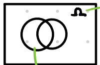
Tema 2: PROBABILIDAD

Conceptos Básicos



Ω → Espacio Muestral

Ej: Dado 6 caras

$\Omega = \{1,2,3,4,5,6\}$

Evento: Cualquier subconjunto Ω

Ej: Sacar un 1 en un 1er Lanzamiento ▸ $\{1\} \subset S$

Sacar un 1 ó un 2 1er Lanzamiento ▸ $\{1,2\} \subset S$

Falla antes de 5º ▸ $F(0,5)$

Si $A \cap B = \emptyset$ → Mutuamente Excluyentes

Si $A \cup B = \Omega$ → Suceso seguro

Álgebra de los eventos

- $A \subseteq B$ ⟶ si $\begin{cases} A \subset B \\ B \subset A \end{cases} = A = B$
- Leyes conmutativas
  $A \cup B = B \cup A$ $\quad A \cap B = B \cap A$
- Leyes asociativas
  $(A \cup B) \cup C = A \cup (B \cup C)$ $\quad (A \cap B) \cap C = A \cap (B \cap C)$
- Leyes Distributivas
  $(A \cup B) \cap C = (A \cap C) \cup (B \cap C)$
- Leyes de Morgan
  $\overline{(A \cup B)} = \bar{A} \bar{\cup} \bar{B} = \bar{A} \cap \bar{B}$

Regla LaPlace

Casos EquiProbables:

$$P(E) = \frac{\text{nº casos favorables}}{\text{nº Casos Totales}}$$

¿Probabilidad 🎲 salga un 1 ó un 3?

$P(\{1\} \cup \{3\}) = \frac{2}{6}$

Axiomas de Probabilidad

$P(E)$: nº q describe como de habitual/raro ⋯› evento ocurra

Interpretación frecuentista: Si $P(E) \approx A$ y repites varias veces el experimento, la proporción de veces q ocurre el evento es A

Axiomas de Kolmogorov
1. $P(E) \geq 0$
2. $P(\Omega) = 1$
3. Mutuamente Exclusivos: $E_i \cap E_j = \emptyset$ $\quad i \neq j$

$$P\left(\bigcup_{i=1}^{n} E_i\right) = \sum_{i=1}^{n} P(E_i)$$

$P(E_1 \cup E_2 \cup E_3) = P(E_1) + P(E_2) + P(E_3)$

Teoremas de Probabilidad

- Si $E_1 \subset E_2$ ⋯⋯› $P(E_1) \leq P(E_2)$ y $P(E_2 - E_1) = P(E_2) - P(E_1)$
  (E_1: $\{1\}$, E_2: {impar})
- $0 \leq P(E) \leq 1$ $\quad$ • $P(\emptyset) = 0$
- $P(\bar{E}) = 1 - P(E)$
- Formulas de Inclusión - exclusión
  $$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

Análisis Combinatorio

Permutaciones

¿De cuantas formas puede ordenarse en Fila 5 🧍?

$\underline{5}\ \underline{4}\ \underline{3}\ \underline{2}\ \underline{1}$

↳ $5 \cdot 4 \cdot 3 \cdot 2 \cdot 1$

▸ IMP el orden
▸ NO Repetición

$\boxed{P_n = n!}$

Variaciones

10 atletas → ¿De cuantas formas puede acabar el Podio?

$\frac{\text{oro}}{10}\ \frac{\text{plata}}{9}\ \frac{\text{bronce}}{8} = 10 \cdot 9 \cdot 8$

$\boxed{P(n,r) = \frac{n!}{(n-r)!}}$ número de variaciones de n objetos tomados de r en r

$\frac{10!}{(10-3)!} = \frac{10!}{7!} = \frac{10 \cdot 9 \cdot 8 \cdot \cancel{7!}}{\cancel{7!}} = 10 \cdot 9 \cdot 8$

▸ NO repetición
▸ IMP orden

Combinaciones

Grupos de 3 números (sin importar el orden) q pueden formarse a partir de 1,2,3,4,5?

$\{1,2,3\}$ → 123 132 213 231 312 321 ⇒ 1 CASO

__ __ __

↳ $\frac{5 \cdot 4 \cdot 3}{3 \cdot 2 \cdot 1} = 10$

▸ NO orden
▸ Repetición

$C(n,r) = \binom{n}{r} = \frac{n!}{r!\,(n-r)!}$

Permutaciones con Repetición

¿Cuantas "Palabras" ≠ pueden formarse a partir de "PEPPER"?

