Simulación de Probabilidades con R-Commander

Índice

1. Fundamento teórico **1**
  1.1. Ejemplo: Obtener la Probabilidad de obtener cara en el lanzamiento de una moneda. 1

2. Simulación con R **1**

3. Simulación con R Commander **2**

4. Actividades **4**
  4.1. Lanzamiento de una moneda 4
  4.2. Simulación del funcionamiento de un circuito 5
  4.3. Concurso de triples da NBA (2018). 5

1. Fundamento teórico

El fundamento teórico para la obtención de probabilidades por simulación se basa en la llamada **Estabilidad de frecuencias**, es decir, dado un suceso **A** de un experimento aleatorio ξ, si repetimos indefinidamente e independientemente el experimento entonces:

$$\frac{\text{n}^{\text{o}} \text{ de veces que ocurre A}}{n} \longmapsto P(A)$$

Nota: Esto no quiere decir que $\lim_{n\to\infty} \frac{\text{n}^{\text{o}} \text{ de veces que ocurre A}}{n} = P(A)$.

Dende un punto de vista práctico, indica que para un valor **n** grande podemos aproximar $P(A)$ pola frecuencia relativa de aparición del suceso **A**. Por lo tanto, con uso del ordenador, podemos repetir aleatoriamente y una cantidad grande de veces el experimento en un tiempo relativamente pequeño.

1.1. Ejemplo: Obtener la Probabilidad de obtener cara en el lanzamiento de una moneda.

- $n_c = 0$
- Repetir **n** veces (**n** grande):
  a) Lanzar la moneda,
  b) Observar el resultado, si es `cara`, $n_c = n_c + 1$
- Aproximar la probabilidad de cara por su frecuencia de aparición:

$$P(cara) = \frac{n_c}{n}$$

Si $n = 10$ y los resultados son: +, c, c, +, +, c, c, +, c, + la probabilidad aproximada es: 0.5.

2. Simulación con R

Para simular experimentos discretos se puede usar, de forma general, la rutina `sample()` (ver la ayuda de la función escribiendo en R el comando *help(sample)*).

sample(x, size, replace = FALSE, prob = NULL)

```
sample(x=c("c","+"), size=10, replace=TRUE)
```

```
##  [1] "c" "c" "c" "c" "+" "+" "c" "c" "c" "+"
```

Esta función (como cualquier otra de R), toma como base la función `runif()` que genera valores aleatorios de una distribución continua que en este caso es la distribución Uniforme en el intervalo (0,1). Esta distribución Uniforme verifica que la probabilidad de un intervalo es su longitud. Por lo que, dado un suceso A con probabilidad p, se debe determinar un intervalo de la distribución uniforme de longitud p.

```
# Equivalentemente usando runif()
  p <- 1/2 # Prob. de Cara
  x <- runif(n=10)
  # resultado, cruz==FALSE, cara==TRUE
  x < p
```

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

```
  c("cara","cruz")[(x < p)+1]
```

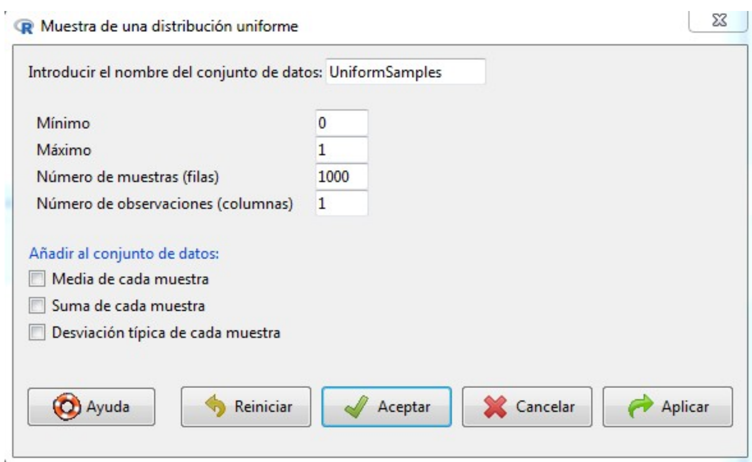
```
##  [1] "cara" "cara" "cara" "cruz" "cara" "cruz" "cruz" "cruz" "cara" "cara"
```

