UNIVERSIDAD DE EL SALVADOR

FACULTAD MULTIDISCIPLINARIA DE OCCIDENTE

DEPARTAMENTO DE MATEMÁTICA

Licenciatura en Estadística

Control Estadistico del Paquete R

"UNIDAD CUATRO"

Alumna:
Erika Beatríz Guillén Pineda

Fecha de elaboración
Santa Ana - 27 de noviembre de 2015

1. INTRODUCCIÓN

La Inferencia Estadística es: El conjunto de métodos estadísticos que permiten deducir (inferir) como se distribuye (comporta) la población en estudio o las relaciones estocásticas entre varias variables de interés a partir de la información suministrada por una muestra aleatoria.

La Inferencia Estadística paramétrica plantea tres tipos de problemas:

- Estimación puntual: en la que pretendemos dar un valor puntual del parámetro.
- Estimación por intervalos: en elque buscamos un intervalo en el que confiamos se encuentre el verdadero valor de teta desconocido.
- Contraste de hipótesis: donde buscamos probar una declaración o un supuesto acerca del valor de uno (o más) parámetro(s) teta.

Para llevar a cabo lo anterior, se parte del supuesto de que la distribución de la(s) característica(s) que se está estudiando pertenece a una familia conocida de distribuciones, siendo únicamente desconocidos los parámetros que la definen. Por lo regular pertenecen a la familia normal o a cualquiera que pueda obtenerse a partir de ella como lo es: la t de Student, la Chi-Cuadrado o la F de Snedecor.

2. ESTIMACIÓN PUNTUAL

Un estadístico teta estimado es igual a f(X_1, X_2, X_3,...,X_n) es un estimador adecuado de un parámetro teta, si cumple las siguientes propiedades:

- **Insesgadez:** si la esperanza matemática del estimador coincide con el valor del parámetro al cual está intentado estimar E[teta estimado]=teta. Es decir, la distribución de probabilidad del estimador se concentra alrededor del valor que intenta predecir.
- **Consistencia:** si el estimador converge en probabilidad al valor del parámetro que está intentado estimar conforme crece el tamaño de la muestra. Es decir si $\tilde{teta}_n$ representa el estimador para una muestra de tamaño n, entonces se dice que $\tilde{teta}$ es consistente si: el limite de n cuando tiende a infinito de E[teta estimado]=teta.
- **Eficiencia:** si entre todos los posibles estimadores (insesgados o no) que pueden obtenerse es el que tenga la menor varianza posible.

Se verifica fácilmente que la media muestral (estimador de la media poblacional) cumple estas tres y aún más propiedades.

3. ESTIMACIÓN POR INTERVALOS DE CONFIANZA.

La idea de la estimación por intervalos de confianza radica en encontrar dos números reales, digamos $\tilde{t}eta_1$ y $\tilde{t}eta_2$, tales que el parámetro desconocido *teta* que se quiere estimar pertenezca al intervalo formado por dichos valores con probabilidad alta, digamos $1 - a$. Es decir;

$$\mathrm{P}[\tilde{t}eta_1 <= teta <= \tilde{t}eta_2] = 1 - a$$

Donde $\tilde{t}eta_1$ y $\tilde{t}eta_2$ sean valores que dependan únicamente del estimador $\tilde{t}eta$ y de los valores observados en la muestra X_1, X_2, X_3,...,X_n.

Se verifica fácilmente que cuando la característica de interés X sigue una distribución conocida la cual es simétrica (como la normal o la binomial o sus derivadas), y además los estimadores son insesgados los mejores intervalos, en el sentido de su anchura, son los intervalos simétricos alrededor del parámetro a estimar.

Hay que tener en cuenta que $1 - a$: es la probabilidad de que parámetro se encuentre en el intervalo antes de extraer la muestra. Una vez seleccionada la muestra esta probabilidad es 1 ó 0, dependiendo de si el parámetro se encuentra o no en el intervalo. En este sentido es que no se habla de probabilidad sino de confianza.

El concepto de confianza puede interpretarse de la siguiente manera: si se repitiera el experimento muestral (se tomarán muchas muestras) muchas veces, en aproximadamente el 100(1-a) % de los casos se confiaría que los intervalos de confianza encontrados contengan al verdadero valor del parámetro *teta* a estimar.

4. SIMULACIÓN DEL CONCEPTO DE INTERVALO DE CONFIANZA PARA ESTIMAR UN PARÁMETRO.

- 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.

Entonces X tiene una distribución binomial con párametros n=20 y p=0.5.

La función para generar cada una de las muestras, junto con los límites inferior y superior de los intervalos de confianza se muestra en seguida y le hemos llamado "simulIntProp".