▸ Hay Repeticiones

$P_1\ P_2\ P_3\ E_1\ E_2\ R$

$P_2\ P_1\ P_3\ E_1\ E_2\ R$

{ = Palabra → $\frac{6!}{3! \cdot 2! \cdot 1!} = 60$ casos

$\boxed{P_n = \frac{n!}{n_1!\, n_2! \ldots n_k!}}$ } n objetos clasificados en k clases de objetos idénticos

Ejemplo: Cadena 4 caracteres, los dos primeros letras y los dos últ números [Letras NO se pueden repetir]

$\underbrace{\underline{26}\ \underline{25}}_{\text{letras}}\ \underbrace{\underline{10}\ \underline{10}}_{n^{os}}$

¿Importa el orden? → Sí ⟿ Variación

Letras: Variación sin repetición ▸ $V_{26,2} = \frac{26!}{(26-2)!} = 26 \cdot 25$

Números: Variación con repetición ▸ $VR_{10,2} = 10^2$

Total: $26 \cdot 25 \cdot 10^2$

Variaciones con repetición

¿Cuántas cadenas de 5 bits existen?

$\underline{2}\ \underline{2}\ \underline{2}\ \underline{2}\ \underline{2}$

↓

2^5 cadenas ≠

$\boxed{V_r(n,r) = n^r}$

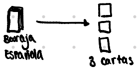
- repetición de r elementos tomados entre n

▸ Repetición
▸ imp el orden

10000 ≠ 01000

Tema 2: PROBABILIDAD

Procedimiento Constructivo / algorítmico



¿ Probabilidad de obtener una Pareja ?

Caso 1: No importa el orden

```r
sim_cases = function(){
  cartas = paste(c("E", "C", "B", "O"), rep(1:10, each = 4))
  set::as:set(sample(cards, 3))           #--> nº formas-Combinacion: Choose(40,3)
}

sim_fav_case() = function(){
  # 1) Elijo un numero para mi pareja
  numero_pareja = sample(1:10,1)                          #--> nº formas: 10
  # 2) ELijo 2 palos entre los 4 posibles para combinar con el numero
  palos = sample(c("E", "C", "B", "O"),2)                 #--> nº formas: choose (4,2)
  mi_pareja = paste (palos, numero_pareja)
  # 4) Entre las 40 - 4 cartas con un numero distinto a mi pareja, elijo una
  num_validos = setdiff(1:10, numero_pareja)
  cartas_restantes = paste(c("E", "C", "B", "O"), rep(num_validos, each = 4))
  final_carta = set::as.set(c(mi_pareja, final_carta))    #--> nº formas = 36
  num_validos
}
```

$$P(\text{una Pareja}) = \frac{10 \cdot \binom{4}{2} \cdot 36}{\binom{40}{3}} \simeq 0.219 \Rightarrow \frac{(10 * \text{choose}(4,2) * 36)}{\text{choose}(40,30)}$$

Paste (1:5, c("O","A")) ⟶ 1O 2A 3O 4A 5O
setdiff (1:5 , 3) → 1 2 4 5 (nº 1 al 5, ≠ del 3)
Paste (1:10 , rep (1:3, each = 2) → 1 1 2 1 3 2 4 2 5 3 6 3 7 1 ...

Caso 2: Sí importa el orden

```r
sim_cases = function(){
  cartas = paste(c("E", "C", "B", "O"), rep(1:10, each = 4))
  sample(cards, 3)          #--> nº formas: variaciones(40,3)
}

sim_fav_cases = function(){
  # 1) Elijo un numero para mi pareja
  numero_pareja = sample(1:10,1)                          #--> nº formas: 10
  # 2) Entre las 40 - 4 cartas con un numero distinto a mi pareja, elijo una
  num_validos = setdiff(1:10, numero_pareja)
  cartas_restantes = paste(c("E", "C", "B", "O"), rep(num_validos, each = 4))
  carta_final = sample(cartas_restantes, 1)               #--> nº formas:36
  # 3) Esta vez, debo elegir una posición para la carta
  cartas = rep(NA, 3)
  indice = sample (1:3, 1)                                #--> nº formas: 3
  cartas[indice] = carta final
  # 4) Elijo 2 palos entre los 4 posibles. El orden importa
  palos = sample(c("E", "C", "B", "O"), 2)                #--> nº formas: variaciones(4,2)
  mi_pareja = paste(palos, numero_pareja)
  cartas[setdiff(1:3, indidce)] = mi_pareja
  cartas
}

(10 * 36 * 3 * variaciones(4,2))/variaciones(40,3)
```

