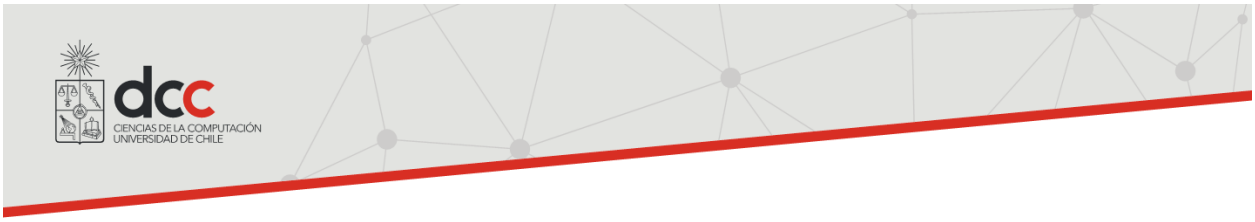
Tarea 1



Tarea 1: Foundations

CC6104: Statistical Thinking

Integrantes :

- Rafael De La Sotta
- Felipe Ortúzar

Cuerpo Docente:

- Profesor: Felipe Bravo M.
- Auxiliar: Sebastian Bustos e Ignacio Meza D.

Fecha límite de entrega:

Índice:

Objetivo

Bienvenid@s a la primera tarea del curso Statistical Thinking. Esta tarea tiene como objetivo evaluar los contenidos teóricos de la primera parte del curso, los cuales se enfocan principalmente en análisis exploratorio de datos y conceptos introductorios de probabilidades. Si aún no han visto las clases, se recomienda visitar los enlaces de las referencias.

La tarea consta de una parte teórica que busca evaluar conceptos vistos en clases. Seguido por una parte práctica con el fín de introducirlos a la programación en R enfocada en el análisis estadístico de datos.

Instrucciones:

- La tarea se realiza en grupos de **máximo 2 personas**. Pero no existe problema si usted desea hacerla de forma individual.
- La entrega es a través de u-cursos a más tardar el día estipulado en la misma plataforma. A las tareas atrasadas se les descontará un punto por día.

- El formato de entrega es este mismo **Rmarkdown** y un **html** con la tarea desarrollada. Por favor compruebe que todas las celdas han sido ejecutadas en el archivo html.
- Al momento de la revisión tu código será ejecutado. Por favor verifica que tu entrega no tenga errores de compilación.
- No serán revisadas tareas desarrolladas en Python.
- Está **PROHIBIDO** la copia o compartir las respuestas entre integrantes de diferentes grupos.
- Pueden realizar consultas de la tarea a través de U-cursos y/o del canal de Discord del curso.

Referencias:

Slides de las clases:

- Introduction to Statistical Thinking
- Introduction to R
- Descriptive Statistics
- Probability

Videos de las clases:

- Introduction to Statistical Thinking: video1 video2
- Introduction to R: video1 video2 video3 video4
- Descriptive Statistics: video1 video2 video3 video4
- Probability: video1 video2 video3 video4 video5 video6

Primera Parte: Preguntas Teóricas

A continuación, se presentaran diferentes preguntas que abordan las temáticas vistas en clases. Por favor responda cada una de estas preguntas de forma breve, no más de 4 o 5 lineas.

Pregunta 1: ¿Por qué la estadística es importante?, ¿Que nos permite realizar con los datos?. De algún ejemplo.

La estadística tiene como tema central la organización y análisis de datos. Con ella se puede describir, decidir y predecir a partir de estos. La estadística logra transformar los datos en información confiable sobre fenomenos, y esta información, debido a su confiabilidad, permite predecir, detectar tendencias y tomar desiciones.

Pregunta 2: Un amigue cercano a usted le comenta que le preocupa salir a la calle cuando hay ofertas en los helados, esto debido a que ha visto el siguiente titular en un famoso diario chileno: “El aumento en la compra de helados tiene una alta correlación con la muerte de personas en Santiago”. ¿Que le recomendaría a su amigue sobre el titular leído?, ¿Debería preocuparse tanto?.

