Introducción a R

26 de abril de 2017

R

¿Qué es?

¿Quién lo usa?

Vamos a usar R + RStudio(IDE) :

- https://cran.r-project.org/mirrors.html
- https://www.rstudio.com/products/rstudio/

Cosas básicas

Asignación

```
r1 <- 1
r2 = 2

# Si pongo una variable sin ninguna operacion se imprime en pantalla
r1
```

```
## [1] 1
```

```
r2
```

```
## [1] 2
```

Una vez que una variable se crea, "vive" en la sesión salvo que la borre con `rm`

Tipos de datos

numeric/double

operaciones: +, - , *, /, ^, sqrt(.), exp(.), log(.), etc.

```
area1 <- 2 * pi * r1
```

```
area1
```

```
## [1] 6.283185
```

logical (TRUE/FALSE)

```
bool1 <- (area1 == 2)
```

```
(TRUE && (2 < 1)) || TRUE
```

```
## [1] TRUE
```

Tipos de datos

character

```
palabra1 <- "probando"
palabra2 <- "hola"

paste(palabra1,palabra1,palabra2,sep=", ") # concatenar caracteres con separador
```

```
## [1] "probando, probando, hola"
```

missing

```
varmissing<- NA
is.na(varmissing)
```

```
## [1] TRUE
```

Tipos de datos

vector

```
vector1 <- c(1, 5, 9, -1, 4)
vector1
```

```
## [1]  1  5  9 -1  4
```

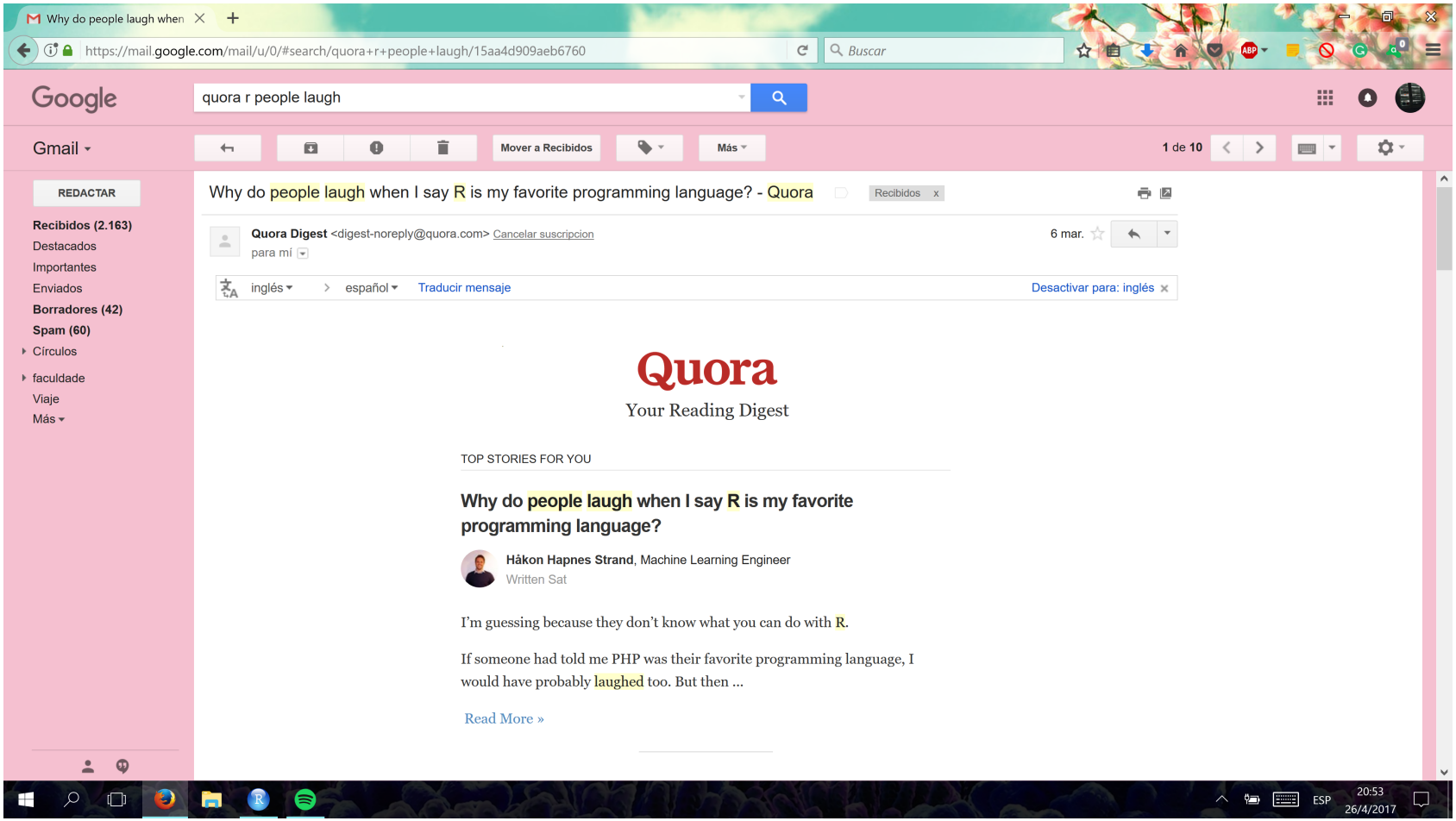
```
vector2 <- seq(1, 2, 0.5)
vector2
```

```
## [1] 1.0 1.5 2.0
```

```
vector3 <- rep(5, 7)
vector3
```

```
## [1] 5 5 5 5 5 5 5
```

