

If, else, else if

Laboratorio de Procesamiento de Información Meteorológica y
Oceanográfica DCAO - FCEN - UBA



August 31, 2023



Una de las herramientas más utilizadas en un código con las condiciones. Evaluando si se cumplen las condiciones requeridas para un proceso se puede ordenar el flujo de un código.

En R se usa la sentencia condicional IF (si) cuando deseamos que una operación se ejecute únicamente cuando una condición se cumple.

Entonces en R, y en la mayoría de los lenguajes de programación, le tenemos que indicar a la computadora que:

SI la condición es verdadera, ENTONCES ejecuta las operaciones:

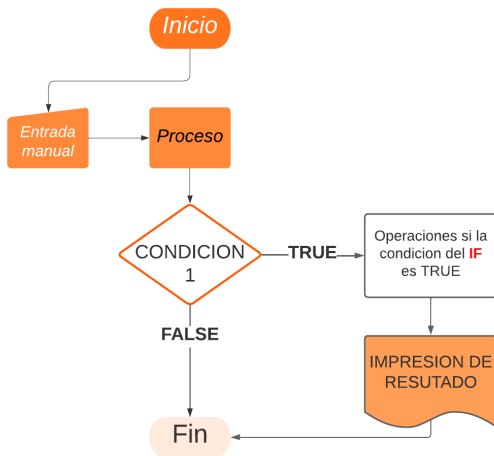


Figure: Diagrama de flujo IF



Repaso: variables lógicas

Las variables lógicas son aquellas que solo pueden tener dos valores (cierto o falso) estos representan el resultado de evaluar si se cumple una condición o de una operación lógica:

Ejemplos condición:

- $a > b$
- $a = b$

Ejemplos de operaciones lógicas:

- $a > b \wedge b > 0$ \wedge es "y"
- $a = b \vee b > 0$



La sintaxis para escribir condicionales en R es la siguiente:

```
if(condicion) {  
  # debe devolver TRUE o FALSE  
  operaciones_si_la_condicion_es_TRUE  
}
```



Si la condición se cumple, es decir, si es verdadera (TRUE), entonces se realizan las operaciones. En caso contrario, no ocurre nada y el código con las operaciones no es ejecutado.



Ejemplo IF

Si la sentencia se cumple:

$x \leftarrow 4$

```
if (x > 0) {  
print ("Numero positivo")  
}
```

```
## [1] "Numero positivo"  
# Se cumple la condicion ya que x es 4 ( mayor a 0)  
# y se muestra "Numero positivo"
```




Ejemplo IF

Si la sentencia no se cumple:

```
x ← -5
```

```
if (x > 0) {  
  print("Numero_positivo")  
}
```

```
# La condicion no se cumple,  
# por eso no se ejecutan las sentencias
```



La sentencia ELSE complementa a la sentencia IF, ya que permite escribir acciones a ejecutar SI la condición no se cumple. Es decir, podemos dar instrucciones tanto para un valor TRUE de la condición como para un valor FALSE de la condición.

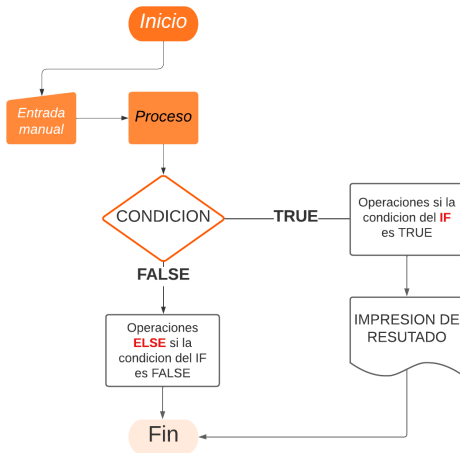


Figure: Diagrama de flujo con sentencia ELSE



Sintaxis IF, ELSE en R

La sintaxis es similar a la sintaxis cuando sólo usamos IF, sólo agregamos instrucciones adicionales para cuando la condición da FALSE como resultado. OJO, 'else' debe estar escrito en la misma línea que el cierre de la sentencia IF.

```
if(condicion) {  
  operaciones_si_la_condicion_es_TRUE  
} else {  
  operaciones_si_la_condicion_es_FALSE  
}
```



Ejemplo IF, ELSE en R

```
x <- -5
if (x > 0) {
  print("Numero_positivo")
} else{
  print("Numero_negativo_o_cero")
}
```



Calculemos la raíz cuadrada de un numero cualquiera. Para esto le vamos a pedir al usuario que ingrese por pantalla un numero.

