

**UNIVERSIDAD DE EL SALVADOR
FACULTAD MULTIDISCIPLINARIA DE OCCIDENTE
DEPARTAMENTO DE MATEMÁTICA**



Licenciatura en Estadística

Control Estadístico del Paquete R

”UNIDAD UNO”

**Alumna:
Erika Beatríz Guillén Pineda**

**Fecha de elaboración
Santa Ana - 27 de noviembre de 2015**

1. ESTRUCTURA CONDICIONAL: LA ORDEN IF() Y IFELSE()

- Por ejemplo, ejecute las siguientes instrucciones

```
x <- c(6:-4);  
x  
  
## [1] 6 5 4 3 2 1 0 -1 -2 -3 -4  
  
sqrt(x) # Produce un mensaje de advertencia  
  
## Warning in sqrt(x): Se han producido NaNs  
  
## [1] 2.449490 2.236068 2.000000 1.732051 1.414214 1.000000 0.000000  
## [8] NaN NaN NaN NaN  
  
sqrt(ifelse(x >= 0, x, NA)) # No produce advertencia  
  
## [1] 2.449490 2.236068 2.000000 1.732051 1.414214 1.000000 0.000000  
## [8] NA NA NA NA  
  
ifelse(x >= 0, sqrt(x), NA) # Produce un mensaje de advertencia  
  
## Warning in sqrt(x): Se han producido NaNs  
  
## [1] 2.449490 2.236068 2.000000 1.732051 1.414214 1.000000 0.000000  
## [8] NA NA NA NA  
  
# Comente las diferencias entre cada una de las instrucciones anteriores.
```

2. ESTRUCTURAS ITERATIVAS O DE REPETICIÓN: FOR(), WHILE() Y REPEAT()

- Ejemplo:

```
x <- c(2, 6, 4, 7, 5, 1)
suma<-0; for(i in 1:3) suma = suma+x[i]; suma

## [1] 12
```

3. FUNCIONES ESCRITAS POR EL USUARIO

- Ejemplo 1: Definir en R la función cuadrática $y = f(x) = (3x^2) - (5x) + 2$

```
func.cuadratica <- function(x)
{
  3*x^2-5*x+2
}
y <- func.cuadratica(2);y

## [1] 4
```

NOTA: Toda función para usarla debe estar cargada en el área de trabajo (Workspace). Es decir, primero es necesario correr el código necesario el código de la función y asegurarse que no contenga errores de sintaxis.

- Ejemplo 2: Se quiere definir una función para calcular la media de un vector de datos

Una definición podría ser:

```
media <- function(x)
{
  n = length(x)
  suma <- 0.0
  for(i in 1:n) suma = suma + x[i]
  media = suma/n
}
save(media, file= "media.RData")
rm(list=ls(all=TRUE))
load("media.RData")

x <- 1:5;
(media(x)) # Se usa doble par\ 'entesis para que muestre el resultado en pantalla
```

```
## [1] 3

y <- c(5, NA , 4, 9);
(media(y)) # El resultado no puede calcularse pues falta un dato

## [1] NA

z <- c(5, 1 , 4, 9);
(media(z))

## [1] 4.75

(media) # Nos muestra el código de la función

## function(x)
## {
##
##   n = length(x)
##   suma <- 0.0
##   for(i in 1:n) suma = suma + x[i]
##   media = suma/n
##
## }
```

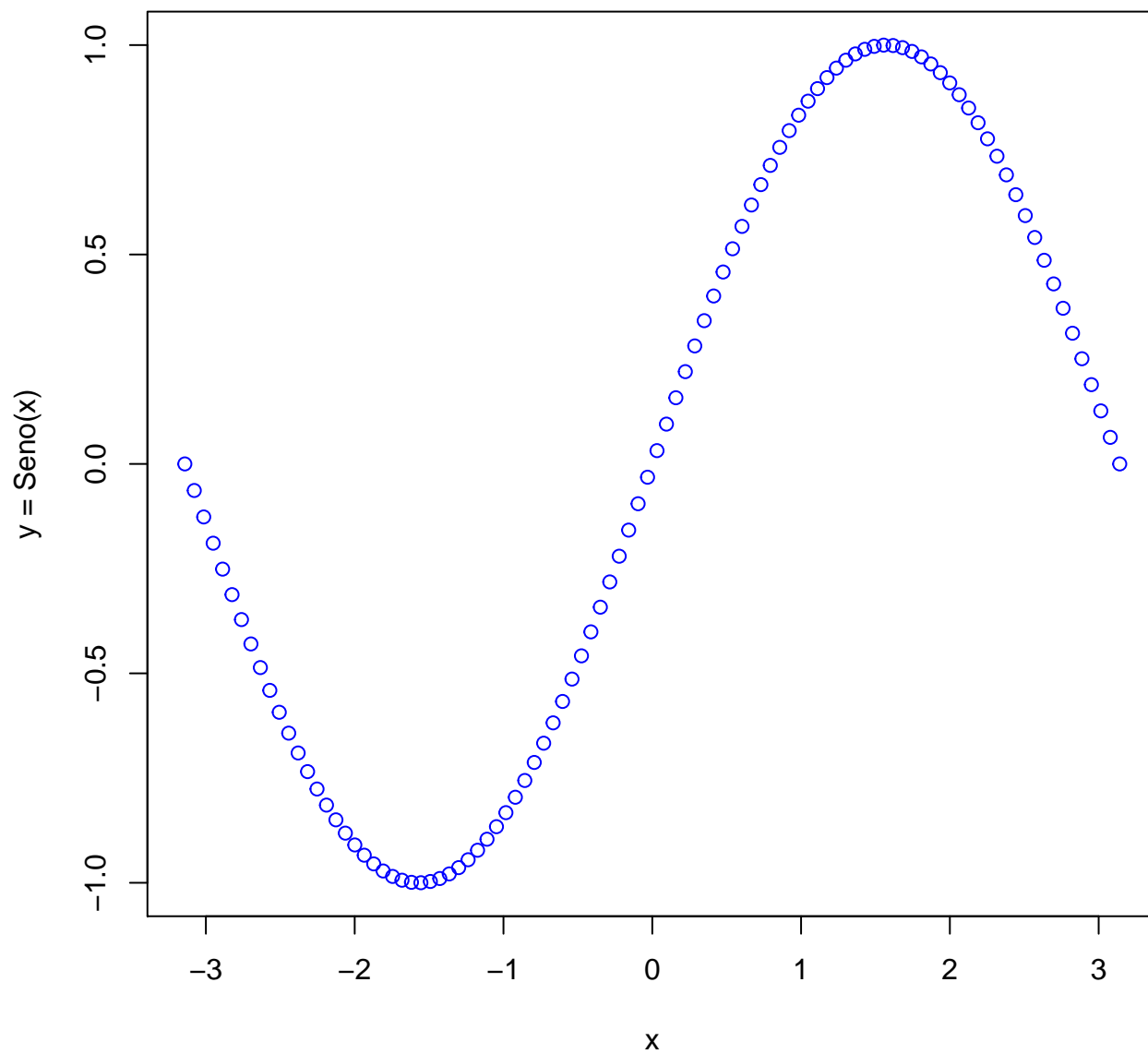
- Ejemplo 3: Se quiere definir una función para graficar la función seno de x

