

# Utilidad esperada

## Problema 1:

Realiza una simulación para calcular la esperanza de hijos por pareja en un país. Supón que las probabilidades de tener 0, 1, 2, 3, y 4 hijos son 10%, 25%, 35%, 20%, y 10% respectivamente.

```
# Parámetros
probabilidades <- c(0.10, 0.25, 0.35, 0.20, 0.10)
hijos <- c(0, 1, 2, 3, 4)
num_simulaciones <- 10000

# Simulación
set.seed(2014)
simulaciones <- sample(hijos, num_simulaciones, replace = TRUE, prob = probabilidades)

# Calcular la esperanza
esperanza <- mean(simulaciones)
print(paste("La esperanza de hijos por pareja es:", esperanza))
```

```
## [1] "La esperanza de hijos por pareja es: 1.9534"
```

## Problema 2:

Simula el valor esperado de acciones antes del anuncio de ganancias trimestrales. Si el anuncio es favorable, las acciones se venden a \$100 cada una, pero si es negativo, se venden a \$10. La probabilidad de un buen informe es del 40%.

```
# Parámetros
precio_favorable <- 100
precio_negativo <- 10
prob_favorable <- 0.40
prob_negativo <- 0.60
num_simulaciones <- 10000

# Simulación
set.seed(2014)
resultados <- sample(c(precio_favorable, precio_negativo),
                     num_simulaciones, replace = TRUE,
                     prob = c(prob_favorable, prob_negativo))

# Calcular el valor esperado
valor_esperado <- mean(resultados)
print(paste("El valor esperado de las acciones es:", valor_esperado))
```

```
## [1] "El valor esperado de las acciones es: 45.406"
```

## Problema 3:

Simula la utilidad esperada de llevar un paraguas considerando dos estados del mundo: lluvioso (30%) y soleado (70%). Llevar el paraguas tiene una utilidad de 10 en caso de lluvia y -1 en caso de sol. No llevarlo

tiene una utilidad de -10 en caso de lluvia y 0 en caso de sol.

```
# Parámetros
utilidades <- list(
  llevar = c(lluvia = 10, sol = -1),
  no_llevar = c(lluvia = -10, sol = 0)
)
probabilidades <- c(lluvia = 0.30, sol = 0.70)
num_simulaciones <- 10000

# Simulación
set.seed(123)
estados <-
  sample(names(probabilidades), num_simulaciones, replace = TRUE, prob = probabilidades)

# Calcular la utilidad esperada
utilidad_llevar <- mean(sapply(estados, function(x) utilidades$llevar[x]))
utilidad_no_llevar <- mean(sapply(estados, function(x) utilidades$no_llevar[x]))

print(paste("Utilidad esperada de llevar el paraguas:", utilidad_llevar))

## [1] "Utilidad esperada de llevar el paraguas: 2.2472"
print(paste("Utilidad esperada de no llevar el paraguas:", utilidad_no_llevar))

## [1] "Utilidad esperada de no llevar el paraguas: -2.952"
```

#### Problema 4:

Simula la utilidad esperada de Boris al comer un malvavisco, considerando dos hipótesis causales: con deficiencia de vitaminas (33.3%) y sin deficiencia (66.7%). Los valores de utilidad y probabilidades se deben ajustar según el caso presentado en el problema de Boris.

```
# Parámetros
probabilidades <- list(
  V = 1/3,
  no_V = 2/3
)
utilidades <- list(
  ME = -50, # Malvaviscos con deficiencia de vitaminas
  no_ME = 10 # No hay calambres
)
probabilidades_condicionales <- list(
  V = list(M = 0.75, no_M = 0.25),
  no_V = list(M = 0.12, no_M = 0.88)
)
num_simulaciones <- 10000

# Simulación
set.seed(2014)
resultados <- replicate(num_simulaciones, {
  if (runif(1) < probabilidades$V) {
    estado <- "V"
  } else {
    estado <- "no_V"
  }
})
```

```

    if (runif(1) < probabilidades_condicionales[[estado]]$M) {
      utilidad <- utilidades$ME
    } else {
      utilidad <- utilidades$no_ME
    }

    return(utilidad)
  })

# Calcular la utilidad esperada
utilidad_esperada <- mean(resultados)
print(paste("La utilidad esperada de comer un malvavisco es:", utilidad_esperada))

## [1] "La utilidad esperada de comer un malvavisco es: -9.962"

