tạo nhãn trục (đánh dấu) khi x hoặc y là hệ số.

Number: số nguyên; số khoảng điều kiện, đối với a và b, có thể có độ dài 2. Nó chỉ được sử dụng nếu biến điều hòa tương ứng không phải là a factor.

Overlap: số <1; phần trùng lặp của các biến điều hòa, có thể có độ dài 2 đối với hướng x và y. Khi chồng chéo <0, sẽ có khoảng trống giữa các lát dữ liệu.

xlim: phạm vi cho trục x.

ylim: phạm vi cho trục y.

…: đối số bổ sung cho chức năng bảng điều khiển.

x: một vectơ số.

Ví dụ:

coplot(lat ~ long | depth, data = quakes)

given.depth <- co.intervals(quakes$depth, number = 4, overlap = .1)

coplot(lat ~ long | depth, data = quakes, given.v = given.depth, rows = 1)

1. layout

Mục đích: Chia thiết bị thành nhiều hàng và cột như có trong ma trận mat, với chiều rộng cột và chiều cao hàng được chỉ định trong các đối số tương ứng.

layout(mat, widths = rep.int(1, ncol(mat)),

heights = rep.int(1, nrow(mat)), respect = FALSE)

layout.show(n = 1)

Ý nghĩa các tham số:

mat: một đối tượng ma trận xác định vị trí của các hình \ (N \) tiếp theo trên thiết bị đầu ra. Mỗi giá trị trong ma trận phải là 0hoặc một số nguyên dương. Nếu \ (N \) là số nguyên dương lớn nhất trong ma trận, thì các số nguyên \ (\ {1, \ dấu chấm, N-1 \} \) cũng phải xuất hiện ít nhất một lần trong ma trận.

Widths: vectơ giá trị cho độ rộng của các cột trên thiết bị. Chiều rộng tương đối được chỉ định bằng các giá trị số. Chiều rộng tuyệt đối (tính bằng cm) được chỉ định với lcm()hàm (xem ví dụ).

Heights: vectơ giá trị cho độ cao của các hàng trên thiết bị. Chiều cao tương đối và tuyệt đối có thể được chỉ định, xem widthsở trên.

Respect: hoặc một giá trị logic hoặc một đối tượng ma trận. Nếu thứ hai, thì nó phải có cùng kích thước matvà mỗi giá trị trong ma trận phải là 0hoặc 1.

N: số lượng các số liệu để vẽ.

x: một kích thước được hiểu là một số cm.

Ví dụ:

nf <- layout(matrix(c(1,1,0,2), 2, 2, byrow = TRUE), respect = TRUE)

layout.show(nf)