3. Simulación con R Commander

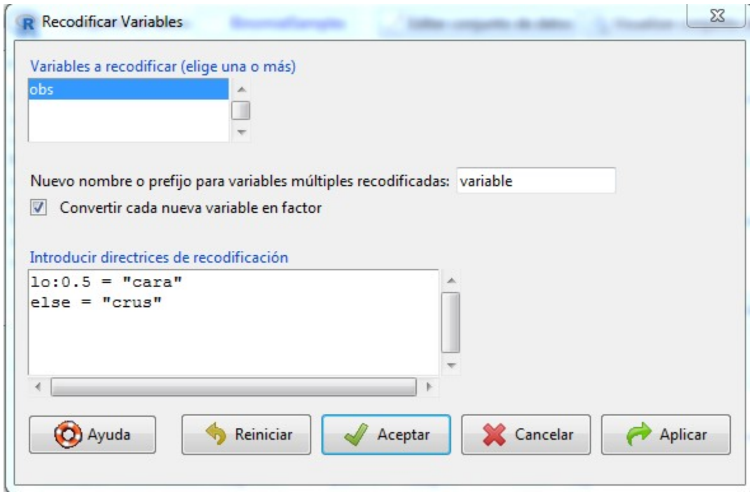
Si el número de repeticiones es n debemos obtener n filas del conjunto de datos, cada fila representa una repetición independiente del experimento. Para ello, se usan los menús de generación de muestras de variables aleatorias (discretas o continuas) que hay en *Distribuciones* o en *Estadística básica/Variables aleatorias*.

Así para el caso de la probabilidad de obtener "cara", con n=1000, vamos al menú *Estadística básica/Variables aleatorias/Distribuciones continuas/Distribución uniforme/Muestra de una distribución uniforme*

1.- Generamos los valores aleatorios de la uniforme.



2.- Transformamos los valores aleatorios.



3.- Obtenemos el resumen del resultado del experimento.

```
UniformSamples <- as.data.frame(matrix(runif(1000*1, min=0, max=1), ncol=1))
  rownames(UniformSamples) <- paste("sample", 1:1000, sep="")
  colnames(UniformSamples) <- "obs"
UniformSamples <- within(UniformSamples, {
  variable <- Recode(obs, 'lo:0.5 ="cara"; else = "cruz"', as.factor=TRUE)
})

# Estadística Básica/Estadística Descriptiva/Distribución de frecuencias de variables cualitativas
calcular_frecuencia(df.nominal=UniformSamples["variable"], ordenado.frec=FALSE, df.ordinal=NULL,
  cuantil.p=0.5, iprint = TRUE)
```

```
##
## -------------------------------
##
## Variábeis nominais:
##
## Variábel: variable
##       ni    fi
## cara 486 0.486
## cruz 514 0.514
## N= 1000
```

Para este ejemplo en concreto podemos ahorrar el paso de la uniforme generando la muestra de una binomial de parámetros $n = 1, p = 0.5$ con el menú *Estadística básica/Variables aleatorias/Distribuciones continuas/Distribución Binomial/Muestra de una distribución Binomial*

```
BinomialSamples <- as.data.frame(matrix(rbinom(100*1, size=1, prob=0.5), ncol=1))
rownames(BinomialSamples) <- paste("sample", 1:100, sep="")
colnames(BinomialSamples) <- "obs"
# Estadística Básica/Estadística Descriptiva/Distribución de frecuencias de variables cualitativas
calcular_frecuencia(df.nominal=BinomialSamples["obs"], ordenado.frec=FALSE, df.ordinal=NULL,
  cuantil.p=0.5, iprint = TRUE)
```

```
##
## -------------------------------
##
## Variábeis nominais:
##
```

```
## Variábel: obs
##    ni   fi
## 0 52 0.52
## 1 48 0.48
## N= 100
```

4. Actividades

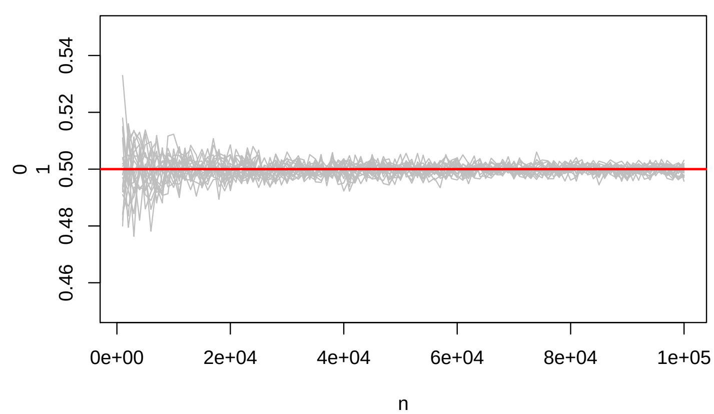
4.1. Lanzamiento de una moneda

En este ejercicio nos centraremos en el experimento *lanzamiento de una moneda*. Inicialmente simularemos el lanzamiento de una moneda ficticia con el objetivo de calcular la probabilidad de obtener el resultado *cara*.

