

CHULETARIO PRÁCTICA 1

1. Asignar valores o nombres:

> **P<-6+3** #o expresión equivalente > **P=6+3**

> **P** #o expresión equivalente > **print(P)**

[1] 9

> Aguacate<-9/3 #Cuidado con mayúsculas y minúsculas

> name1= "Membrillo"

> name2= "Chocolate"

1. Comandos básicos:

> rnorm(1)	Genera un dato al azar muestreado de una distribución normal de media 0 y varianza 1.
> 21*2.5-rnorm(1) [1] 52.64696	
>runif(N)	Obtener N números aleatorios en el intervalo [0, 1].
> 100*runif(10)	10 valores aleatorios en el intervalo [0,100].
>sample(a:b,M)	Obtener M números enteros aleatorios en el intervalo [a, b] con o sin repetición.
> sample(a:b,M,replace=TRUE)	Con valores repetidos
> sample(a:b,M,replace=FALSE)	Sin valores repetidos
>ls()	Listar todo lo que hay en el directorio de trabajo.
> ls.str()	Obtener información sobre los objetos que hay en la memoria.
> ls(pat="m")	Listar todos los objetos cuyo nombre contenga un carácter en partículas, en este caso la letra "m".
> ls(pat = "^n")	Listar objetos cuyo nombre comienza con un carácter en particular, en este caso que comiencen por "n".
rm() o rm(list=ls(all='TRUE'))	Borrar elementos de la memoria.
rm(name1)	Borrar un elemento concreto.

2. Operaciones:

Suma +
Resta -
Multiplicación *
División /
Elevar a una potencia ** o ^

- Constantes numéricas

pi = número pi
exp(1) = número e
5e4 = 5×10^4

- Operadores:

< Menor
> Mayor
<= Menor o igual
>= Mayor o igual
!= Distinto
== Igualdad lógica

- Funciones

Funciones logarítmicas

log(x) logaritmo neperiano.
log10(x) logaritmo en base 10.
log2(x) logaritmo en base 2.
logb(x,base) logaritmo en cualquier base.
exp(x) función exponencial.

Funciones trigonométricas

sin(x) seno.
cos(x) coseno.
tan(x) tangente trigonométrica.
asin(x) arco seno.
acos(x) arco coseno.
atan(x) arco tangente.

Otras funciones

abs(x) valor absoluto.
sqrt(x) raíz cuadrada.
factorial(x) factorial.
choose(n,x) binomio de Newton n sobre x .

3. Vectores y matrices:

Vectores

Nombre vector <-c(____,____,.....)
> z1<-c(10, -3.5, "pepito") #El vector de 3 componentes es z1.
> z1[3] #Para acceder a un componente del vector, en este caso, la tercera componente.

```
> y1<-c(10, -3.5, 8)
> Tutu<-y1[1]+y1[2] #Sumar las componentes 1 y 2 y almacenarlo en Tutu.
```

- **Operar con vectores:**

```
> vector1<-c(17/3,4,exp(pi))
> vector2<-c(sin(4*pi/3),-36,5^3)
> vector1+vector2
> vector2-vector1
> vector2*vector1 #Se multiplica componente 1 * componente 1 del otro vector.
```

Para hacer el **producto escalar:**

```
> vector1%*%vector2
```

Matrices:

```
>Nombre de la matriz<-matrix(c(____,____,____), nrow=____,ncol=____)
```

Donde nrow es el número de filas y ncol es el número de columnas.

```
> A<-matrix(c(3,4,5,6), nrow = 2, ncol = 2)
```

```
      [,1] [,2]
[1,]    3    5
[2,]    4    6
```

```
>A[2,1] Para acceder a una posición
[1] 4
```