

Principios de Estadística

Leonardo Collado Torres y María Gutiérrez Arcelus

Licenciatura en Ciencias Genómicas, UNAM

www.lcg.unam.mx/~lcollado/index.php

www.lcg.unam.mx/~mgutierr/index.php

Cuernavaca, México
Febrero - Junio, 2009

Muestreo de poblaciones

Principios de
Estadística

Intro

Problema

Solución

1 Intro

2 Problema

3 Solución

Descripción de la actividad

- Hoy regresaremos al problema que les describí en "letras.pdf" y aprenderemos a usar la función `sample`.
- La sesión de hoy será mas bien práctica y de nuevo, tendrán que entregar un script comentado en la página de Cursos¹.

¹Si termina la gran mayoría en clase, mejor!

sample

- Primero que nada, veamos los argumentos de la función **sample**.

```
> args(sample)
```

```
function (x, size, replace = FALSE, prob = NULL)  
NULL
```

- *x* es la población de la cual queremos obtener una muestra. Generalmente es un vector.
- *size* es el tamaño de nuestra muestra.
- *replace* lo usamos para hacer un muestro con o sin reemplazo.
 - ▶ ¿Qué creen que sea eso?
- *prob* es un vector con las probabilidades para cada elemento de nuestra población.

Example (Califs)

Imaginemos que sus calificaciones son al azar y queremos obtener una muestra de como les va a ir en el próximo examen. La población es de 0 hasta 10 y el tamaño de nuestra muestra es de 20. Encontramos la calificación promedio.

- 1 ¿Qué pasa si las probabilidades son iguales?
- 2 ¿Y si nuestro vector prob es igual a $(.005, .008, .012, .015, .05, .1, .126, .129, .25, .18, .125)$?

- Por sencillez, guardaremos los resultados en los objetos x y y .

```
> x <- sample(x = 0:10, size = 20,  
+           replace = T)  
> y <- sample(x = 0:10, size = 20,  
+           replace = T, prob = c(0.005,  
+           0.008, 0.012, 0.015, 0.05,  
+           0.1, 0.126, 0.129, 0.25,  
+           0.18, 0.125))  
  
> table(x)  
> table(y)
```

Nuestros resultados

- Veamos los datos en formato de tabla.

	0	1	3	4	5	7	8	9	10	
1	1	1	3	2	1	2	1	4	3	3

Table 1: Califs 1er caso

	1	4	5	6	7	8	9	10	
1	1	1	1	3	1	1	4	7	2

Table 2: Califs 2ndo caso

- ¿Salió lo que esperabamos dado nuestras probabilidades?

```
> mean(x)
```

```
[1] 5.95
```

```
> mean(y)
```

```
[1] 7.4
```


La simulación inicial

- Primero que nada, lo siguiente es una simulación sencilla para una población de tamaño 50. Veamos que pasa si asumimos la misma probabilidad para todos los elementos y con tamaños de muestra diferentes: desde 1 hasta 1000.

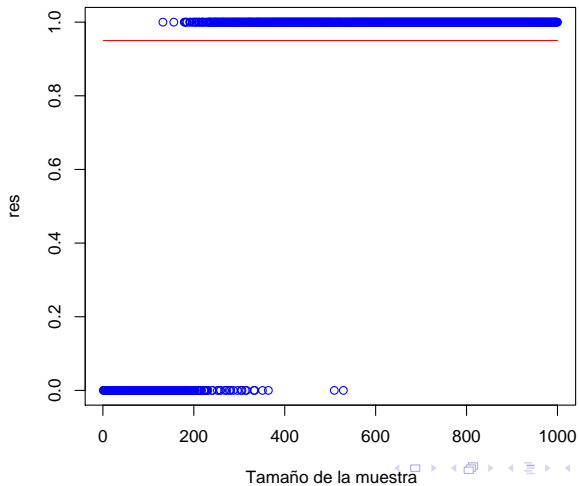
```
> pob <- 1:50
> res <- NULL
> for (i in 1:1000) {
+   muestra <- sample(pob, size = i,
+     replace = T)
+   prueba <- length(unique(muestra))/length(pob)
+   if (prueba >= 1) {
+     res <- c(res, 1)
+   }
+   else {
+     res <- c(res, 0)
+   }
+ }
```

- Si gráficamos nuestros resultados, veremos algo con dos estados.
- Como extra, podemos usar `lines` para marcar en el eje Y el 0.95

```
> plot(1:1000, res, col = "blue",  
+      main = "Una simulación", xlab = "Tamaño de la muestra")  
> lines(x = 1:1000, y = rep(0.95,  
+      1000), col = "red")
```

1era simulación

Una simulación



Problema

- Basándose en la simulación anterior, quiero que hagan 100 simulaciones para una población de tamaño 10 y con probabilidades iguales.
- Cada simulación debe explorar tamaños de muestra desde 1 hasta 100.
- Hagan una sola gráfica donde resuman sus resultados con la línea en el 0.95 en el eje Y .

- Almacenen sus resultados en una matriz ² en vez de en un vector nulo.
- Utilicen dos ciclos para llenar su matriz. Ya les di una **super** ayuda con los 0s y 1s.
- Como tienen que resumir sus 100 simulaciones en una gráfica, saquen la _ _ _ _ _ .

²Pueden llenarla con 0s o NA

- Usando los elementos que les dije en los "Tips", obtenemos esto:

```
> pob <- 1:10
> res <- matrix(0, 100, 100)
> for (i in 1:100) {
+   for (j in 1:100) {
+     muestra <- sample(pob,
+       size = j, replace = T)
+     prueba <- length(unique(muestra))/length(pob)
+     if (prueba >= 1) {
+       res[i, j] <- 1
+     }
+   }
+ }
```

- El código para la gráfica es casi el mismo :). Noten el uso de `colMeans`.

```
> plot(1:100, colMeans(res), col = "blue",  
+      main = "100 simulaciones",  
+      xlab = "Tamaño de la muestra",  
+      ylab = "Media de las 100 simulaciones")  
> lines(x = 1:100, y = rep(0.95,  
+      100), col = "red")
```

Gráfica final

100 simulaciones