Una definición de esta función puede ser:

```
Seno <- function(x)
{
  y = sin(x)
  plot(x, y, main="Ejemplo de gráficos en R",
  xlab="x", ylab="y = Seno(x)", col="blue", pch=1)
}
# Pruebe la función con el siguiente vector:

x<-seq(-pi, pi, len=100)

Seno(x)
```

Ejemplo de gr'aficos en R

4. EJERCICIOS PROPUESTOS

- Ejercicio 1: Escriba una función para encontrar el factorial de un número mayor que cero

```

fac<-function(f){ prod<-1 # Inicializar el producto en 1
if (f==0){ # Cuando el valor ingresado es cero
  prod<-1 # El factorial es 1
  return(prod)
}
else{
  if(f<0) # Cuando el valor ingresado es negativo
    print("No existe el factorial de un n\ 'umero negativo")
  else {
    int<-c(1:f) # Cuando el valor es positivo

    for(i in int)
      prod<-prod * i
    return(prod)
  }
}
}
fac(4)

## [1] 24

```

- Ejercicio 2: Escriba una función para encontrar la varianza o la cuasi-varianza de un vector de datos.

```

vx<-function(k) { suma <- 0.0
  z<-length(k)
  for(i in 1:z){
    suma = suma + k[i]
    media = suma/z # Obtener la media aritmetica del vector
    for(i in 1:z){
      vx<-k[i]-media
      vx<-(vx)/z
    }
  }
  return(vx)
}
k <- c(2,3,4)
(vx(k))

## [1] 0.3333333

```

- Ejercicio 3: Escriba una función para encontrar la media geométrica de un vector de datos

```
# Obtener la raíz n-esima de cualquier valor
raiz=function(m,n){ # Este es la función que llamamos en los dos códigos siguientes
  raiz=n^(1/m)
  return(raiz)
}
raiz(3,27)

## [1] 3

# Caso cuando el vector esta ordenado, es decir desde 1 hasta el valor deseado
MG<-function(m)
{prod<-1 # Este código se parece al del factorial de un numero positivo
int<-c(1:m)

for(i in int)
  prod<-prod * i
raiz=raiz(m,prod)
# Obtener la raíz n-esima (corresponde a la cantidad de valores en el vector)
return(raiz)
}
MG(5)

## [1] 2.605171

# Caso cuando el vector es de cualquier forma que el usuario desee
MG2<-function(g) {product<-1 # Inicializar el producto
p<- length(g) # Guardar la longitud del vector
for(i in 1:p)
  product<-product * g[i] # Realizar el producto de cada valor del vector
MG2<-product # Guardar el producto en MG2
raiz<-raiz(length(g),MG2)
# Obtener la raíz n-esima (corresponde a la cantidad de valores en el vector)
return(raiz)
}
g<-c(2,3,4,5) # Pasar el vector de datos
(MG2(g))

## [1] 3.309751
```

- Ejercicio 4: Escriba una función para encontrar la media armónica de un vector de datos

```
MA <- function(x)
{
  suma <- 0.0 # Inicializar suma
  n <- length(x) # Guardar la longitud del vector
  for(i in 1:n)
    suma <- suma + (1/x[i])
  # Realizar la suma, pero de los reciprocos (1/x[i]) de los valores del vector
  denom <- suma/n # Encontrar la media de los reciprocos
  MA<-1/denom
}
x <- c(2,3)
# Pasar el vector de datos
(MA(x))

## [1] 2.4
```