

Sección 3

Unidad 1

- Crear vectores a partir de propiedades de otros vectores
- Utilizar operadores lógicos sobre vectores
- Extraer elementos de un vector que satisfagan ciertas condiciones lógicas

Unidad 2

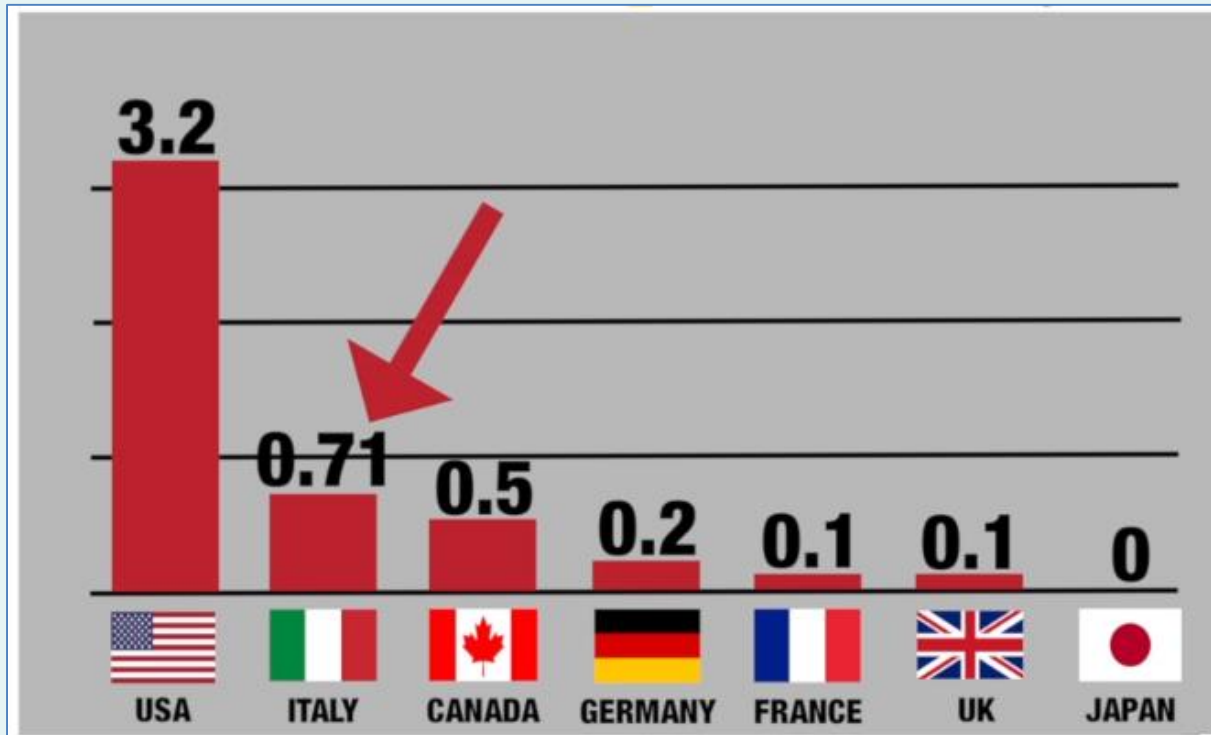
- Seleccionar datos en data tables utilizando funciones de la librería dplyr
- Modificar un data table agregando o cambiando columnas
- Crear data frames

Unidad 3

- Graficar datos en diferentes tipos de gráficos

Indexado

- Podemos aplicar operaciones lógicas para crear vectores a partir de datos seleccionados.



- Comparamos un vector a un valor lógico

```
> murder_rate <- murders$total / murders$population *  
  100000  
>  
> index <- murder_rate <= 0.71  
> index  
[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
[8] FALSE FALSE FALSE FALSE TRUE FALSE FALSE  
[15] FALSE TRUE FALSE FALSE FALSE FALSE FALSE  
[22] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
[29] FALSE TRUE FALSE FALSE FALSE FALSE TRUE  
[36] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
[43] FALSE FALSE FALSE TRUE FALSE FALSE FALSE  
[50] FALSE FALSE
```

- Obtenemos los elementos que cumplen condición

```
> murders$state[index]  
[1] "Hawaii" "Iowa" "New Hampshire"  
[4] "North Dakota" "Vermont"  
> sum(index)  
[1] 5
```

Operadores lógicos

Operador	Descripción
==	Es igual
!=	Distinto
<, <=, >, >=	Menor, menor o igual, mayor, mayor o igual
&&	Operador and (y)
	Operador or (o)
!	Operador not (no)

- Por ejemplo el operador lógico ‘y’

```
> TRUE & TRUE  
[1] TRUE  
> TRUE & FALSE  
[1] FALSE
```

```
> FALSE & FALSE  
[1] FALSE
```

Obtener un vector de valores lógicos

- Creamos condiciones lógicas

```
> oeste <- murders$region == "west"  
> seguro <- murder_rate <= 1
```

- Utilizamos el operador ‘&’ para identificar qué estados cumplen ambas condiciones

```
> index<-seguro&oeste  
> murders$state[index]  
[1] "Hawaii" "Idaho" "Oregon" "Utah"  
"Wyoming"
```

Funciones de indexado - Which()

Which(): Nos retorna las posiciones o índices de las entradas de un vector lógico que son TRUE

```
> x <- c(FALSE, TRUE, FALSE, TRUE, TRUE, FALSE)
> which(x)
[1] 2 4 5
```

- Obtenemos un elemento específico

```
> index <- which(murders$state == "Massachusetts")
> index
[1] 22
> murder_rate[index]
[1] 1.802179
```

match()

Match(): Busca por entradas en un vector y retorna las posiciones o índices para acceder a ellas.

```
> index <- match(c("New York", "Florida", "Texas"), murders$state)
> index
[1] 33 10 44
> murders$state[index]
[1] "New York" "Florida"  "Texas"
> murder_rate[index]
[1] 2.667960 3.398069 3.201360
```


%in%

%in%: Si en lugar de la posición queremos saber si un elemento de un vector se encuentra en un segundo vector utilizamos la función %in%

```
> x <- c("a", "b", "c", "d", "e")  
> y <- c("a", "d", "f")  
> y %in% x  
[1] TRUE TRUE FALSE
```