

Creación de vectores (secuencias de números aleatorios)

Una de las capacidades más importantes de R es la posibilidad de creación de vectores a partir de secuencias de **números aleatorios**. Cabe señalar que, en realidad, R no genera números puramente aleatorios, sino pseudoaleatorios. Esto significa que el algoritmo que genera los números necesita una semilla (función `set.seed()`) para inicializarse. Si conoces la semilla, podemos reproducir exactamente la secuencia de números en diferentes ordenadores, cosa que es imposible sin la semilla (proporcionará diferentes resultados).

```
set.seed(123)
```

Fundamentalmente, se utilizan dos funciones para la generación de estos números:

- La función `runif()`: genera números racionales (con decimales).

```
help(runif)
runif(10,                # Cantidad de número racionales
      3,                 # Valor mínimo
      6)                 # Valor máximo
```

- La función `sample()` genera **números enteros** (sin decimales)

```
help(sample)
sample(1:30,             # Intervalo de valores (entre 1 y 30 i
      nclusive)         # Cantidad de números en el intervalo
      size = 10,        # anterior
      replace = FALSE)  # TRUE permite que se repitan los números
```

La función `sample()` también se puede utilizar con **cadenas de caracteres** o con **fechas**.

```
sample(c("María", "Pedro", "Jaime"), # Caracteres a reproducir
      size = 10,                     # Número de caracteres
      replace = TRUE,                 # Se pueden repetir
      prob=c(0.5,0.3,0.2))           # Probabilidades de aparición
n
```

La función `rnorm()` crea una **población teórica con distribución normal** (con una media y una desviación estándar concretas) .

```
rnorm(10,                # Número de valores
      100,               # Media
      20)                # Desviación típica
```

También podemos crear vectores a partir de **progresiones aritméticas**.

```
help(seq)
```

```
# Secuencia ascendente empezando en 2, acabando en 50 y separados por 1,5
seq(2,
    50,
    by=1.5)
# Secuencia descendente empezando por 80, acabando por 4 y separados por 3,5
seq(80,
    4,
    by=-3.5)
# Error: falta el signo (negativo) de la progresión
seq(80,
    4,
    by=3.5)
# Progresión aritmética de 2 a 10 con 10 miembros
seq(2, 10,
    length.out=10)
# Progresión aritmética desde 2, con 10 miembros cada 0.5
seq(2,
    by=0.5,
    length.out=10)
```

De nuevo, es posible crear estas progresiones con cadenas de caracteres.

```
alfabeto <- letters[1:11]          # Progresión de las 11 primeras letras del alfabeto
alfabeto
```

La función `rep()` crea vectores a partir de repeticiones

```
rep(1,6)                          # El número 1 repetido 6 veces
rep("Santander",5)                # Igual, pero los caracteres siempre entre comillas
re entre comillas

sample(rep(c("lectura", "musica", "deportes"),25))    # Igual, pero los caracteres siempre entre comillas

rep(c(1,2,3),times=5)             # Secuencia 1 2 3 repetida 5 veces

rep(c(1,2,3),each=5)              # Secuencia 1 2 3 pero cada valor se repite 5 veces

rep(c(1,2,3,4),
    times=c(2,3,4,5))             # Repite cada elemento un diferente número de veces
```