

# Why you should use **R**

free

open source

platform independent

Mac, Windows, Linux

many free packages available

versatility



# Why **you** should use R

get grip on your data

do all analyses you want

be independent of licenses

learn programming

make impressive graphs

have fun



# Object Types

- vector

```
num.vector = c(2, 6, 33, NA)
```

```
log.vector = c(TRUE, FALSE, TRUE)
```

```
char.vector = c('Hello', '', 'R')
```





# Object Types

- `matrix`

`a[3, 2]`

7	4	33	10
2	24	10	1
NA	6	NA	22
16	18	15	31

`apply(a, 2, sum)= 25 52 58 64`



# Object Types

- *array*

1	4	7	10
2	5	8	11
3	6	9	12



# Object Types

- array

	5	4	7	10
1	2	5	8	11
2	4	6	53	2
3	0	9	12	



# Object Types

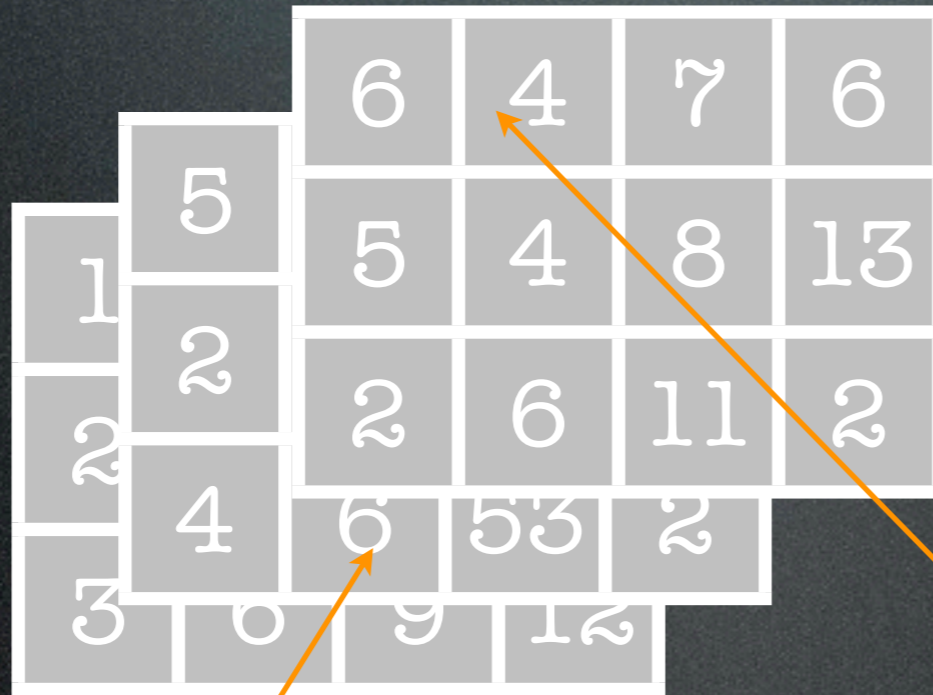
- array

		6	4	7	6
	5	5	4	8	13
1	2	2	6	11	2
2	4	6	5	3	2
3	0	9	12		



# Object Types

- array



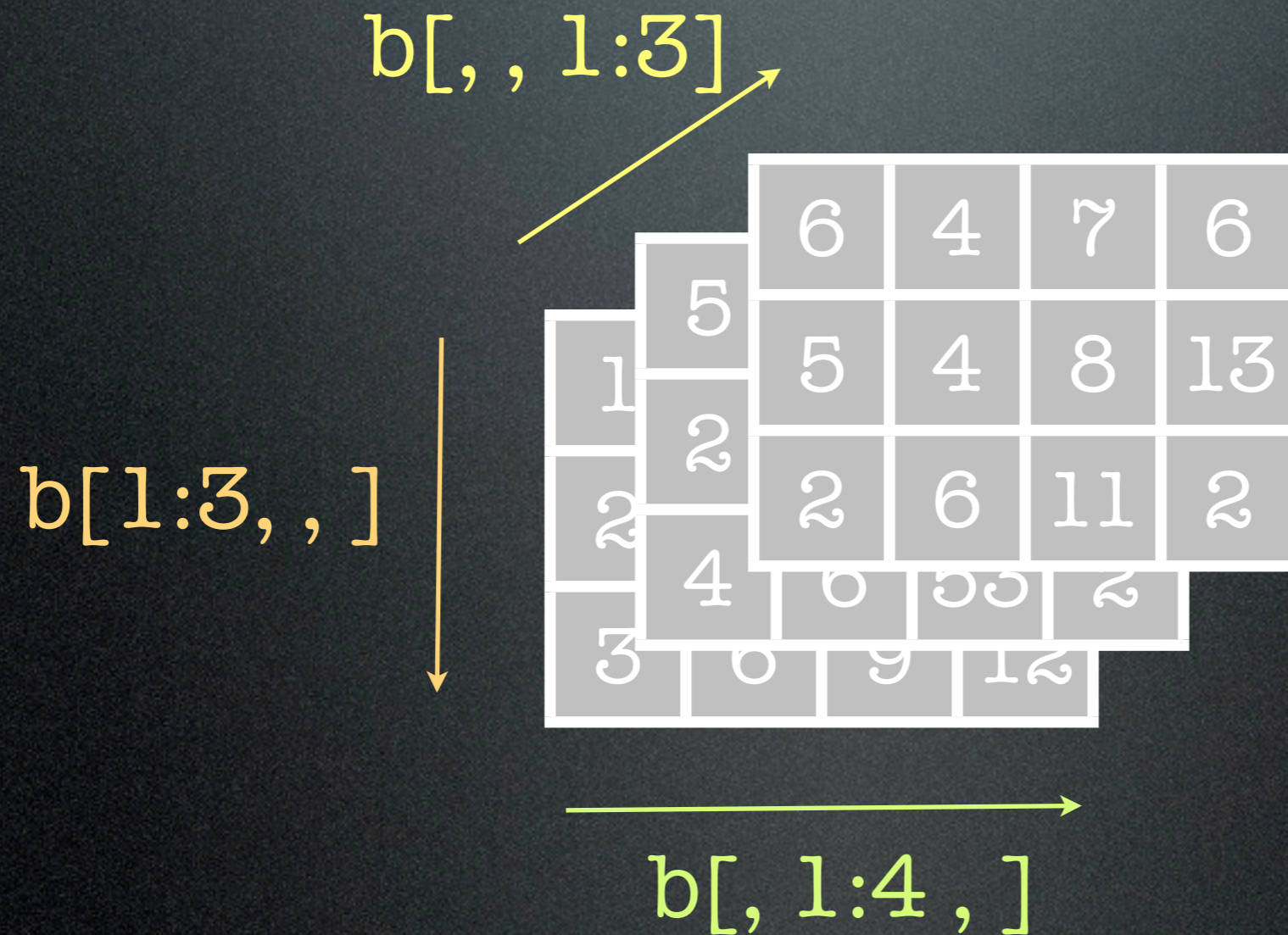
`b[3, 2, 2]`

`b[1, 2, 3]`



# Object Types

- array

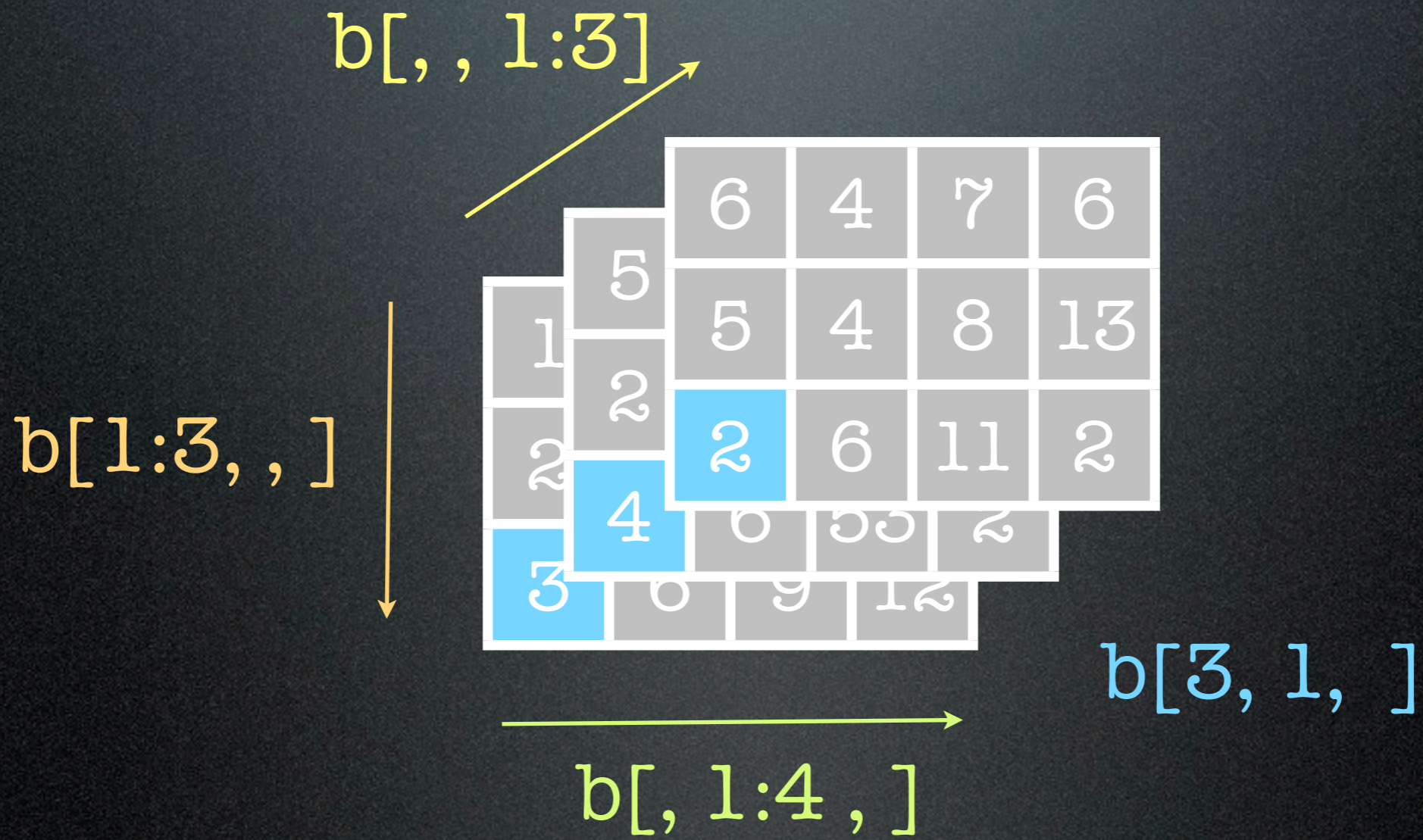