Asignación de Probabilidad en espacios finitos Equiprobables

En un bar, cinco amigos han pedido tres cafés con leche y dos cañas. ¿De cuántas formas pueden repartirse las bebidas?

````
```{r}
#COMBINATORIA


c("tino" = "canna", "Gauss" = "cafe")
#tino será la posición 1 y gauss la 2
c("canna", "cafe") #el orden IMPORTA
c("canna", "cafe", "cafe") # puede haber elementos REPETIDOS

#ORDEN importa + elemn REPETIDOS = Permutaciones con repeticiones (N!)
 5! / (2! 1!)

    # Elegir 5 numeros, 3, y estos representan las personas que toman cafe
     # Esto sería el problema de las bolas
      choose(5,3)
````

¿Cuál es la probabilidad de obtener una mano de tres espadas y dos copas de una baraja española?

````
```{r}
# 3 espadas y 2 copas

#CASOS TOTALES
casos_totales = function(){
  baraja = paste(c('H', 'C', "B", "O"), rep(1:10, each = 4))
  set::as.set(sample(baraja,5))    #choose(40,5)
}

#CASOS FAVORABLES
casos_fav = function(){
    # 1°) sacamos dos compas --> choose (10,2)
    copas = sample("C" , 1:10, 2)
    # 2°) sacamos 3 espadas --> choose(10,3)
    espadas = sample("E", 1:10, 3)
}

laPlace = (choose(10,2) * choose(10,3))/choose(40,5)
```
````

El próximo Enero vas a visitar las ciudades \textit{A, B, C, D}.
Si cada visita requiere un día ¿De cuántas formas distintas puedes programar tus viajes?

````
```{r}
# Para cada ciudad --> q dia vamos a ir (de 31 dias hay q elegir 4)

# Elijo que dia visito la ciudad: necesito 4 nums
# 7, 8, 9, 10 ≠ 8, 7, 30 , 2 -->  IMPORTA EL ORDEN
# Problema del podium...

  # 1) Elijo un numero para la ciudad a --> de 31
  # 2) Elijo un numero para la ciudad b --> de 30
  # 3) Elijo un numero para la ciudad c --> de 29
  # 4) Elijo un numero para la ciudad d --> de 28

  31*30*29*28 #variacion (31,4)
```
````

Durante el mes de Enero, deseas viajar a las ciudades \textit{A, B, C, D} en este orden. ¿Cuántas formas distintas de viajar tienes?

````
```{r}
# Elijes 4 numeros al azar --> el menor: A, el 2° menor:B,...
# 12 15 30 20 --> A=12, B=15, C=20, D=30
# 30 15 12 20 --> A=12, B=15, C=20, D=30
    # NO IMPORTA EL ORDEN
    # NO HAY REPETICIONES
choose( 31 ,4 ) #combinacion de 31 elemntos sobre 4

rep("no viajo",31)
calendario = rep("no viajo" , 31 )
calendario
#calendario[posiciones] = "X"
calendario[sample(31,4)] = "X"

```
````

¿Cuál es la probabilidad de sacar 25 caras en 50 tiradas de una moneda?