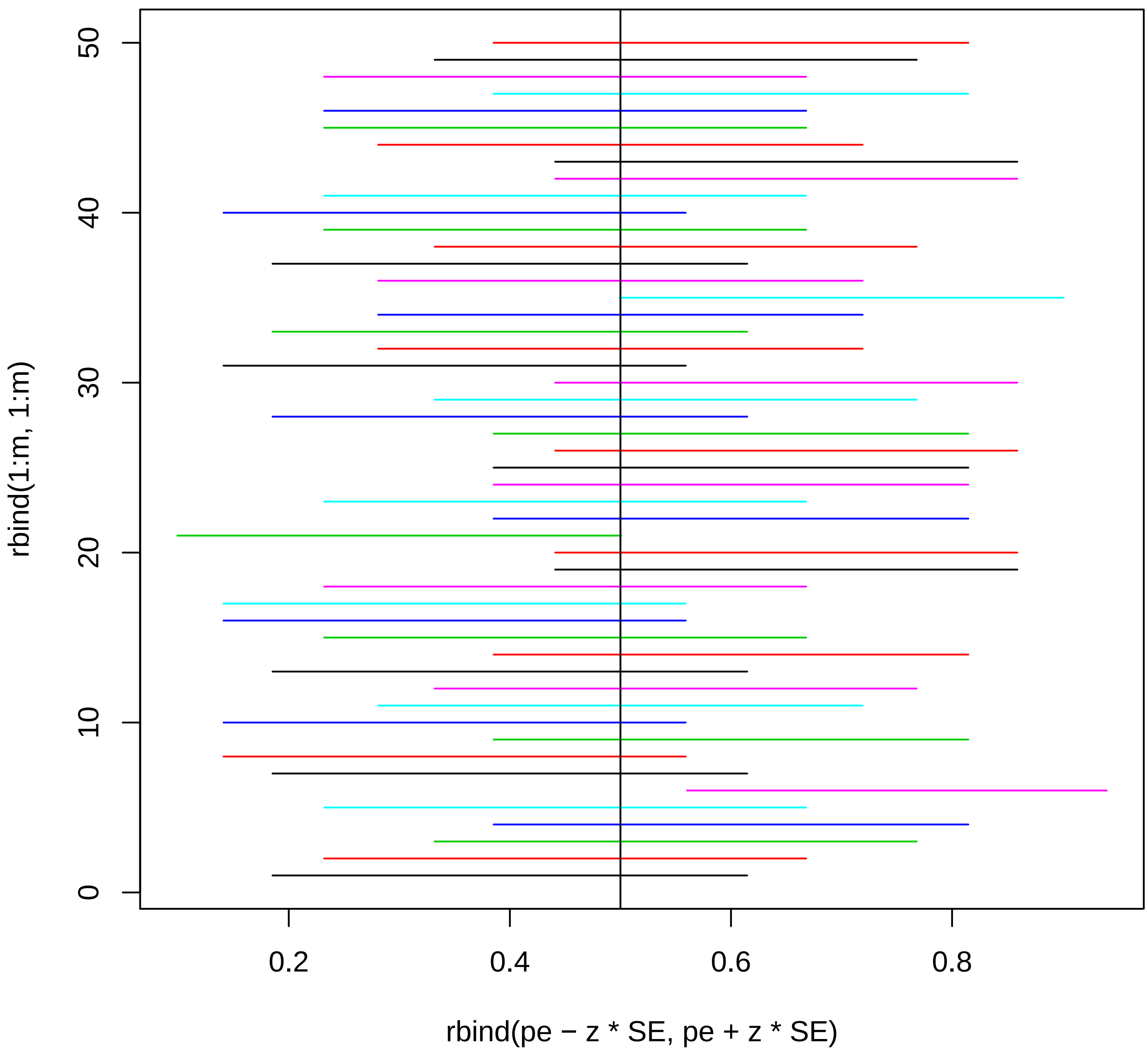
```
simulIntProp <- function(m=5, n=1, p, nivel.conf=0.95)
{
X <- rbinom(m, n, p)
# Matriz con 1000 valores aleatorios binomial(n,p), 50 muestras cada una de tamaÃ±o 20
pe <<- X/n
```

```
 # Calcula la proporciÃ³n estimada en cada una de las muestras.
SE <<- sqrt(pe*(1-pe)/n)
# Calcula la desviaciÃ³n estÃ!‘ndar 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 cuÃ¡ntos intervalos se encuentra la verdadera proporciÃ³n.
for(i in 1:m)
if ((p >= Intervalo[i, 1]) && (p <= Intervalo[i, 2]))
nInter <<- nInter + 1
# funciÃ³n que cuenta cuÃ!‘ntos intervalos contienen el verdadero valor del parÃ!‘metro.
return(nInter)
}
n=20; m= 50; p=0.5; nivel.conf=0.95
simulIntProp(m, n, p, nivel.conf)




Intervalo # para visualizar cada uno de los intervalos generados
```

```
## [24,] 0.38529670 0.8147033
## [25,] 0.38529670 0.8147033
## [26,] 0.44096270 0.8590373
## [27,] 0.38529670 0.8147033
## [28,] 0.18529670 0.6147033
## [29,] 0.33196777 0.7680322
## [30,] 0.44096270 0.8590373
## [31,] 0.14096270 0.5590373
## [32,] 0.28086936 0.7191306
## [33,] 0.18529670 0.6147033
## [34,] 0.28086936 0.7191306
## [35,] 0.49916346 0.9008365
## [36,] 0.28086936 0.7191306
## [37,] 0.18529670 0.6147033
## [38,] 0.33196777 0.7680322
## [39,] 0.23196777 0.6680322
## [40,] 0.14096270 0.5590373
## [41,] 0.23196777 0.6680322
## [42,] 0.44096270 0.8590373
## [43,] 0.44096270 0.8590373
## [44,] 0.28086936 0.7191306
## [45,] 0.23196777 0.6680322
## [46,] 0.23196777 0.6680322
## [47,] 0.38529670 0.8147033
## [48,] 0.23196777 0.6680322
## [49,] 0.33196777 0.7680322
## [50,] 0.38529670 0.8147033

nInter # para visualizar en cuÃ¡ntos de estos intervalos se encuentra la

## [1] 49

# verdadera proporciÃ³n.
```