# Object Types

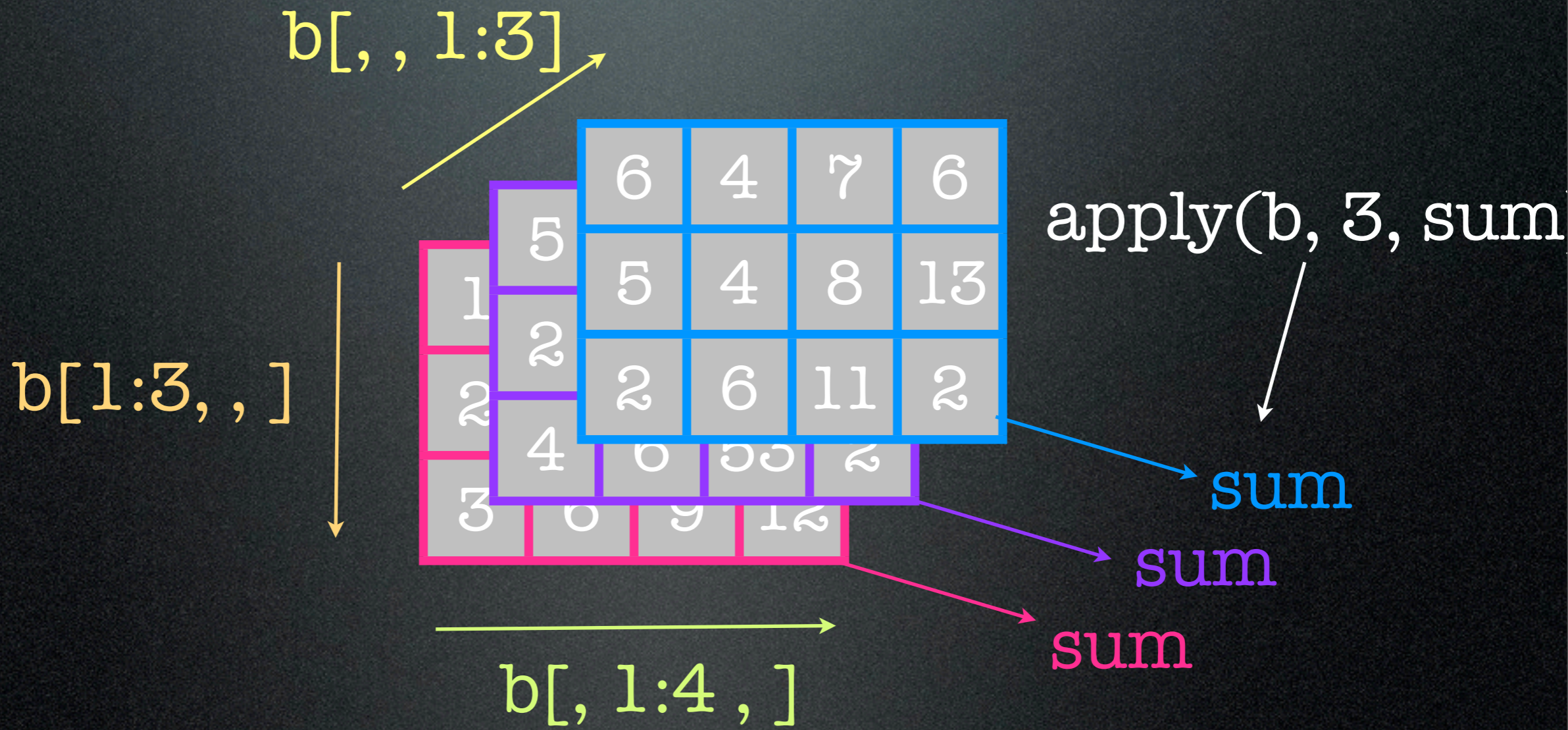
- array





# Object Types

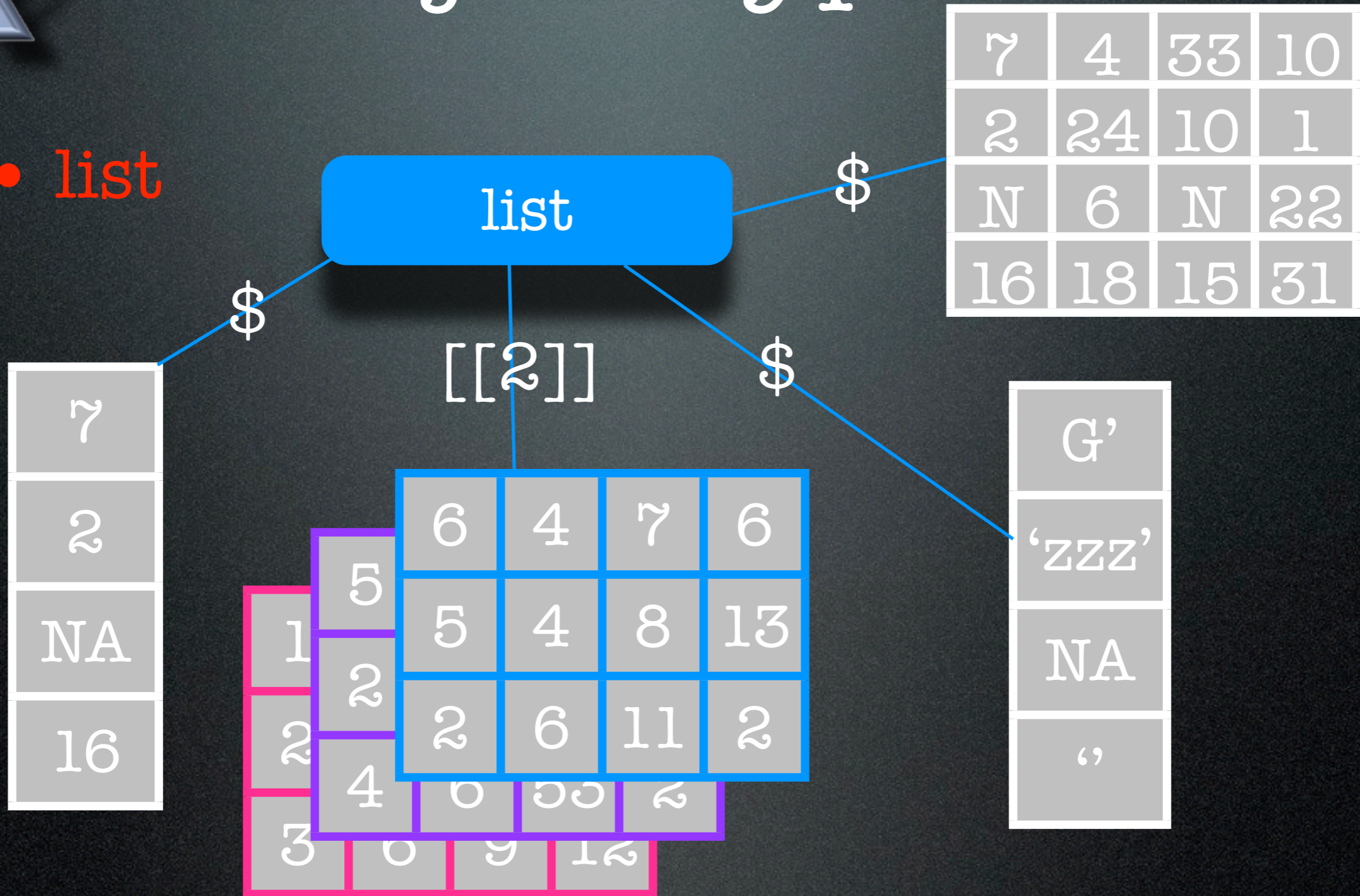
- array





# Object Types

- list



```
lapply(a, sum) = c(25, 206, NA, 199)
```



# Object Types

- data.frame (special kind of list)

data.frame

\$v1 [,2] \$sex \$bla

7	4	female	10
2	24	male	1