````
```{r}
#Regla laPlace
#num casos fav / num casos totales

#Casos Totales
#salida: vector de 50 posiciones,
#queremos repetir elemntos luego nos importa el orden
#funcion que saca elemntos al azar: sample
sample(c("C", "X"),1) #simular tirar una moneda al aire
#lo repetimos 50 veces (huir bucles)

#sample funciona como una urna con solo dos posibles resultados
#Sample por defecto hace muestreo sin desplazamiento (urna)
#MUESTREO CON DESPLAZAMIENTO
#meto la mano en la urna, lo cojo, lo anoto y lo vuelvo a dejar
sample(c("C", "X"),50, replace= "TRUE")
# Replace --> vuelve a dejarlo // 50--> repite 50 veces
# Replace = TRUE, es una pista fuerte del patron de bits

num_casos_totales = function(){
  sample(c("C", "X"),50, replace= "TRUE")          #--> 2^50
}

casos_totales = 2^50

#----------------- CASOS FAVORABLES-----------
#resultado experimento es igual a trial
 #generar vectores donde haya 25 caras
 trial = rep("X", 50)    #generar vector lleno de X

#poner 25 "C" en 25 posiciones al azar
#Elegir 25 índices donde poner "C"
sample(1:50, 25)
indices = sample(1:50, 25)   #--> Variaciones (orden) y Combinaciones(no orden)[nuestro caso]
              #indice 1, 2, ... = indice 2, 1, ...
              # choose(50 , 25) [combinacion]

trial[indices] = "C"

num_casos_fav = function(){
   trial = rep("X", 50)
   sample(1:50, 25)
   indices = sample(1:50, 25)
}
num_casos_fav= choose(50,25)

print(num_casos_fav/ casos_totales)
#choose(50,24)/ 2^50
````

10 parejas de hermanos se han apuntado a fútbol. Si se hace un equipo de 8 personas al azar, cuál es la probabilidad de que no haya hermanos en el equipo.

````
```{r}
## PLANTEAMIENTO
# creamos 10 parejas (sample y by=2 ) --> le asignamos unos numeros
#añdimos identificador de género (chico o chica)
paste(1:10, c("O", "A"))

  #ocurre reciclaje, 1 "O" , 2 "A",...
  alumnos = paste(rep(1:10, each=2), c("O", "A"))


sim_all_cases = function(){
  alumnos = paste(rep(1:10, each=2), c("O", "A"))        #--> Combinacion choose(20,8)
  as.set::as.set(sample(alumnos, 8 ))  #transformar a un conjunto el set
}
casos_totales = choose(20,8)

##Forzar a q no haya hermanos en el equipo
##Forzar a q haya 8 numeros distintos

   sim_fav_cases= function(){
      #1) elijo 8 numeros distintos
      parejas_hermanos = sample(1:10, 8)                 # --> choose(10,8)
      #2) falta por saber si es chico o chica
      sexo = sample(c("O","A"), 8 , replace = TRUE)      # --> 2^8
   paste(parejas_hermanos, sexo)
   }

 casos_favorables = 2^8 *choose(10,8)
 print(casos_favorables/casos_totales)
````

10 tarjetas blancas y 15 tarjetas negras se barajan conjuntamente.
Calcula la probabilidad de que la primera tarjeta blanca esté en la posición 5.

````
```{r}
#CASOS TOTALES
tarjetas = c(rep("B", 10), rep("N", 15))
#reordenarlas al azar
sample(tarjetas)

sim_all_cases = function(){
  tarjetas = c(rep("B", 10), rep("N", 15))
  sample(tarjetas)
  #--> Orden imp (posicion 5), hay elem repetidos  == missisipi
  # permutaciones con repeticion : factorial(25)/factorial(15)*factorial(10)
}
casos_totales = factorial(25)/(factorial(15)*factorial(10))

#CASOS FAVORABLES
sim_fav_cases = function(){
  #1) creamos vector N N N N B
  comienzo = c("N", "N", "N", "N", "B")
  #2) añadimos en aleatorio el resto de tarjetas
  resto =  c(rep("B", 9), rep("N", 11))
  vector_final = c(comienzo, sample(resto))
}
    # resto --> hay repeticiones, importa el orden : permutaciones con rep
    casos_favorables = factorial(20) / (factorial(11)*factorial(9))

#RESULTADO
print(casos_favorables/ casos_totales)
````

Si hay n personas en una clase, ¿Cuál es la probabilidad de que ningún par de personas celebren el cumpleaños el mismo día? ¿Y la de que haya al menos una coincidencia?

````
```{r}
# n = 45 personas

#CASOS TOTALES
sim_all_cases = function(n){
  fechas_alumnos = sample(1:365, 45, replace = TRUE)  #--> 365 ^ 45
  fechas_alumnos
}
casos_totales = 365^45

#CASOS FAVORABLES
sim_fav_case = function(){
  fechas_alumnos = sample(1:365, 45) #caso favorable: sin repeticion de fechas
  #--> PODIUM: variaciones (365,45)
}
casos_favorables = variaciones(365 , 45)/(365 ^ 45)

  print(casos_favorables/casos_totales)
  solve1 = casos_favorables/casos_totales ### 1° pregunta