```

### Problema 5

Simula el Brier Score para un conjunto de proposiciones. Considera que se tienen 10 proposiciones con diferentes grados de creencia y resultados verdaderos/falsos.

```

# Parámetros
proposiciones <- 10
grados_de_creencia <- runif(proposiciones)
valores_de_verdad <- sample(c(0, 1), proposiciones, replace = TRUE)
num_simulaciones <- 10000

# Calcular el Brier Score
brier_score <- sum((valores_de_verdad - grados_de_creencia)^2) / proposiciones
print(paste("El Brier Score es:", brier_score))

## [1] "El Brier Score es: 0.537574305443425"

```

### Problema 6

Simula la toma de decisiones estratégicas en una empresa bajo incertidumbre. Supongamos que la empresa puede elegir entre tres proyectos (A, B, y C), cada uno con diferentes probabilidades de éxito y diferentes utilidades esperadas en función del éxito o fracaso.

```

# Parámetros
set.seed(2015)

num_simulaciones <- 10000

# Definir proyectos y sus parámetros
proyectos <- c("A", "B", "C")
prob_exito <- c(A = 0.6, B = 0.4, C = 0.3)
utilidades_exito <- c(A = 100, B = 150, C = 200)
utilidades_fracaso <- c(A = -20, B = -50, C = -100)

# Simulación de los resultados de los proyectos
simulacion_resultados <- data.frame(
  A = rbinom(num_simulaciones, 1, prob_exito["A"]) * utilidades_exito["A"] +
    (1 - rbinom(num_simulaciones, 1, prob_exito["A"])) * utilidades_fracaso["A"],
  B = rbinom(num_simulaciones, 1, prob_exito["B"]) * utilidades_exito["B"] +
    (1 - rbinom(num_simulaciones, 1, prob_exito["B"])) * utilidades_fracaso["B"],

```

```

C = rbinom(num_simulaciones, 1, prob_exito["C"]) * utilidades_exito["C"] +
  (1 - rbinom(num_simulaciones, 1, prob_exito["C"])) * utilidades_fracaso["C"]
)

# Calcular la utilidad esperada de cada proyecto
resultados <- data.frame(
  Proyecto = proyectos,
  UtilidadEsperada = sapply(simulacion_resultados, mean),
  Riesgo = sapply(simulacion_resultados, sd)
)

print("Resultados de la simulación de planificación estratégica:")

## [1] "Resultados de la simulación de planificación estratégica:"
print(resultados)

```

```

## Proyecto UtilidadEsperada Riesgo
## A A 53.05 49.78499
## B B 30.08 77.48415
## C C -12.33 101.01481

```

## Problema 7

Simula la toma de decisiones de inversión en un portafolio de acciones considerando diferentes estrategias de inversión y sus impactos en el valor esperado del portafolio. Supongamos que hay tres estrategias de inversión: conservadora, balanceada y agresiva, cada una con diferentes distribuciones de retorno y riesgo.

```

# Parámetros
set.seed(2014)

num_simulaciones <- 10000

# Definir estrategias de inversión
estrategias <- c("conservadora", "balanceada", "agresiva")
retornos <- list(
  conservadora = rnorm(num_simulaciones, mean = 0.05, sd = 0.02),
  balanceada = rnorm(num_simulaciones, mean = 0.08, sd = 0.05),
  agresiva = rnorm(num_simulaciones, mean = 0.12, sd = 0.10)
)

# Simulación de los retornos
simulacion_resultados <- data.frame(
  conservadora = retornos$conservadora,
  balanceada = retornos$balanceada,
  agresiva = retornos$agresiva
)

# Calcular el valor esperado y el riesgo (desviación estándar) de cada estrategia
resultados <- data.frame(
  Estrategia = estrategias,
  ValorEsperado = sapply(simulacion_resultados, mean),
  Riesgo = sapply(simulacion_resultados, sd)
)

```

```
print("Resultados de la simulación de estrategias de inversión:")
```

```
## [1] "Resultados de la simulación de estrategias de inversión:"
```

```
print(resultados)
```

```
##          Estrategia ValorEsperado      Riesgo
## conservadora conservadora  0.04993734 0.02034537
## balanceada   balanceada   0.07938660 0.05059832
## agresiva     agresiva     0.12103768 0.09963815
```