

UNIVERSIDAD DE EL SALVADOR
0.50 cm] FACULTAD MULTIDISCIPLINARIA DE OCCIDENTE
0.50 cm] DEPARTAMENTO DE MATEMÁTICA



Licenciatura en Estadística

Control Estadístico del Paquete R

”UNIDAD UNO”

Alumna:
Erika Beatriz Guillén Pineda

Fecha de elaboración
Santa Ana - 27 de noviembre de 2015

1. 1. FACTORES NOMINALES Y ORDINALES.

1.1. FACTORES NOMINALES

Ejemplo 1: Variables sexo (categórica) y edad en una muestra de 7 alumnos del curso

```
# Supongamos que se obtuvieron los siguientes datos:

sexo <- c("M", "F", "F", "M", "F", "F", "M");
sexo

## [1] "M" "F" "F" "M" "F" "F" "M"

edad <- c(19, 20, 19, 22, 20, 21, 19);
edad

## [1] 19 20 19 22 20 21 19

# Podemos construir un factor con los niveles o categorías de sexo

FactorSexo = factor(sexo);
FactorSexo

## [1] M F F M F F M
## Levels: F M

# Se pueden ver los niveles o categorías del factor con: levels(FactorSexo)
# Crear una tabla que contenga la media muestral por categoría de
# sexo (nivel del factor):

mediaEdad <- tapply(edad, FactorSexo, mean);
mediaEdad

## F M
## 20 20

# Note que el primer argumento debe ser un vector, que es del cual se
# encontrarán las medidas de resumen; el segundo es el factor que se está
# considerando, mientras que en el tercero se especifica la medida de interés,
# solamente puede hacerse una medida a la vez.
# ¿Qué tipo de objeto es la variable mediaEdad?: is.vector(mediaEdad);
# is.matrix(mediaEdad); is.list(mediaEdad); is.table(mediaEdad); is.array(mediaEdad)

is.vector(mediaEdad);

## [1] FALSE

is.matrix(mediaEdad);
```

```
## [1] FALSE

is.list(mediaEdad);

## [1] FALSE

is.table(mediaEdad);

## [1] FALSE

is.array(mediaEdad)

## [1] TRUE
```

Vemos que el tipo de objeto de la variable mediaEdad es un ARRAY

1.2. FACTORES ORDINALES

La función `ordered()` crea este tipo de factores y su uso es idéntico al de la función `"factor()"`. Los factores creados por la función `"factor()"` los denominaremos nominales o simplemente factores cuando no haya lugar a confusión, y los creados por la función `ordered()` los denominaremos ordinales. En la mayoría de los casos la única diferencia entre ambos tipos de factores consiste en que los ordinales se imprimen indicando el orden de los niveles. Sin embargo, los contrastes generados por los dos tipos de factores al ajustar Modelos lineales, son diferentes.

2. CREACIÓN Y MANEJO DE LISTAS

Ejemplo 1: Crear una Lista con cuatro componentes.

```
lista1<-list(padre="Pedro", madre="Mar?a", no.hijos=3, edad.hijos=c(4,7,9))
lista1

## $padre
## [1] "Pedro"
##
## $madre
## [1] "Mar?a"
##
## $no.hijos
## [1] 3
##
## $edad.hijos
## [1] 4 7 9

is.matrix(lista1);

## [1] FALSE
```

```
is.vector(lista1$edad.hijos)
```

```
## [1] TRUE
```

Ejemplo 2: Acceso a las componentes de una lista:

```
lista1[1] # Accede a la componente como una lista (con etiqueta y valor)
```

```
## $padre
```

```
## [1] "Pedro"
```

```
lista1["padre"] # El acceso es igual que con lista1[1]
```

```
## $padre
```

```
## [1] "Pedro"
```

```
lista1[[2]]
```

```
## [1] "Mar?a"
```

```
# Accede al valor o valores de la componente segunda pero  
# no muestra el nombre de la componente.
```

```
lista1["madre"] # El acceso es igual que con lista1[[1]]
```

```
## $madre
```

```
## [1] "Mar?a"
```

Ejemplo 3: Acceso a los elementos de la cuarta componente:

```
lista1[[4]][2]
```

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

```
# Se indica el elemento a ingresar en el segundo corchete
```

Ejemplo 4: Acceso de las componentes de una lista por su nombre:

```
lista1["padre"]
```

```
## $padre
```

```
## [1] "Pedro"
```

Forma general: Nombre_{de lista}nombre_{de componente}

Por ejemplo :

```
lista1$padre # equivale a  
## [1] "Pedro"  
  
lista1[[1]];  
## [1] "Pedro"  
  
# y  
lista1$edad.hijos[2] # equivale a  
  
## [1] 7  
  
lista1[[4]][2]  
## [1] 7
```