1.- Simular el proceso de tirar una moneda un número grande de veces ($n = 100$).

```
2.- Calcular la proporción de veces que se obtuvo el resultado *cara*.

3.- Repetir el procedimiento pero ahora aumentando el número de lanzamientos a $n=200$ y suponiendo que

4.- El código siguiente genera un gráfico con 20 líneas y cada línea es la unión de la frecuencia relati
```

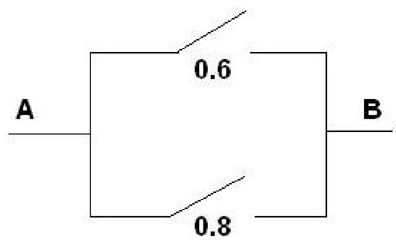
```r
# n vai aumentando de 1000 en 1000
n <- seq(from=1000,to=100000,by=1000)

  ll <- function(n=100, prob=0.5)  return( mean( runif(n) < prob) )# < prob --> cara
prob <- sapply( n, ll, p=0.5)

# e usando agora a función plot temos
plot(n, prob, ylab=c(0,1),type="l",ylim=c(0.45,0.55), col='grey') # Axustar a ollo os límites do eixo
points(n, sapply(n,ll,p=0.5), type="l", col='grey')
# repito isto 18 veces máis
d <- replicate( 18, points( n, sapply(n,ll,p=0.5), type="l", col='grey'))
abline(h=0.5, col="red",lwd=2)
```



Si repetimos indefinidamente el proceso de generar las líneas (en el código reemplazar 18 por `+Inf`), ¿qué gráfico obtendríamos?

4.2. Simulación del funcionamiento de un circuito

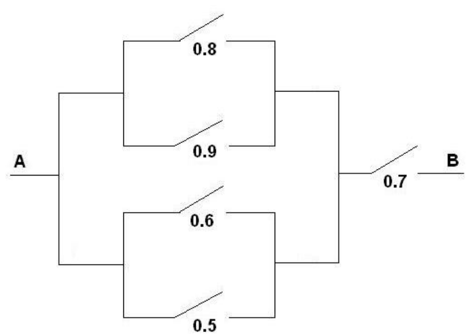
Simular el paso de corriente a través del siguiente circuíto, donde figuran las probabilidades de pasar la corriente por cada uno de los dispositivos. Utilizar una muestra de tamaño 200.



Pasos:

- Generar dos vectores x_1 e x_2 de tamaño *200* que indiquen si el dispositivo superior (con probabilidad 0.6) e inferior (con probabilidad 0.8) estan abierto ou cerrado.
- Crear una nueva variable z que tome el valor 1 si x_1 está abierto o x_2 está abierto, y cero en el otro caso.
- Calcular a frecuencia relativa del valor 1 de la variable z, y ver que se aproxima al valor de la probabilidad de circular corriente por el circuíto.
- Comproba si el valor resultante está cerca del teórico.

Realizar el mismo estudio de simulación pero ahora con el siguinte sistema:



4.3. Concurso de triples da NBA (2018).

Simular el concurso anual All-Star Weekend da NBA de triples que se celebra el sábado anterior al All-Star Game de la NBA (extraido de Wikipedia). Los concursantes tratan de puntuar encestando tanto como pueden en un minuto. Hay cinco carriles de balones desplegados a lo largo del arco de la línea de tres puntos: uno al principio, otro al final, otro en la mitad y otros dos a 45º de la mitad. De los cinco balóns que se disponen en cada carro, cuatro son de un valor de un punto, mientras que el quinto, un balón multicolor, vale dos puntos. Se pretende saber que jugador ganaría entre Larry Bird (ganador los años 1986/87/88), Craig Hodges (ganador 1990/91/92) y el ganador del 2018 Devin Booker, teniendo en cuenta las siguientes porcentajes de acierto desde las 5 posiciones:

- Las porcentajes de Larry Bird son: 35.6 %, 37.6 %, 39.6 %, 37.6 %, 35.6 %.
- Las porcentajes de Craig Hodges son: 36.4 %, 40.0 %, 40.5 %, 40.0 %, 36.4 %.
- Las porcentajes de Devin Booker son: 32.9 %, 35.7 %, 38.5 %, 35.7 %, 32.9 %.

Se supone que el tiempo termina cuando se lanzan los 25 balones. En caso de empate entre los jugadores esta se anula. Simular 100000 juegos independientes y obtener la aproximación de la probabilidad de que gane cada uno de los jugadores.