  ### 2° pregunta
  solucion = 1 - variaciones(365 , 45)/(365 ^ 45)
````
````

Tema 2: PROBABILIDAD

Simulaciones

simula caso ⟶ Simula éxito ➔ Repetimos varias veces

Final: Simulations = replicate (N , { })

veces q se repite

función q simula un caso de éxito

Print (sum(simulations) /N)

Se extraen tres cartas de una baraja española.
¿Cuál es la probabilidad de obtener una pareja (dos cartas del mismo valor)?

```
```{r}
### SIMULACIONES ###
baraja =  paste(rep(1:10, each = 4) , c('H', 'C', "B", "O"))
prueba = sample(baraja,3)
strsplit(prueba , " ") #separamos "1" "B"

#OPCION CON SAPPLY
  #ir elemento a elemento (sapply) y quedarme con el primero de ellos
            #sobre q elemntos quiero iterar --> strsplit
            sapply(strsplit(prueba , " "))
            #aplicar funcion que sobre el vector--> me devuelva el primero
            sapply(strsplit(prueba , " "), function(vec) vec[1])

# OPCION CON PURR
            purrr::map(strsplit(prueba , " "), 1) #1 --> 1º elemento
                      # map --> devuelve lista
                      #map_tipodedatoquequeremos --> map_char, map_int,...
            ns = purrr::map_chr(strsplit(prueba , " "), 1)


#comprobar CASO EXITO
  counts = table(ns)  #contar cuantas veces ocurre cada elemento de nuestro vector
  (length(counts) == 2) && (all(counts == c(1, 2) | counts == c(2, 1)))
  ## longitud 2                cuentas sean 1 y 2 o 2 y 1
  (length(counts) == 2) && (all(sort(counts) == c(1, 2)))
```

```
  ###########
  N = 50000
  simulations = replicate(N,{
    baraja =  paste(rep(1:10, each = 4) , c('H', 'C', "B", "O"))
    prueba = sample(baraja,3)
    ns = purrr::map_chr(strsplit(prueba , " "), 1)
    counts = table(ns)
    (length(counts) == 2) && (all(counts == c(1, 2) | counts == c(2, 1)))
  })

  print(
  sum(simulations)/ N
  )
```

¿Cuál es la probabilidad de sacar 25 caras en 50 tiradas de una moneda?

```
```{r}
# Simulate one trial
trial = sample(0:1, 50, replace = TRUE)
# Check if the event has occurred (TRUE) or not (FALSE)
has_occurred = (sum(trial) == 25)

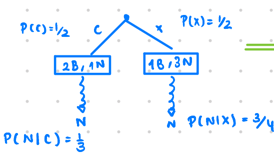
# A simulation involves repeating the previous setup a looooooot of times
# We can use a for loop or...
nb_sims = 50000
events = replicate(nb_sims, {
  trial = sample(0:1, 50, replace = TRUE)
  (sum(trial) == 25)
})
# Use Laplace rule: succesful events / total events
print(sum(events) / nb_sims)
```

En una urna hay 3 bolas rojas numeradas del 1 al 3 y 3 bolas negras numeradas del 1 al 3. Sacamos dos bolas, ¿Probabilidad de sacar dos treses o una bola negra y otra blanca?

```
  #====================PROBLEMA ORIGINAL ==========================
  bolasr = paste(1:3 , "R")
  bolasn = paste(1:3 , "N")
  bolas_total = c(bolasr,bolasn)

  ## Probabilidad UNA NEGRA y UNA ROJA
      prueba = sample(bolas_total, 2)
      ns_color = sapply(strsplit(prueba , " "), function(vec) vec[2])
      #!all(ns_color == "N" | ns_color == "R" )
  ## Probabilidad dos 3's
      #prueba = sample(bolas_total, 2)
      ns_tres = sapply(strsplit(prueba , " "), function(vec) vec[1])
      #all(bolas == 3)
      (!all(ns_color == "N" & ns_color == "R" )) || (all(prueba == 3))
```

Existen dos urnas. La primera tiene dos bolas blancas y una negra; la segunda tiene una blanca y tres negras. Se lanza una moneda al aire. Si sale cara, se saca una bola de la primera urna; si sale cruz, se saca de la segunda. ¿Cuál es la probabilidad de sacar una bola negra? → $P(N)$



$$P(N) = P(C)\cdot P(N|C) + P(x)\cdot P(N|X)$$
$$= \frac{1}{2}\cdot\frac{1}{3} + \frac{1}{2}\cdot\frac{3}{4} = 13/24$$

