

Sumatorios y productorios en R

En esta entrada veremos algunas aplicaciones de los **bucles while y for**: los sumatorios y productorios. Ambas operaciones, aunque son típicas de los ejercicios de algoritmia, son frecuentes en los ejercicios de R realizados en clase, por lo que es útil repasar cómo se haría el algoritmo en el programa.

https://www.youtube.com/watch?v=9zJqvNky_ME&t=2s

Sumatorios:

Los sumatorios (Σ) se tratan de un tipo de operación que consiste en la suma de todos los elementos de una serie de datos. Podremos hacer esta operación tanto con **vectores** como con **matrices**.

En el caso de los vectores, seguimos los siguientes pasos:

- Definimos el vector con las componentes que queramos sumar y definimos también nuestro **vector sumatorio, igualándolo a 0**.
- Abrimos un único bucle for, que varía desde 1 hasta la longitud del vector cuyas componentes queremos sumar.
- Dentro del bucle, introducimos la operación “ Suma = Suma + Vector[i] ”. Así, crearemos un bucle en el que cada vez, al valor total se le irá sumando la componente [i] del vector.
- Cerramos el bucle (**error frecuente**, cuidado con las llaves).

Así, obtendremos algo como:

```
> Variables=c(1,3,5,6,1,6,4,1,8,9)
> Suma=0
> for (i in 1:length(Variables)){
+ Suma=Suma+Variables[i]
+ }
> Suma
[1] 44
> |
```

Si lo que queremos es sumar todos los elementos de una matriz, los pasos son los mismos, con la excepción de que debemos abrir **2 bucles for anillados**, uno en [i] para las filas y otro en [j] para las columnas. Además, deberemos especificar que el elemento de la matriz es el que se encuentra en las posiciones [i, j]. Así obtendremos:

```

> VectorM1=c(7,9,12)
> VectorM2=c(7,3,6)
> VectorM3=c(2,6,2)
> Matriz= cbind(VectorM1, VectorM2, VectorM3)
> SumaMatriz=0
> for (i in 1:3){
+ for (j in 1:length(VectorM1)){
+
+ SumaMatriz=SumaMatriz + Matriz[i,j]
+
+ }
+ }
> SumaMatriz
VectorM1
      54
> |

```

Productorio:

Los productorios (\prod) son operaciones muy similares a los sumatorios, con la diferencia de que en lugar de sumar los elementos, los multiplican.

Dada esta similitud, lo único que tendremos que cambiar a la hora de programar es iniciar el **vector del resultado a 1**, ya que si no nos daría 0. Además, deberemos cambiar la operación a una multiplicación (*).

En el caso de los vectores obtendremos:

```

> Variables=c(1,3,5,6,1,6,4,1,8,9)
> Multiplicacion=1
>
> for (i in 1:length(Variables)){
+ Multiplicacion=Multiplicacion*Variables[i]
+ }
> Multiplicacion
[1] 155520
> |

```

Y en el de las matrices:

```

> VectorM1=c(7,9,12)
> VectorM2=c(7,3,6)
> VectorM3=c(2,6,2)
> Matriz= cbind(VectorM1, VectorM2, VectorM3)
> MultiplicacionMatriz=1
> for (i in 1:3){
+ for (j in 1:length(VectorM1)){
+
+ MultiplicacionMatriz=MultiplicacionMatriz * Matriz[i,j]
+
+ }
+ }
> MultiplicacionMatriz
VectorM1
2286144

```

Por último os recomendamos algunos **ejercicios** que incluyen sumatorios y productorios para practicar.