Tipos de datos: vector



porque indexa desde 1

Tipos de datos: vector1=1, 5, 9, -1, 4

```
vector1[c(1,2)]
```

```
## [1] 1 5
```

```
c(vector1,c(1,2))
```

```
## [1]  1  5  9 -1  4  1  2
```

```
vector1[-2]
```

```
## [1]  1  9 -1  4
```

operaciones: +, -, *(escalar), mean(.), max(.), length(.), etc.

Tipos de datos: vector1=1, 5, 9, -1, 4

```
vector1 < 5
```

```
## [1]  TRUE FALSE FALSE  TRUE  TRUE
```

```
sum(vector1 < 5)
```

```
## [1] 3
```

listas:

pueden tener elementos de todo tipo

```
lista <- list(nombre="Juan", edad=35, nombrehijos=c("Juana", "Ana"))
lista$nombre
```

```
## [1] "Juan"
```

```
lista[[2]]
```

```
## [1] 35
```

if, for, while

Quiero obtener todos los indices TRUE del vector

(vector1<5)=TRUE, FALSE, FALSE, TRUE, TRUE

```
indicestrue <- c()
for (i in 1:length(vector1)) {
  if ((vector1<5)[i]) {
    indicestrue <- c(indicestrue, i)
  }
}
indicestrue
```

```
## [1] 1 4 5
```

Me gustaria reutilizar esto

Funciones

```
IndicesTrue <- function(vector){
  indicestrue <- c()
  for (i in 1:length(vector)) {
    if ((vector)[i]) {
      indicestrue <- c(indicestrue, i)
    }
  }
  indicestrue # o return(indicestrue)
}

IndicesTrue(c(TRUE, TRUE, FALSE))
```

```
## [1] 1 2
```

Cosas de proba

Simulaciones

```
sample(c(0, 1), 10, replace=TRUE)
```

```
##  [1] 0 1 1 1 1 0 0 1 0 1
```

Tirar una moneda equilibrada 10 veces

```
sample(c(0, 1), 10, replace=TRUE, prob=c(0.7, 0.3))
```

```
##  [1] 0 0 0 0 0 0 0 1 0 0
```

Tirar 10 veces una moneda con probabilidad 0.3 de salir cara

```
sample(c(rep("roja", 6), rep("blanca", 3)), 6, replace=FALSE)
```

```
## [1] "blanca" "roja"   "roja"   "roja"   "roja"   "roja"
```

Sacar 6 bolitas sin reposición de una urna con 6 rojas y 3 blancas

Análisis reproducible

Si un análisis que involucra una generación aleatoria

Tengo que garantizar que otra persona pueda obtener los mismos resultados

```
set.seed(0906)
sample(c(0,1),5,replace=TRUE)
```

```
## [1] 1 1 0 0 0
```

```
sample(c("BUENO","MALO"),3,replace=TRUE)
```

```
## [1] "MALO"  "BUENO" "MALO"
```

Variables famosas

Ejemplo con binomial Bi(10,1/2):

```
dbinom(5,10,0.5)    # puntual
```

```
## [1] 0.2460938
```

```
pbinom(2,10,0.5)    # acumulada
```

```
## [1] 0.0546875
```

```
qbinom(0.5,10,0.5)  # percentil
```

```
## [1] 5
```

```
rbinom(20,10,0.5)   # realizaciones del experimento
```

```
##  [1] 3 5 4 3 3 5 7 4 3 6 4 6 2 5 6 8 5 8 4 5
```

también: *geom, *dnbinom, *norm, *gamma, *exp, etc.