NA	6	female	22
16	18	female	31

a[2, 3]

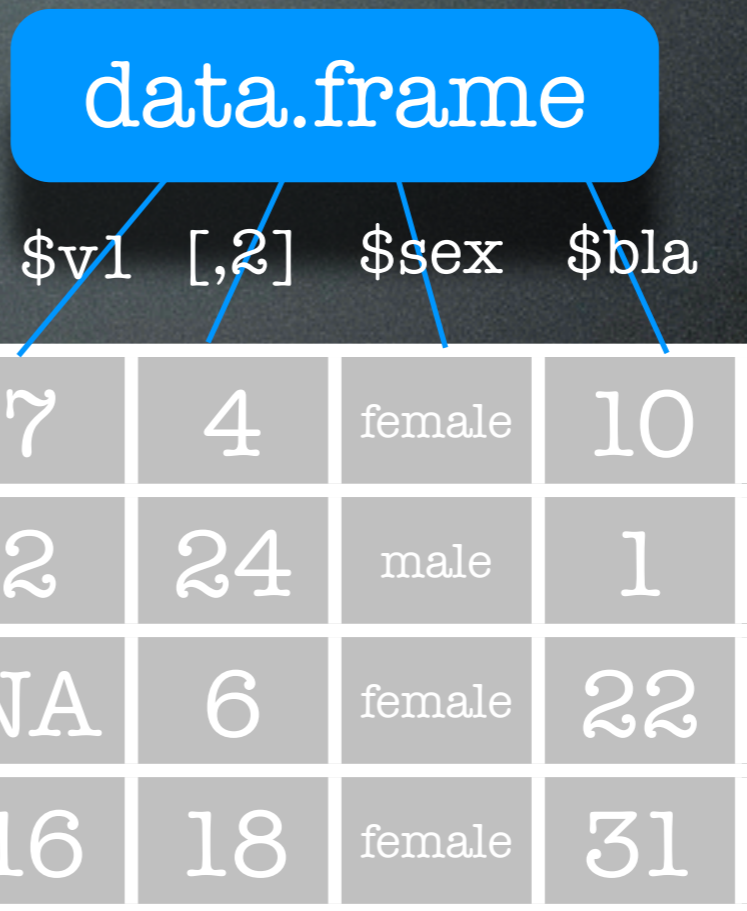
a\$bla[4]

```
apply(a, 2, sum) →  
lapply(a, sum) →  
tapply(a$v1, a$sex, sum) = c(7, 18)
```



# Object Types

- data.frame (special kind of list)



```
apply(a, 2, sum) → 25 52 NA 64  
lapply(a, sum) →  
tapply(a$v1, a$sex, sum) = c(7, 18)
```

