

Introducción a R

R es un lenguaje de programación orientado a la estadística.

Algo parecido a SPSS o Matlab.

Es código abierto y se entrega bajo licencia GPL (versión 2 o posterior). Además existen multitud de paquetes que complementan su funcionalidad.

Asignaciones

Podemos usar `<-` o el símbolo `=` para asignar un valor a una variable

```
a <- 10  
b=2  
"texto variable" -> c
```

```
print(a)
```

```
[1] 10
```

```
paste("El valor de a es:",a)
```

```
[1] "El valor de a es: 10"
```

Operaciones

Operaciones matemáticas

```
b+a #Suma
```

```
[1] 12
```

```
b*a #Multiplicación
```

```
[1] 20
```

```
b-a #Resta
```

```
[1] -8
```

```
b/a #División:
```

```
[1] 0.2
```

```
a/%b #División entera:
```

```
[1] 5
```

```
a %% b #Módulo:
```

```
[1] 0
```

```
a^b #Potencia:
```

```
[1] 100
```

Otro tipo de operador

```
3:10 #Secuencias
```

```
[1] 3 4 5 6 7 8 9 10
```

```
6 %in% 3:10 #contiene
```

```
[1] TRUE
```

Funciones básicas

Las funciones * siempre acaban en () * pueden recibir un número variable de parámetros * pueden existir parámetros por defecto

Ver las variables en memoria:

```
ls()
```

```
[1] "a" "b" "c"
```

Ver el tipo de variable:

```
typeof(a)
```

```
[1] "double"
```

Para concatenar textos usamos paste(a, b, c, ...)

```
paste("El valor de a es ",a)
```

```
[1] "El valor de a es 10"
```

Tipos de datos

Numérico

```
a <- 20  
typeof(a)
```

```
[1] "double"
```

Texto

```
s1 <- "Discovery"  
s1
```

```
[1] "Discovery"
```

```
typeof(s1)
```

```
[1] "character"
```

Logical

```
b1 <- TRUE
b2 <- T
b3 <- FALSE
paste("El valor de b1 es",b1)
```

```
[1] "El valor de b1 es TRUE"
```

```
paste("El valor de b2 es",b2)
```

```
[1] "El valor de b2 es TRUE"
```

```
paste("El valor de b3 es",b3)
```

```
[1] "El valor de b3 es FALSE"
```

Vectores

Los vectores se crean con la función `c()`, todos los elementos de un vector son del mismo tipo. Es decir, no se podrían mezclar caracteres y double.

En Python se llaman listas:

```
a = [1,2,3,40]
```

En Java arrays:

```
int[] a={1,2,3,40};
```

En R son vectores:

```
a = c(1,2,3,40)
```

```
alturas<-c(1.80,1.60,1.50,1.90,1.75,1.63,1.65)
print(paste("alturas es una variable de tipo:",typeof(alturas)))
```

```
[1] "alturas es una variable de tipo: double"
```

```
alturas
```

```
[1] 1.80 1.60 1.50 1.90 1.75 1.63 1.65
```

```
myTxt      <- c("rojo","verde","azul")
myTxt
```

```
[1] "rojo" "verde" "azul"
```

Acceso a elementos:

```
print(paste("El valor del tercer elemento es:",alturas[3]))
```

```
[1] "El valor del tercer elemento es: 1.5"
```

¡¡En R los vectores empiezan en 1!!

Data Frame

Un Dataframe es una tabla como la que podemos encontrar en cualquier hoja de calculo.

	A	B	C
1	Animal	Cantidad	Peso (Kg)
2	Perro	10	30
3	Periquito	3	0.01
4	Gato	4	2
5	Caballo	1	500

La información se estructura en columnas, cada columna puede ser de un tipo de variable diferente: categorica, numerica, fecha, etc..

Cada fila representa un elemento u objeto cuyos atributos están representados por las columnas.

```
df <- data.frame(Animal=c('Perro', 'Periquito', 'Gato', 'Caballo'),
                  Cantidad=c(10,3,4,1),
                  Peso=c(30, 0.01, 2, 500))
```

```
df
```

```
      Animal Cantidad  Peso
1     Perro      10 3e+01
2 Periquito       3 1e-02
```

3	Gato	4	2e+00
4	Caballo	1	5e+02

Podemos acceder a por filas y columnas:

```
df[1,]
```

	Animal	Cantidad	Peso
1	Perro	10	30

```
df[,2]
```

```
[1] 10 3 4 1
```

```
df$Cantidad
```

```
[1] 10 3 4 1
```

Podemos buscar aquellos que pesan más de 20kg

```
df[df$Peso>20,]
```

	Animal	Cantidad	Peso
1	Perro	10	30
4	Caballo	1	500

```
paste('Los animales que pesan más de 20kg son:', paste(df[df$Peso>20,'Animal']))
```

```
[1] "Los animales que pesan más de 20kg son: Perro,Caballo"
```