Aproximando probabilidades

Quiero calcular la proba aproximada de un evento:

P(evento) $\approx f_n$, con n grande

con f_n: frecuencia relativa con la que lo veo al evento en n repeticiones

EJEMPLO: aproximar la proba de sacar exactamente 5 caras o una cantidad par de caras en 10 tiros

```
casosfavorables <- 0
for (i in 1 : 1000) {
  muestra <- sample(c(0, 1), 10, replace = TRUE)
  if (sum(muestra) == 5 || (sum(muestra) %% 2) == 0) {
    casosfavorables <- casosfavorables + 1
  }
}
casosfavorables / 1000
```

```
## [1] 0.759
```

Primer ejercicio

Tiramos repetidamente una moneda sesgada con probabilidad 0.6 de salir cara (A) y 0.4 de salir ceca (E). En esta secuencia de tiros, un grupo es una secuencia consecutiva de tiros que salen del mismo lado de la moneda. Por ejemplo, los grupos de AEEEAEAEE son:

(A)(EEE)(A)(E)(A)(EE)

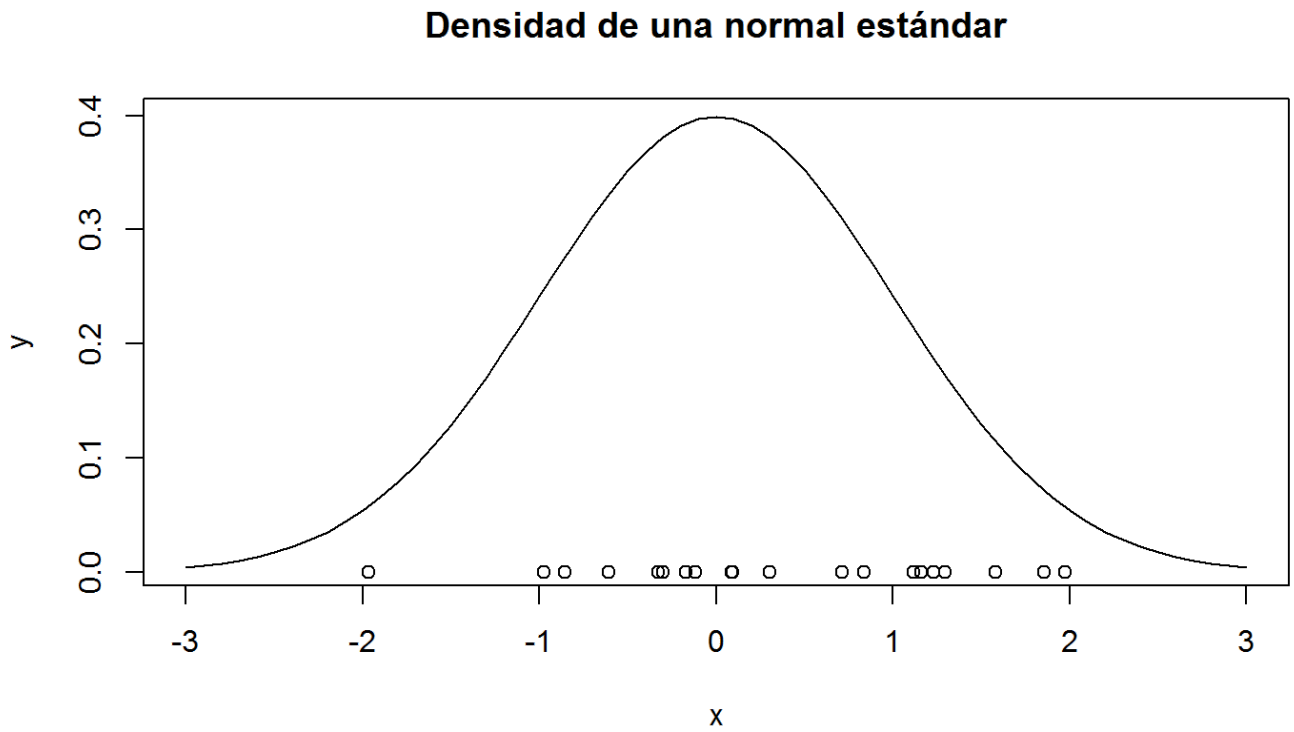
1. ¿Cuál es la probabilidad de exceder 5 grupos en 10 tiradas?
2. ¿Cuál es la probabilidad de ver exactamente 6 grupos en 10 tiradas si se que la cantidad de grupos excede 5?
3. ¿Cuál es el número esperado de grupos en 10 tiradas?

