

می توانیم متریک های از این دو را بسازیم.

• ساختار از تابع matrix()

matrix(data, nrow, ncol, byrow = F)

↳ defaults
 که پنج عنصر هستند در data، با مقول مستقیم در ماتریس

• - 5

> A = matrix(1:8, 2, 4, byrow = F)

می توان نوشت

تبدیلیم بجایم بطوری امتیاز رسم که باید آن را با T جایگزین کرد.

> A

	[,1]	[,2]	[,3]	[,4]
[1,]	1	3	5	7
[2,]	2	4	6	8

> AA = matrix(1:8, 2, 4, byrow = T)

T یا

2

> AA = matrix(1:8, 4, T)

> B = matrix(c(1,5,7,8), 2, 2)

• با استفاده از rbind() و cbind() می توانیم دو ماتریس را به ترتیب

همسایه بطوری و بعضی می توانیم ساختار کرد.

> A

1	3	5	7
2	4	6	8

> a = c(-1, -2, -3, -4)

> rbind(a, A)

-1	-2	-3	-4
1	3	5	7
2	4	6	8

• تابع $\text{diag}()$ ماتریس قطری ایجاد می‌کند.

> $\text{diag}(3)$ → ماتریس همانی 3×3 می‌باشد

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

> $\text{diag}(a)$ → ماتریس قطری با عناصر روی قطر

$$\begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 \\ 0 & 0 & -3 & 0 \\ 0 & 0 & 0 & -4 \end{bmatrix}$$

ایجاب بازم مجموعه‌ای از ماتریس‌ها

> $A[2,3]$ → عنصر یکم از ستون سوم A

> $A[2]$ → آری یک سطر 2 در A یعنی بردار 2 در A که 2 در A است
از ستون‌های ماتریس در A می‌گیرد و عنصر دوم آن بردار را می‌دهد.

> $A[1, c(2,4)]$ → یک دوم در A از ستون اول

> $A[, 2]$ → تمام عناصر سطوحی ستون دوم

> $A[, -2]$ → غیر از ستون دوم تمام عناصر را می‌دهد.

> $A[1, -c(2,3)]$ → عناصر سطوح اول به جز ستون دوم و سوم.

• نتایج تابعی در رابطه با ماتریس‌ها

• تابع $\text{dim}()$ ماتریس را با تابع $\text{dim}()$

> $\text{dim}(A)$

> $\text{dim}(A)[2]$ → تعداد ستون‌ها

> $\text{dim}(A)[1]$ → تعداد سطوح‌ها

\sim
> X = 1:12

> dim(X) = c(1, 2, 3)

لیست
• آر جوامع ساختارهای مختلف، کاراکتری، رادیکال و ...
(list) ساختار

> X ← LETTERS(1:3)

> x ← c(1, 5, 6)

> mylist = list(X, x)

> mylist

\$X
[1] a b c → ساختار \$ اسمی اجزای مختلف در list
رشته‌های ساده

\$x
[1] 1 5 6 →

• می‌توان نام مقدرها را در list مقدر داد.

> mylist = list(names = X, scores = x)

• با استفاده از علامت \$ می‌توان به اجزای مختلف list دسترسی داشت.

> mylist\$names
[1] a b c
> mylist[[1]]

• طول لیست نشان دهنده تعداد مؤلفه‌های لیست است.

> length(mylist)

[1] 2

• آر جوامع نام اجزای لیست هستند. این را می‌توانیم تابع

> names(mylist)

[1] "names" "scores"

data.frame

• روشی آید تا در این ساختارهای مختلف داده، در کنار هم، استفاده از تابع (data.frame) است.

• تفاوت این تابع با (list) در این است که data.frame مانند یک ماتریس عمل می کند. در واقع ساختارهای مختلف داده را مانند به داده های دیگر و آسانی دست. لازم است طول تمام به داده ها یکسان باشد (تعداد سطرها برابر). همچنین می توان با ستون های آن اعمال صوری انجام داد.

• یکی با list می توان اعمال صوری انجام داد. اعضای data.frame عنوان ستون نام از تغییرات ساخته نمی شوند. از طرفی، محدودیت تعداد سطرها برابر است. در واقع در data.frame، ساختار نامی برای داده ها، متفاوتی بود. همچنین اعضای data.frame می تواند خودشان ساختار data.frame باشند.

> X = 1:3

> Y = c("a", "b", "h")

> Z.ls = list (myX = X , Y)
اسم دلخواه

> Z.df = data.frame (X , Y)
اسم دلخواه

> Z.ls * 3 → ستون های عددی را ضریب 3 و کلاس های سایر را همواره
حفظی دست.

> Z.df * 3 →

> Z2.ls = list (Z.ls , c(1, 5, 4))

→ اعضای data.frame می تواند data.frame باشد.

as.data.frame ~ data.frame as.list ~ data.frame ~ list

> data.frame(z.ls)

as.list ~ data.frame

> as.list(z.df)

names() ~ data.frame

> names(z.df)

> names(z.ls)