

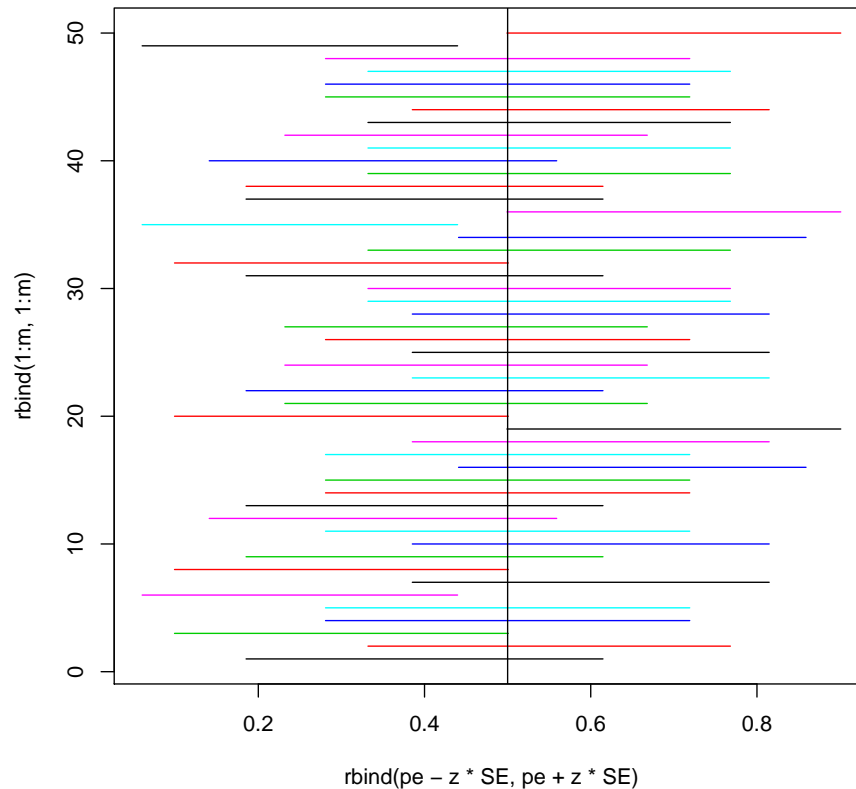
GuÃas prÃcticas

Erika MartÃnez

November 27, 2015

UNIDAD 4: Practica 17 - Inferencia estadística, Estimación.
SIMULACION DEL CONCEPTO DE INTERVALO DE CONFIANZA PARA
ESTIMAR UN PARAMETRO. Ejemplo 1.

```
#Sea la variable aleatoria X = el número de caras obtenidas, al lanzar una moneda  
#balanceada 20 veces. Simulamos 50 muestras para generar intervalos de 95% de  
#confianza y así poder estimar la proporción verdadera de caras (p), y encontrar  
#en cuántos de estos intervalos se encuentra el verdadero valor de la proporción  
  
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] 47  
  
#Gráfico que muestra los intervalos de confianza de 95% que contienen y no contienen el ver  
#valor del parámetro p.  
matplot(cbind(pe - z*SE, pe + z*SE), cbind(1:m, 1:m), type="l", lty=1)  
abline(v=p)
```



Ejercicio 1.

#Sea la variable aleatoria X = el número que se obtiene al lanzar un dado no cargado
 #30 veces. Simular 56 muestras para generar intervalos de 95% de confianza para
 #estimar el promedio (p), y encontrar cuántos de estos intervalos contiene el valor
 #medio verdadero.

```
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 cuantos intervalos contienen el verdadero valor del parámetro.
  return(nInter)
}
n=30; m= 56; p=0.5; nivel.conf=0.95
simulIntProp(m, n, p, nivel.conf)

## [1] 54

```