Segundo ejercicio

Aproximar la probabilidad de ganar el juego de Monty Hall para las dos estrategias posibles: quedarse o cambiar.

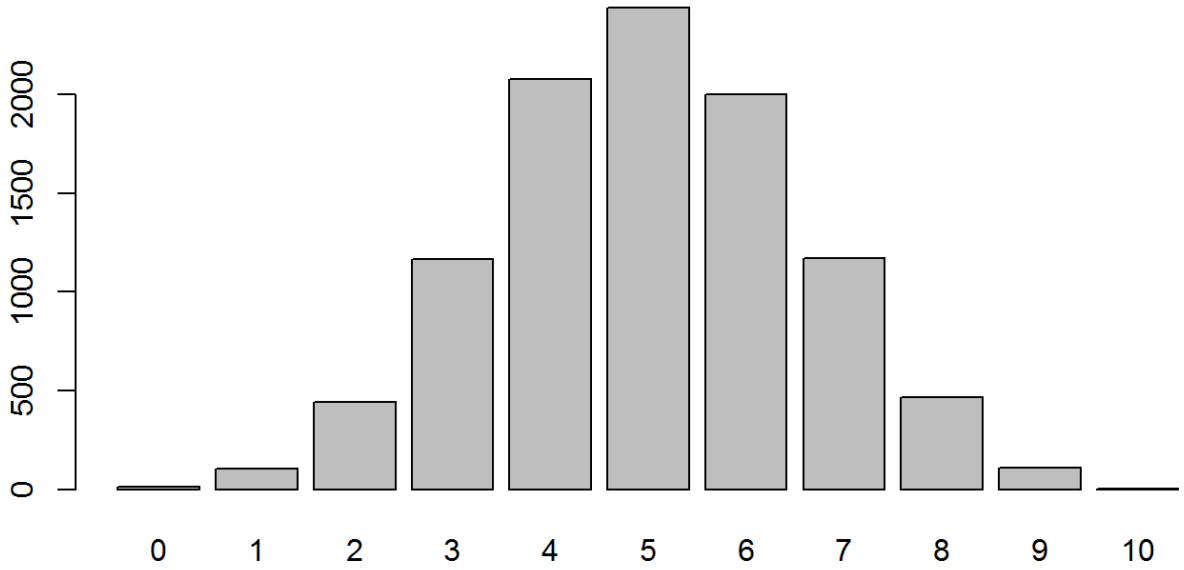
Gráficos

```
x <- seq(-3,3,0.1)
y <- dnorm(x)
plot(x,y,type="l",main="Densidad de una normal estándar")
points(rnorm(20),rep(0,20))
```



Gráficos

```
h <- rbinom(10000,10,0.5)
barplot(table(h))
```



Secretary problem

Reglas:

- Quiero contratar una sola persona entre n (conocido) candidatos.
- Los candidatos son entrevistados secuencialmente en un orden al azar.
- Puedo rankear a los candidatos de mejor a peor sin empates. La decisión se contratar a un candidato debe basarse sólo en el ranking relativo de los candidatos entrevistados hasta el momento.
- Un candidato rechazado no puede ser llamado ni contratado luego.
- Sólo estoy contento si elijo al mejor.

Si mi estrategia es descartar los primeros $r-1$ candidatos y después elegir el candidato que sea el mejor en el ranking relativo de los entrevistados hasta el momento.

1. Programar una función que dado n y p un porcentaje calcule la probabilidad de elegir al mejor dejando pasar aprox al p porciento de los n candidatos.
2. Hacer un gráfico de la probabilidad de elegir al mejor en función del porcentaje de $n = 50$.
3. ¿Cómo qué porcentaje de $n = 50$ debería tomar a r para maximizar la probabilidad de elegir al mejor?

Último

Comparar el grafico de barras de una hipergeometrica (500,350,10) con el de una binomial con parametros (10,0.7).