

General R

R Basics

Data Classes

-
- `"hey" "I'm a string"`
- `TRUE FALSE` **not**

Data Types

- vector
- matrix
- data.frame
- array

nifti

Initializing: vectors

- `c()`

```
v = c(1, 4, 3, 7, 8)
print(v)
```

```
[1] 1 4 3 7 8
```

-

```
w = 1:5
print(w)
```

```
[1] 1 2 3 4 5
```

Assignment

R

=

<-

```
w = 1:5  
w <- 1:5
```

=

•

•

\$

-

• can

•

—

Help

• `help ?`

`help`

• `help c`

```
?c  
help(topic = "c")
```

• `help ?? help.search`

```
??c  
help.search(pattern = "c")
```

Some Details

- `y` `Y`
-
- `#`

`package::function()`

`::`

```
utils::help("c")
```


Initializing: matrices and arrays

- `m`
-

```
m = matrix(1:12, nrow = 3)
print(m)
```

```
      [,1] [,2] [,3] [,4]
[1,]    1    4    7   10
[2,]    2    5    8   11
[3,]    3    6    9   12
```

- `a`

```
a = array(1:36, dim = c(3, 4, 3))
```

- `dim()`

```
dim(a)
```

```
[1] 3 4 3
```

Subsetting: vectors

•

1

```
print(v)
```

```
[1] 1 4 3 7 8
```

```
print(v[4])
```

```
[1] 7
```

```
print(v[1:3])
```

```
[1] 1 4 3
```

```
print(v[c(1,3,5)])
```

```
[1] 1 3 8
```

Subsetting: matrices

- `[row, column]`

```
print(m[1,3])
```

```
[1] 7
```

```
print(m[1:2,3:4])
```

```
      [,1] [,2]  
[1,]    7  10  
[2,]    8  11
```

- `row column`

```
print(m[,4])
```

```
[1] 10 11 12
```

```
print(m[2,])
```

```
[1] 2 5 8 11
```

Subsetting: arrays

• `[x, y, z]`

```
print(a[1,1,1])
```

```
[1] 1
```

```
dim(a[,4,])
```

```
[1] 3 3
```

```
a[,4]
```

Operators in R: return numeric

- `+` `-` `*` `/` `^`
- `log` `abs` `sqrt`

```
print(v); print(w)
```

```
[1] 1 4 3 7 8
```

```
[1] 1 2 3 4 5
```

```
print(v + 4)
```

```
[1] 5 8 7 11 12
```

```
print(v + w)
```

```
[1] 2 6 6 11 13
```

```
print(sqrt(w^2))
```

```
[1] 1 2 3 4 5
```

Operators in R: return logical

- `>` `>=` `<` `<=` `==` `!=`
- `!` `&` `|`
- `all()` `TRUE` `any()`

```
print(!FALSE)
```

```
[1] TRUE
```

```
print(TRUE | FALSE)
```

```
[1] TRUE
```

```
print(FALSE & FALSE)
```

```
[1] FALSE
```

```
c(all(c(TRUE, FALSE)), any(c(TRUE, FALSE)))
```

```
[1] FALSE TRUE
```

Subsetting with logicals

which

TRUE

```
which(v > 5)
```

```
[1] 4 5
```

```
v[ which(v > 5) ]
```

```
[1] 7 8
```

```
v[ v > 5 ]
```

```
[1] 7 8
```

Website