Probabilidad Condicionada

$$\boxed{P(B|A) = \frac{P(B\cap A)}{P(A)}} \longrightarrow P(B\cap A) = \begin{matrix} P(B|A)\cdot P(A) \\ P(A|B)\cdot P(B) \end{matrix}$$

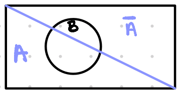
suceso — info extra

Existe un 30 % de probabilidad de que tu empresa cree una nueva delegación en Toledo. Si es así, existe un 60 % de probabilidad de que te den el puesto de "informático jefe". ¿Cuál es la probabilidad de que seas elegido informático jefe en Toledo?

$P(T)$ — $P(J|T)$ — $P(J\cap T)$

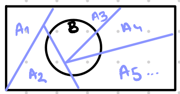
$$P(J\cap T) = P(T)\cdot P(J|T) = 0'3\cdot 0'6 = 0'18$$
$$\circ\ P(J)\cdot P(T|J)$$

Fórmula de Probabilidad Total (Laplace)



$$\rightarrow B = (B\cap A)\cup(B\cap \bar{A})$$ axioma 3

$$P(B) = P(B\cap A) + P(B\cap \bar{A})$$
$$= P(A)P(B|A) + P(\bar{A})P(B|\bar{A})$$



Particiones arbitrarias

$$\rightarrow P(B) = P(B|A_1)P(A_1) + P(B|A_2)P(A_2) + \dots P(B|A_n)P(A_n)$$

Fórmula de Bayes

$$P(A_j|B) = \frac{P(A_j\cap B)}{P(B)} \dashrightarrow \frac{P(B\cap A_j)}{\sum_{i=1}^{n} P(A_i)\cdot P(B|A_i)}$$

$$¿P(C|N)? = \frac{P(C\cap N)}{P(N)}$$
$$= \frac{P(C)\cdot P(N|C)}{P(C)P(N|C) + P(x)\cdot P(N|x)}$$
$$= \frac{\frac{1}{2}\cdot\frac{1}{3}}{13/24} = 4/13$$

En el problema de las monedas y las urnas, ¿cuál es la probabilidad de que haya salido cara la moneda si la bola extraída ha sido negra? ¿y si ha sido blanca?

Tema 2: PROBABILIDAD

[Problema de las bolas → en R]

```r
Existen dos urnas, en la primera: 2 bolas blancas, 1 negra.
En la segunda: 3 bolas negras, 1 blanca
se lanza moneda, si cara cogemos la primera urna.
¿Probabilidad negra?
¿Probabilidad de cara si ha salido negra?
```{r}

###PROBABILIDAD NEGRA
  # 1) Generar la funcion q simule cualquier caso
  sim_game = function(){
    urna_c = c("B", "B", "N")
    urna_x = c("B", rep("N", 3))

    #tirar moneda al aire
   moneda =  sample(c("X", "C"), 1)
    #sale cara --> urna c sino x
   if(moneda == "C") {
     bola = sample(urna_c, 1)
   } else {
     bola = sample(urna_x, 1)}
     bola
  }
bola = sim_game()

  # 2)Comprobar si es un caso de éxito
  bola == "N"  #éxito

    #hacerlo muchas veces--> replicate
    N = 5000
   sims = replicate(N, {
      bola = sim_game()
      bola == "N"
    })

  # 3) Aplicar LaPlace
    print(
      sum(sims) / N
    )
```

```r
#TRUE FALSE --> FALSE ¿Como lo hacemos?
  # cond1 = game[1] == "C" & game[2] == "N"
  # NUNCA --> cond1 = game == c("C", "N") -----> all(game == c("C", "N"))
```

```r
###PROBABILIDAD DE CARA SI HA SALIDO NEGRA
    #dos condiciones de éxito
      #--> asociada al numerador (C y N )
      # --> asociada al denominador (N)

    sim_game2 = function(){
    urna_c = c("B", "B", "N")
    urna_x = c("B", rep("N", 3))

