

# Un poco sobre funciones. . .

Jorge Loría

September 7, 2017

## Recapitulacion clase pasada

Vimos que existen 5 tipos básicos de estructuras:

- ▶ Vectores

# Recapitulacion clase pasada

Vimos que existen 5 tipos básicos de estructuras:

- ▶ Vectores
- ▶ Listas

# Recapitulacion clase pasada

Vimos que existen 5 tipos básicos de estructuras:

- ▶ Vectores
- ▶ Listas
- ▶ Matrices

# Recapitulacion clase pasada

Vimos que existen 5 tipos básicos de estructuras:

- ▶ Vectores
- ▶ Listas
- ▶ Matrices
- ▶ DataFrames

# Recapitulacion clase pasada

Vimos que existen 5 tipos básicos de estructuras:

- ▶ Vectores
- ▶ Listas
- ▶ Matrices
- ▶ DataFrames
- ▶ Arrays

# Principio

*Todo lo que existe es un objeto, y todo lo que sucede es una llamada a una función - John Chambers*

# Principio

*Todo lo que existe es un objeto, y todo lo que sucede es una llamada a una función - John Chambers*

```
f1 <- function(x) x^2
```



# Principio

*Todo lo que existe es un objeto, y todo lo que sucede es una llamada a una función - John Chambers*

```
f1 <- function(x) x^2
```

```
f1(10)
```

```
## [1] 100
```

## Partes de las funciones en R

- ▶ `body()`

## Partes de las funciones en R

► `body()`

```
body(f1)
```

```
## x^2
```

# Partes de las funciones en R

- ▶ `body()`

```
body(f1)
```

```
## x^2
```

- ▶ `formals()`

# Partes de las funciones en R

- ▶ `body()`

```
body(f1)
```

```
## x^2
```

- ▶ `formals()`

```
formals(f1)
```

```
## $x
```

La parte menos conocida

## La parte menos conocida

- ▶ `environment()`

## La parte menos conocida

► `environment()`

```
environment(f1)
```

```
## <environment: R_GlobalEnv>
```



## Las funciones también son un objeto

Por lo que puede ser parámetros de otras funciones, por ejemplo:

```
x <- list(1:10,32:22,-1:20)
sapply(x,sum)
```

```
## [1] 55 297 209
```

## Yo Dawg'

Entonces hay mucha flexibilidad con lo que se hace! Además, pueden definirse funciones dentro de otras funciones

```
f2 <- function(x){  
  f3 <- function(y){  
    x+y  
  }  
  f3(5)  
}
```

```
f2(2)
```

```
## [1] 7
```