

```

#GUIA 17

simulIntProp <- function(m=5, n=1, p, nivel.conf=0.95)
{
  X <- rbinom(m, n, p)
  # Matriz con 1000 valores aleatorios binomial(n,p), 50 muestras cada una de tamaño 20
  pe <- X/n
  # Calcula la proporción estimada en cada una de las muestras.
  SE <- sqrt(pe*(1-pe)/n)
  # Calcula la desviación estándar estimada en cada una de las muestras.
  alfa <- 1-nivel.conf
  z <- qnorm(1-alfa/2)
  Intervalo <- cbind(pe - z*SE, pe + z*SE)
  # genera los extremos del intervalo de confianza
  nInter <- 0
  # un contador para conocer en cuántos intervalos se encuentra la verdadera proporción.
  for(i in 1:m)
  if ((p >= Intervalo[i, 1]) && (p <= Intervalo[i, 2]))
  nInter <- nInter + 1
  # función que cuenta cuántos intervalos contienen el verdadero valor del parámetro.
  return(nInter)
}
n=20; m= 50; p=0.5; nivel.conf=0.95
simulIntProp(m, n, p, nivel.conf)

## [1] 46

Intervalo # para visualizar cada uno de los intervalos generados

##           [,1]      [,2]
## [1,] 0.09916346 0.5008365
## [2,] 0.49916346 0.9008365
## [3,] 0.18529670 0.6147033
## [4,] 0.33196777 0.7680322
## [5,] 0.56022730 0.9397727
## [6,] 0.23196777 0.6680322
## [7,] 0.23196777 0.6680322
## [8,] 0.09916346 0.5008365
## [9,] 0.06022730 0.4397727
## [10,] 0.44096270 0.8590373
## [11,] 0.23196777 0.6680322
## [12,] 0.44096270 0.8590373
## [13,] 0.33196777 0.7680322
## [14,] 0.49916346 0.9008365
## [15,] 0.09916346 0.5008365
## [16,] 0.28086936 0.7191306

```

```

## [17,] 0.28086936 0.7191306
## [18,] 0.18529670 0.6147033
## [19,] 0.44096270 0.8590373
## [20,] 0.33196777 0.7680322
## [21,] 0.28086936 0.7191306
## [22,] 0.44096270 0.8590373
## [23,] 0.02469549 0.3753045
## [24,] 0.38529670 0.8147033
## [25,] 0.28086936 0.7191306
## [26,] 0.14096270 0.5590373
## [27,] 0.02469549 0.3753045
## [28,] 0.18529670 0.6147033
## [29,] 0.33196777 0.7680322
## [30,] 0.23196777 0.6680322
## [31,] 0.23196777 0.6680322
## [32,] 0.23196777 0.6680322
## [33,] 0.23196777 0.6680322
## [34,] 0.28086936 0.7191306
## [35,] 0.44096270 0.8590373
## [36,] 0.33196777 0.7680322
## [37,] 0.28086936 0.7191306
## [38,] 0.38529670 0.8147033
## [39,] 0.44096270 0.8590373
## [40,] 0.14096270 0.5590373
## [41,] 0.23196777 0.6680322
## [42,] 0.49916346 0.9008365
## [43,] 0.44096270 0.8590373
## [44,] 0.09916346 0.5008365
## [45,] 0.33196777 0.7680322
## [46,] 0.28086936 0.7191306
## [47,] 0.38529670 0.8147033
## [48,] 0.14096270 0.5590373
## [49,] 0.38529670 0.8147033
## [50,] 0.23196777 0.6680322

nInter

## [1] 46

# para visualizar en cuantos de estos intervalos se encuentra la verdadera proporcin.
#Grfico que muestra los intervalos de confianza de 95% que contienen y no contienen el verdadero
matplot(rbind(pe - z*SE, pe + z*SE), rbind(1:m, 1:m), type="l", lty=1)
abline(v=p)

```