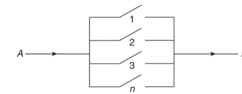
    #tirar moneda al aire
   moneda =  sample(c("X", "C"), 1)
    #sale cara --> urna c sino x
   if(moneda == "C") {
     bola = sample(urna_c, 1)
    }else {
     bola = sample(urna_x, 1)}
    c(moneda, bola)
  }
    # 1) Simular Juegos
    game = sim_game2()
    # 2) Dos condiciones de éxito
    cond1 = game[1] == "C" & game[2] == "N" #numerador
    cond2 = game[2] =="N" #denominador
    c(cond1,cond2)
    N=60
    sims2 = replicate(N, {
      game = sim_game2()
      cond1 = game[1] == "C" & game[2] == "N"
    cond2 = game[2] =="N"
    c(cond1,cond2)
    })
    #sims2--> matriz ::= num      num    num      num
    #                    denom   denom  denom    denom

    #SOLO QUEREMOS TRUE/TRUE
      #Calcular numero éxitos en el numerador
    sum(sims2[1, ])
      #Calcular numero éxitos en el denominador
    sum(sims2[2, ])
    print(
      sum(sims2[1, ]) / sum(sims2[2, ])
    )
```

Independencia de Sucesos

A y B son independientes $\dashrightarrow$ $\boxed{P(B|A) = P(B) \;\blacktriangleright\blacktriangleright\; P(A\cap B) = P(A)\cdot P(B)}$

Un sistema se dice en paralelo si funciona cuando al menos uno de sus componentes funciona.



$P(\text{circuito funcione}) = P\left(\begin{smallmatrix}\text{al menos 1}\\ \text{rama funcione}\end{smallmatrix}\right)$

$= P(\text{funcione } R_1) + P(R_2) + P(R_3) \dots$

ó

$= 1 - P(\text{todas fallen})$

Si la probabilidad de que el componente i-ésimo funcione es p_i, ¿Cuál es la probabilidad de que el sistema funcione?

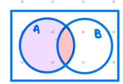
$P(\text{todas fallen}) = P(F_1 \cap F_2 \cap F_3 \dots) \xrightarrow[\text{indep}]{\text{sucesos}} P(F_1)\cdot P(F_2)\cdot P(F_3)\dots = \prod_{i=1}^{n} P(F_i) = \prod_{i=1}^{n} (1-p_i)$

$P(\text{circuito funcione}) = 1 - \prod_{i=1}^{n} (1-p_i)$ //

Demuestra que si A y B son sucesos independientes, también los son A y B^c.

Tenemos $\qquad$ **Buscamos**

$P(A\cap B) = P(A)\cdot P(B) \rightsquigarrow P(A\cap \bar{B}) = P(A)\cdot P(B)$



3 axioma K >>> Si $A\cap B = \emptyset \to P(A\cup B) = P(A) + P(B)$

$P(A) = P\left((A\cap B) \cup (A\cap \bar{B})\right)$ 3 K.

$= P(A\cap B) + P(A\cap \bar{B})$ suc. indep

$= P(A)\cdot P(B) + P(A\cap \bar{B})$

$P(A) = P(A)\cdot P(B) + P(A\cap \bar{B})$

$P(A) - [P(A)\cdot P(B)] = P(A\cap \bar{B})$

$= P(A)\cdot[1 - P(B)] = P(A\cap \bar{B})$

$= P(A)\cdot P(\bar{B}) = P(A\cap \bar{B})$ c.q.d //

Se lanza una moneda y un dado. ¿Son independientes los sucesos de obtener cara con la moneda y obtener más de 2 con el dado?

$\Omega = \{(c,1), (c,2) \dots (c,6), (x,1), (x,2), \dots (x,6)\}$

$\hookrightarrow C = \{(c,1) \dots (c,6)\}$

$\hookrightarrow S_{>2} = \{(c,3) \dots (c,6), (x,3), \dots (x,6)\}$

LaPlace

C y $S_{>2}$ independientes: Probar $P(C\cap S_{>2}) = P(C)\cdot P(S_{>2})$

$P(C\cap S_{>2}) = \dfrac{4 \to \text{nº casos fav}}{\Omega = 12 \to \text{nº casos totales}} = \dfrac{1}{6}$ //

$P(C)\cdot P(S_{>2}) = [6/12]\cdot[8/12] = \frac{1}{2}\cdot\frac{2}{3} = 1/6$ //

= c.q.d
```