Este numero puede ser positivo, negativo o cero. Si es positivo o igual a cero podemos calcular la raíz sin problema Pero si es negativo, R no podra calcular la raíz cuadrada, entonces podemos imprimir un mensaje que pida al usuario que ingrese un número válido.



Una solución:

```
if (valor >= 0) {  
  raiz_cuadrada <- sqrt(valor)  
} else {  
  print("Ingrese un numero mayor o igual a 0")  
}
```



Operaciones lógicas en el condicional

Como repasamos antes, una variable lógica puede ser el resultado de una condición o de una operación lógica entre variables lógicas. Veamos un ejemplo simple para ver la sintaxis.

Con la operación Y (&):

```
x <- 1
y <- 2
if ((x > 0) & (y > 0)) {
  print("Ambos datos son positivos")
}
## [1] "Ambos datos son positivos"
```

```
x<-1y<-2if((x>0)&(y>0)){ print("Ambos dat
```




Operaciones lógicas en el condicional

Con la operación O (|):

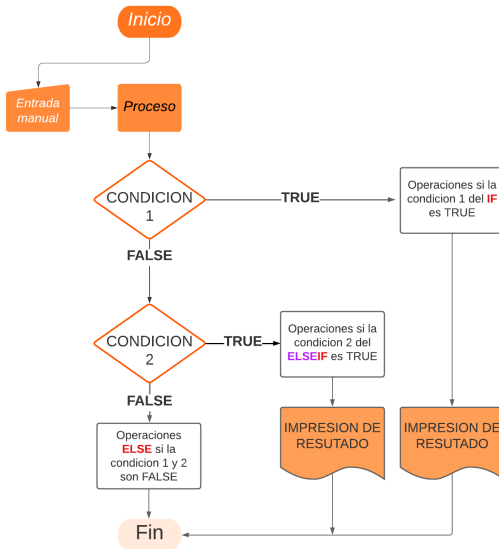
```
x <- 1
y <- -2
if ((x > 0) | (y > 0)) {
  print(" Algun_dato_es_positivo")
}
## [1] "Algun dato es positivo"
```



Quiero re-truco. Podemos tener más de una condición para probar. En este caso podemos usar else if, lo cual permite agregar tantas condiciones como sea necesario. La sintaxis es la siguiente:

```
if (condicion1) {  
operaciones_si_la_condicion1_es_TRUE  
} else if (condicion2) {  
operaciones_si_condicion1_es_F_y_condicion2_es_T  
} else {  
operaciones_si_condicion1_es_F_y_condicion2_es_F  
}
```

ELSE IF



Ejemplo ELSE IF



Vamos a hacer un código para evaluar si, dada una nota, un parcial está aprobado o desaprobado. Consideremos que se aprueba con 4. Puede suceder que el número ingresado no sea una nota entre 0 y 10, por lo tanto, el código tiene que ser capaz de dar una respuesta en esta situación.



Ejemplo ELSE IF

Una solución:

```
nota <- as.numeric(readline("Ingrese su nota: "))  
if (nota >=4 & nota <=10){  
  print("Se encuentra aprobado")  
} else if (nota < 4 & nota >0){  
  print("Se encuentra desaprobado")  
} else {  
  print("Ingrese una nota valida")  
}
```

Crear un programa que le pida a un usuario el ingreso por consola de:

- Tmax: temperatura máxima registrada (en $^{\circ}\text{C}$)
- Tmin: temperatura mínima registrada (en $^{\circ}\text{C}$)

Si la Tmax ingresada es menor a la Tmin, el código debe imprimir el mensaje : 'Tmax menor a Tmin, verificar los datos'. Si en cambio, la Tmax es mayor a la Tmin, el código debe calcular la amplitud térmica diaria (la amplitud es la resta entre la temperatura máxima y mínima) y luego imprimir un cartel: 'La amplitud diaria fue de 'VALOR AMPLITUD' $^{\circ}\text{C}$ '. Si las temperaturas ingresadas tuvieron el mismo valor, el programa debe imprimir el mensaje: 'Tmax igual a Tmin, verificar datos'