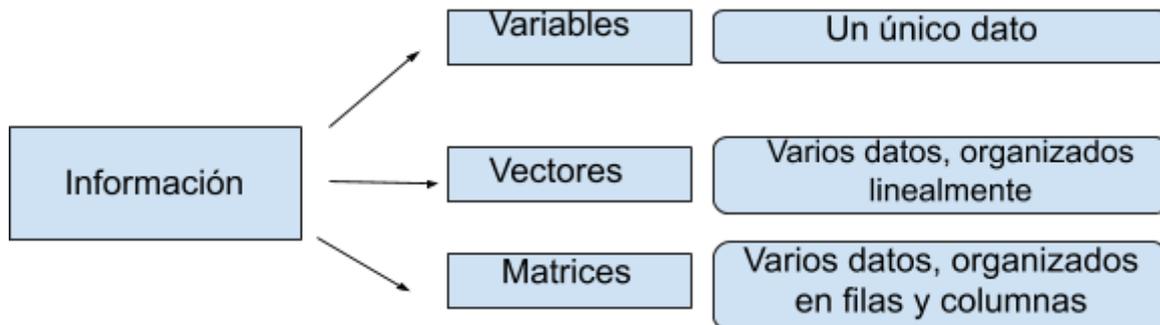


Almacenamiento de información: variables, vectores y matrices

En este apartado veremos las distintas herramientas que podemos emplear en el programa R para almacenar datos e información. Nos centraremos principalmente en variables, vectores y matrices, ya que son usados en la mayoría de aplicaciones del programa.

Como se ha mencionado, las principales formas de almacenamiento son:



Los 3 pueden almacenar tanto datos numéricos como escritos, pero si queremos almacenar los segundos se deberá escribir el dato entre comillas (' _ ' o " _ " indistintamente, siempre que las de apertura y cierre sean las mismas).

Variables: almacenan un único dato. Para asignarlas simplemente elegiremos un nombre para nuestra variable y mediante = o <- le asignamos un valor.

```
> a<-4
> a
[1] 4
>
> b='Olivia'
> b
[1] "Olivia"
> |
```

Debido a que almacena sólo un dato, en el momento en el que se asigne otro valor a la misma variable, perderá el asignado anteriormente.

Vectores: almacenan infinitos valores, organizados de manera lineal. Para construir un vector haremos NombreVector<- o = c(valor1, valor2, valor3,...., valorN). Es importante no olvidarse de la c anterior al paréntesis, ya que si no, R no lo tomará como un vector sino como una variable, y al tener varios componentes provocaremos un error.

```
> Meses<-c("enero","febrero","marzo","abril")
> Meses
[1] "enero" "febrero" "marzo" "abril"
>
> Produccion<-c(12,15,73,25)
> Produccion
[1] 12 15 73 25
>
> VectorEjemplo<-c("B",15,"nuez", 34, 66)
> VectorEjemplo
[1] "B" "15" "nuez" "34" "66"
> |
```

Nota: se pueden mezclar números y variables escritas a la hora de hacer un vector, pero R lo tomará todo como escritos y no como números + escritos.

Matrices: las matrices almacenan infinitos datos, organizados en forma de columnas y filas. A la hora de construir una matriz, hay distintos métodos:

En primer lugar nos encontramos con el comando "**Matrix**". Con él, a partir de un vector podremos generar una matriz de las dimensiones que queramos, siempre que el vector tenga los componentes suficientes.

Para ello, introduciremos NombreMatriz<-matrix(vector que queramos incluir, nrow=n, ncol=m). Siendo nrow el número de filas y ncol el número de columnas.

Es decir, necesitamos un vector que tenga las componentes necesarias para rellenar las n filas (un múltiplo de las dimensiones de la matriz, si tenemos una matriz de 3x3, un múltiplo de 3).

```
> a<-c(5,3,7,8,2,6)
> M<-matrix(a, nrow=3, ncol=3)
> M
      [,1] [,2] [,3]
[1,]    5    8    5
[2,]    3    2    3
[3,]    7    6    7
> |
```

Nota: si nuestro vector no tiene las suficientes componentes para llenar todos los elementos de la matriz, se repetirán los primeros (ver ejemplo).

Por otro lado tenemos los comandos **rbind** y **cbind**, los cuales nos permiten crear una matriz a partir de vectores dispuestos en filas y columnas respectivamente. Para ello, necesitaremos definir previamente tantos vectores como filas o columnas queramos que tenga nuestra matriz.

```
> A<-c(9,7,2,4,2)
> B<-c(6,2,7,3,6)
> C<-c(1,5,6,0,3)
> M1<-cbind(A,B,C)
> M1
      A B C
[1,] 9 6 1
[2,] 7 2 5
[3,] 2 7 6
[4,] 4 3 0
[5,] 2 6 3
> M2<-rbind(A,B,C)
> M2
      [,1] [,2] [,3] [,4] [,5]
A      9    7    2    4    2
B      6    2    7    3    6
C      1    5    6    0    3
> |
```

Nota: si los vectores con los que construimos la matriz no tienen las mismas componentes, se repetirán los primeros valores del vector con menos componentes.

Todas estas herramientas son fundamentales para realizar las [operaciones básicas de R](#), o para funciones más complejas como la creación de [Tablas data.frame](#) o [funciones](#).

Si es tu primera vez usando R o aún no tienes el programa instalado, recomendamos ver [cómo instalar R](#) e [introducción a R](#). Recomendamos también repasar los [errores más frecuentes](#) usando estos comandos y realizar los [ejercicios propuestos](#).