Ejemplo: Actividad deportiva

```
activities<-read.csv('data/activities_strava.csv', stringsAsFactor=T)
head(activities)
```

	Activity.ID	Activity.Date	Activity.Name	Activity.Type
1	2583296920	Aug 1, 2019, 3:51:15 PM	Afternoon Ride	Ride
2	2583296929	Aug 2, 2019, 5:53:50 AM	Morning Ride	Ride
3	2584111125	Aug 2, 2019, 1:15:49 PM	Afternoon Ride	Ride

4	2613027582	Aug 12, 2019, 3:58:59 PM	Afternoon Ride	Ride
5	2631611632	Aug 19, 2019, 5:24:35 AM	Morning Ride	Ride
6	2632473031	Aug 19, 2019, 1:29:48 PM	Afternoon Ride	Ride

	Activity.Description	Elapsed.Time	Distance	Relative.Effort	Commute
1	NA	50436	8.14	NA	false
2	NA	1745	7.96	NA	false
3	NA	944	6.97	NA	false
4	NA	4170	10.52	NA	false
5	NA	1744	7.62	NA	true
6	NA	1046	7.62	NA	true

	Activity.Gear	Filename	Athlete.Weight	Bike.Weight
1	NA	activities/2583296920.gpx	0	NA
2	NA	activities/2583296929.gpx	0	NA
3	NA	activities/2584111125.gpx	0	NA
4	NA	activities/2613027582.gpx	0	NA
5	NA	activities/2631611632.gpx	0	NA
6	NA	activities/2632473031.gpx	0	NA

	Elapsed.Time.1	Moving.Time	Distance.1	Max.Speed	Average.Speed	Elevation.Gain
--	----------------	-------------	------------	-----------	---------------	----------------

1	50436	38641	8143.7	14.2	NA	29.82933
2	1745	1682	7960.3	13.8	NA	128.79434
3	944	944	6969.6	14.9	NA	17.21061
4	4170	3003	10528.8	11.8	NA	62.29892
5	1744	1611	7628.7	12.0	NA	128.79936
6	1046	1038	7620.3	13.9	NA	30.42585

	Elevation.Loss	Elevation.Low	Elevation.High	Max.Grade	Average.Grade
1	NA	605.7	704.5	21.7	-1.03392804
2	NA	584.3	704.5	13.0	1.18462873
3	NA	584.3	704.5	10.5	-1.59837008
4	NA	765.5	794.6	7.2	-0.02944305
5	NA	584.3	704.5	12.9	1.33181274
6	NA	584.3	704.5	10.1	-1.30965972

	Average.Positive.Grade	Average.Negative.Grade	Max.Cadence	Average.Cadence
1	NA	NA	NA	NA
2	NA	NA	NA	NA
3	NA	NA	NA	NA
4	NA	NA	NA	NA
5	NA	NA	NA	NA

5	NA	NA	NA	NA
6	NA	NA	NA	NA
Max.Heart.Rate Average.Heart.Rate Max.Watts Average.Watts Calories				
1	NA	NA	NA	NA
2	NA	NA	NA	NA
3	NA	NA	NA	NA
4	NA	NA	NA	NA
5	NA	NA	NA	NA
6	NA	NA	NA	NA
Max.Temperature Average.Temperature Relative.Effort.1 Total.Work				
1	NA	NA	NA	NA
2	NA	NA	NA	NA
3	NA	NA	NA	NA
4	NA	NA	NA	NA
5	NA	NA	NA	NA
6	NA	NA	NA	NA
Number.of.Runs Uphill.Time Downhill.Time Other.Time Perceived.Exertion				
1	NA	NA	NA	NA
2	NA	NA	NA	NA
3	NA	NA	NA	NA
4	NA	NA	NA	NA
5	NA	NA	NA	NA
6	NA	NA	NA	NA

translation.missing..en.US.lib.export.portability_exporter.activities.horto
n_values.type

1
NA
2
NA
3
NA
4
NA
5
NA
6
NA

translation.missing..en.US.lib.export.portability_exporter.activities.horto
n_values.start_time

1
NA
2
NA

NA

3

NA

4

NA

5

NA

6

NA

Weighted.Average.Power Power.Count Prefer.Perceived.Exertion

1 NA NA NA

2 NA NA NA

3 NA NA NA

4 NA NA NA

5 NA NA NA

6 NA NA NA

Perceived.Relative.Effort Commute.1 Total.Weight.Lifted From.Upload

1 NA NA NA NA

2 NA NA NA NA

3 NA NA NA NA

4 NA NA NA NA

5 NA NA NA NA

6 NA NA NA NA

Grade.Adjusted.Distance Weather.Observation.Time Weather.Condition

1 NA NA NA

2 NA NA NA

3 NA NA NA

4 NA NA NA

5 NA NA NA

6 NA NA NA

Weather.Temperature Apparent.Temperature Dewpoint Humidity

Weather.Pressure

1 NA NA NA NA

NA

2 NA NA NA NA

NA

3 NA NA NA NA

NA

4 NA NA NA NA

NA

5 NA NA NA NA

NA

6 NA NA NA NA

NA

Wind Speed Wind Gust Wind Bearing Precipitation Intensity Sunrise Time

	wind.Speed	wind.Gust	wind.Bearing	Precipitation.Intensity	Sunrise.Time
1	NA	NA	NA	NA	NA
2	NA	NA	NA	NA	NA
3	NA	NA	NA	NA	NA
4	NA	NA	NA	NA	NA
5	NA	NA	NA	NA	NA
6	NA	NA	NA	NA	NA

