

Programación para el Análisis de Datos

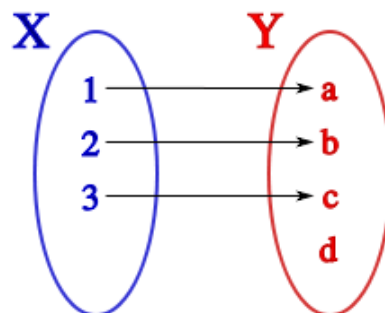
Funciones y estructuras de control 

Daniel Fraiman

Maestría en Ciencia de Datos, Universidad de San Andrés

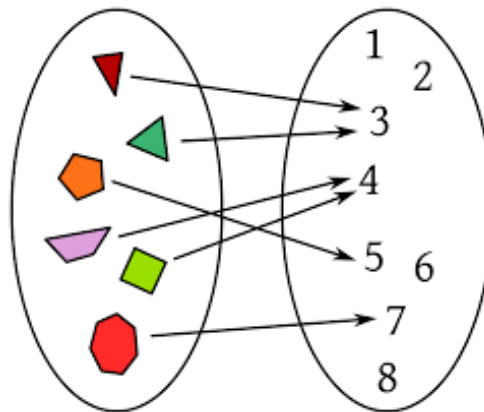
1 / 11

Funciones



2 / 11

Funciones



3 / 11

Funciones

Una función tiene un espacio de salida y un espacio de llegada

- Espacio de Salida: argumentos de la función.
- Espacio de Llegada: lo que devuelve la función.
- Estos espacios pueden ser
 - un número, un caracter
 - una matriz, una lista
 - un gráfico, una imagen, archivo de audio, archivo de texto
 - ...
- La “operación” en si podemos decir que es la función



- muchas funciones predeterminadas. `mean(x)`, `unique(x)`, etc
- podemos crear nuestras funciones.

4 / 11

Funciones

Sintaxis para crear funciones

```
f <- function(argumento)
{
  ...
  ...
  return(salida) # no tiene porque terminar en return
}
```

Ejemplo 1

```
f1 <- function(x)
{ media<-mean(x)
  varianza<-var(x)
  maximo<-max(x)
  return(c(media, varianza)) }
```

5/11

Funciones

Ejemplo 2

```
f2 <- function(x,y)
{ z <-x+y
  return(z)
}
```

Ejemplo 3

```
f3 <- function(x,y,titulo)
{ plot(x,y,main=titulo)
}
```

6/11

Funciones

Funciones dentro de funciones

Una función puede contener una o varias funciones. Esto hace los códigos más simples de leer.

Ejemplo 4

```
f4 <- function(x,y)
  { salida <- f2(x,y) # f2 tiene que estar definida para que
corra
    f3(x,salida) # f3 tiene que estar definida para que corra
  }
```

7 / 11

Estructuras de control

Sintaxis for

```
for (i in 1:10) # también for (auto in garage)
  { print(i) # print(auto)
  }
```

Sintaxis while

```
i<-1
while (i < 8)
  { print(i) ; i<- i+1
  }
```

8 / 11

Estructuras de control

Algunas condiciones lógicas

- `==`
- `<` , `>` , `>=` , `<=`
- `%in%` `# if(palabra %in% c("auto","casa","moto")){`
- `(condicion1) || (condicion2)` `# Union de las condiciones, se pueden poner muchas`
- `(condicion1) && (condicion2)` `# Intersección de las condiciones, se pueden poner muchas`
- `(condicion1 || condicion2) && (condicion3 || condicion4)` `# recordar conjuntos`

9/11

Estructuras de control

Sintaxis `if`

```
if(sum(x)>2)
{ print("suma mayor a 2")
}
```

10/11

Estructuras de control

Sintaxis if else

```
if(condicion)
    { expresion1
    } else { # else va en el mismo renglón del cierre de
paréntesis
    expresion2
    }
```

Sintaxis ifelse

```
ifelse(condicion, expresion1,expresion2)
```