# tapply

```
tapply(y, x, mean)
```

y	x
43	1
51	2
32	1
4	1
2	2
42	2
5	2
42	1

# tapply

```
tapply(y, x, mean)
```

```
mean(c(43, 32, 4, 42))
```

y	x
43	1
51	2
32	1
4	1
2	2
42	2
5	2
42	1

# tapply

```
tapply(y, x, mean)
```

```
mean(c(43, 32, 4, 42))
```

```
mean(c(51, 2, 42, 42))
```

y	x
43	1
51	2
32	1
4	1
2	2
42	2
5	2
42	1



# tapply

Not necessarily in  
one data.frame!

```
tapply(y, x, mean)
```

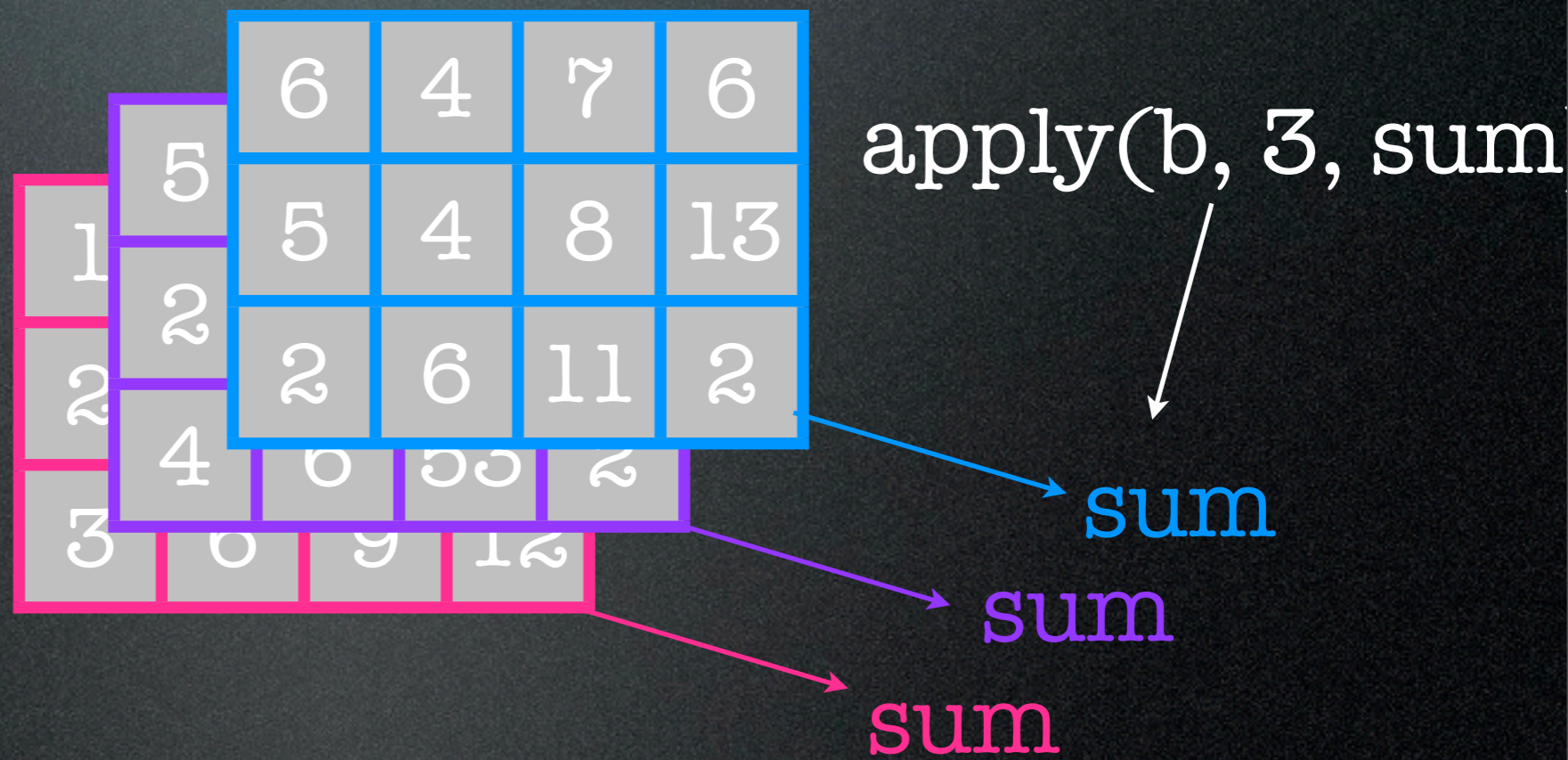
```
mean(c(43, 32, 4, 42))
```

```
mean(c(51, 2, 42, 42))
```

y	x
43	1
51	2
32	1
4	1
2	2
42	2
5	2
42	1



# apply





# apply