	Sunset.Time	Moon.Phase	Bike	Gear	Precipitation.Probability	Precipitation.Type
--	-------------	------------	------	------	---------------------------	--------------------

1	NA	NA	NA	NA	NA	NA
2	NA	NA	NA	NA	NA	NA
3	NA	NA	NA	NA	NA	NA
4	NA	NA	NA	NA	NA	NA
5	NA	NA	NA	NA	NA	NA
6	NA	NA	NA	NA	NA	NA

	Cloud.Cover	Weather.Visibility	UV.Index	Weather.Ozone
1	NA	NA	NA	NA
2	NA	NA	NA	NA
3	NA	NA	NA	NA
4	NA	NA	NA	NA
5	NA	NA	NA	NA
6	NA	NA	NA	NA

translation.missing..en.US.lib.export.portability_exporter.activities.horton_values.jump_count

1
NA
2
NA
3
NA
4
NA
5
NA
6
NA

translation missing en US lib export portability_exporter activities horton

```
translation.missing..en.US.lib.export.portability_exporter.activitieshorto
n_values.total_grit
1
NA
2
NA
3
NA
4
NA
5
NA
6
NA
```

```
translation.missing..en.US.lib.export.portability_exporter.activitieshorto
n_values.avg_flow
1
NA
2
NA
3
NA
4
NA
5
NA
6
NA
```

Nos podemos quedar con ciertas columnas de interés:

```
activities <- activities[,c('Activity.Name', 'Activity.Type', 'Elapsed.Time')
activities$duracion_minutos <- activities$Elapsed.Time/60
head(activities)
```

	Activity.Name	Activity.Type	Elapsed.Time	Distance	duracion_minutos
1	Afternoon Ride	Ride	50436	8.14	840.60000
2	Morning Ride	Ride	1745	7.96	29.08333
3	Afternoon Ride	Ride	944	6.97	15.73333
4	Afternoon Ride	Ride	4170	10.52	69.50000
5	Morning Ride	Ride	1744	7.62	29.06667
6	Afternoon Ride	Ride	1046	7.62	17.43333

```
summary(activities)
```

Activity.Name	Activity.Type	Elapsed.Time	Distance
Morning Ride :45	Hike: 1	Min. : 197	Min. : 0.590
Afternoon Ride:28	Ride:92	1st Qu.: 1230	1st Qu.: 7.715
Evening Ride :17	Run : 1	Median : 1745	Median : 7.820
Afternoon Hike: 1	Walk: 1	Mean : 2627	Mean : 7.698
Lunch Hike : 1		3rd Qu.: 2095	3rd Qu.: 8.000
Morning Hike : 1		Max. : 50436	Max. : 15.200
(Other) : 2			

duracion_minutos

Min. : 3.283

1st Qu.: 20.508

Median : 29.083

Mean : 43.776

3rd Qu.: 34.917

Max. : 840.600

Podemos quedarnos solo con las actividades del tipo "Afternoon Ride" y con un tiempo inferior a 30 minutos:

```
afternoon_activity <- activities[
  activities$Activity.Name=='Afternoon Ride' &
  activities$duracion_minutos<30,]
head(afternoon_activity)
```

	Activity.Name	Activity.Type	Elapsed.Time	Distance	duracion_minutos
3	Afternoon Ride	Ride	944	6.97	15.733333
6	Afternoon Ride	Ride	1046	7.62	17.433333
7	Afternoon Ride	Ride	508	2.28	8.466667
9	Afternoon Ride	Ride	1198	7.72	19.966667
11	Afternoon Ride	Ride	1264	7.78	21.066667
13	Afternoon Ride	Ride	1185	7.72	19.750000

```
summary(afternoon_activity)
```

Activity.Name	Activity.Type	Elapsed.Time	Distance
Afternoon Ride:26	Hike: 0	Min. : 197	Min. : 0.590
Afternoon Hike: 0	Ride:26	1st Qu.: 1134	1st Qu.: 7.620
Evening Ride : 0	Run : 0	Median : 1212	Median : 7.720
Lunch Hike : 0	Walk: 0	Mean : 1151	Mean : 7.206

Morning Hike : 0
Morning Ride : 0
(Other) : 0

3rd Qu.:1263 3rd Qu.:7.780
Max. :1419 Max. :8.030

duracion_minutos

Min. : 3.283

1st Qu.:18.904

Median :20.200

Mean :19.188

3rd Qu.:21.046

Max. :23.650