Gráfico que muestra los intervalos de confianza de 95 % que contienen y no contienen el verdadero valor del parámetro p.

```
matplot(rbind(pe - z*SE, pe + z*SE), rbind(1:m, 1:m), type="l", lty=1)
abline(v=p)
```



0
10
20
30
40
50
rbind(1:m, 1:m)
0.2
0.4
0.6
0.8
rbind(pe − z * SE, pe + z * SE)

- **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 (μ), 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), 56 muestras cada una de tamaÃ±o 30
pe <<- X/n
 # Calcula la proporciÃ³n estimada en cada una de las muestras.
SE <<- sqrt(pe*(1-pe)/n)
# Calcula la desviaciÃ³n estÃ!‘ndarestimada 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 cuÃ¡ntos intervalos se encuentra la
#verdadera proporciÃ³n.
for(i in 1:m)
if ((p >= Intervalo[i, 1]) && (p <= Intervalo[i, 2]))
nInter <<- nInter + 1
# funciÃ³n que cuenta cuÃ!‘ntos intervalos contienen el verdadero valor del parÃ!‘metro.
return(nInter)
}
n=30; m= 56; p=0.17; nivel.conf=0.95
simulIntProp(m, n, p, nivel.conf)


## [1] 54

Intervalo # para visualizar cada uno de los intervalos generados

##               [,1]       [,2]
##  [1,]  0.033308009 0.30002532
##  [2,] -0.007351649 0.20735165
##  [3,]  0.081984476 0.38468219
##  [4,]  0.011691522 0.25497514
##  [5,]  0.033308009 0.30002532
##  [6,]  0.108424383 0.42490895
##  [7,]  0.011691522 0.25497514
##  [8,] -0.030900702 0.09756737
##  [9,]  0.011691522 0.25497514
## [10,] -0.007351649 0.20735165
## [11,]  0.011691522 0.25497514
```

```
## [12,]  0.011691522 0.25497514
## [13,]  0.056864469 0.34313553
## [14,] -0.007351649 0.20735165
## [15,]  0.081984476 0.38468219
## [16,]  0.136017648 0.46398235
## [17,]  0.164646492 0.50202017
## [18,]  0.011691522 0.25497514
## [19,]  0.056864469 0.34313553
## [20,]  0.033308009 0.30002532
## [21,]  0.164646492 0.50202017
## [22,]  0.011691522 0.25497514
## [23,]  0.011691522 0.25497514
## [24,]  0.056864469 0.34313553
## [25,] -0.022594020 0.15592735
## [26,]  0.011691522 0.25497514
## [27,] -0.007351649 0.20735165
## [28,] -0.007351649 0.20735165
## [29,] -0.007351649 0.20735165
## [30,]  0.011691522 0.25497514
## [31,]  0.033308009 0.30002532
## [32,]  0.011691522 0.25497514
## [33,]  0.056864469 0.34313553
## [34,]  0.033308009 0.30002532
## [35,]  0.033308009 0.30002532
## [36,]  0.081984476 0.38468219
## [37,]  0.033308009 0.30002532
## [38,]  0.056864469 0.34313553
## [39,] -0.007351649 0.20735165
## [40,] -0.007351649 0.20735165
## [41,]  0.056864469 0.34313553
## [42,] -0.007351649 0.20735165
## [43,]  0.011691522 0.25497514
## [44,]  0.108424383 0.42490895
## [45,]  0.011691522 0.25497514
## [46,]  0.011691522 0.25497514
## [47,] -0.007351649 0.20735165
## [48,]  0.011691522 0.25497514
## [49,]  0.081984476 0.38468219
## [50,]  0.011691522 0.25497514
## [51,]  0.011691522 0.25497514
## [52,]  0.011691522 0.25497514
## [53,]  0.081984476 0.38468219
## [54,]  0.081984476 0.38468219
## [55,] -0.007351649 0.20735165
## [56,]  0.033308009 0.30002532
```

```
nInter # para visualizar en cuÃ¡ntos de estos intervalos se encuentra la
```

```
## [1] 54

# verdadero valor medio.
```