Le diría que tuviera cuidado con mezclar los términos de correlación y causalidad. Es posible que la correlación se deba a que existen más muertes en el verano, que es la estación donde se vende más helado, por dar un ejemplo. Es decir, no porque exista correlación en dos variable signifique que haya una causalidad entre ellas, y por lo tanto lo que le recomendaría al amig" es que investigue mejor a la fuente que el diario cita y no lo que sale en el diario chileno, para que encuentre la razón de tal correlación y así deje probablemente de preocuparse.

Pregunta 3: Señale las diferentes aplicaciones que poseen las visualizaciones: Boxplot, histograma, gráfico de pie y scatterplot.

El Boxplot está hecho en base a los percentiles. Nos permite identificar la distribución en base a sus cuartiles, su simetría y sus outliers. A continuación se muestras sus aplicaciónes:

- Boxplot de solo una variable.
- Separa una variable en categorias y hacer boxplot de cada una de estas.

- Comparar distintos boxplot en un mismo gráfico.

El histograma muestra la distribución de valores de una variable. Se aplica para descomponer variables con un cierto orden, pues los datos son agrupados en "bins", permitiendo elegír la continuidad del gráfico. A continuación se muestras sus aplicaciónes:

- Histograma por cantidad.
- Histograma por densidad (áreas deben sumar 1).

El gráfico de pie es utilizado para mostrar frecuencia de clases en una variable. Normalmente se utiliza para variables categóricas. A continuación se muestras sus aplicaciónes:

- Dos dimensiones
- Tres dimensiones

Scatterplot compara dos variables numéricas en un plano carteciano, donde los valores de cada dato determinan su posición. A continuación se muestras sus aplicaciónes:

- Scatterplot entre dos variables numericas.
- Combinación de scatterplot entre un grupo de variables.
- Scatterplot de tres dimensiones.

Pregunta 4: Suponga que esta estudiando la diferencia en los sueldos de las personas que viven en Santiago y Rancagua. Suponiendo que los datos poseen outliers, ¿Que métrica de resumen utilizaría para comparar los datos?. Justifique su respuesta.

Respuesta Aquí

Probablemente utilizaría boxplot para aprender sobre estos datos en particular. La razón es que boxplot muestra la mediana, los rangos intercuartiles, el mínimo y el máximo que son buenas medidas para entender los datos. Pero más importante aún, boxplot permite visualizar de manera especial a los outliers de los datos, y por lo tanto es una herramienta que los señala directamente.

Usando scatter plots o histogramas también se podría estudiar el dataset, pero con respecto a los outliers se tendrían que identificar de manera visual e indirecta, con mayor dificultad en algunos casos. Por ejemplo en el histograma se podría ver cuando una barra está separada del resto de las barras.

Si el dataset contiene outliers que sólo son detectables mirando múltiples columnas, entonces es mejor tomar un rumbo más matemático/programación para obtener alguna lista o cantidad de outliers.(4 o más dimensiones). Con 3 dimensiones todavía se podría ocupar un scatter plot en 3D para visualizar la información y tratar de encontrar los outliers.

Pregunta 5: En base al mismo dataset de sueldos para las regiones de Santiago y Rancagua, le comentan que existe un error en los datos y que estos deben ser modificados aumentando un 10% el valor original y sumando 15.000 a cada uno de los datos. ¿Como se ve afectada la media, mediana y desviación estándar con esta modificación?. Explique a través de ecuaciones el cambio que experimentan las métricas de resumen respecto al valor original, considere para el caso de la media $\bar{X}_{old} = \frac{1}{m}\sum_{i=1}^{m} x_i$ y $sd_{old} = \sqrt{\frac{1}{(m-1)}\sum_{i=1}^{m}(x_i - \bar{x})^2}$ para la desviación estándar.

- Media

$\bar{X}_{new} = \frac{1}{m}\sum_{i=1}^{m}(x_i \cdot 0.1 + 15.000)$

$= \bar{X}_{old} \cdot 0.1 + 15.000 \cdot \frac{m}{m}$

$= \bar{X}_{old} \cdot 0.1 + 15.000$

- Desviación estándar

$$sd_{new} = \sqrt{\frac{1}{(m-1)} \sum_{i=1}^{m} (x_i \cdot 0.1 + 15.000 - \bar{x}_{new})^2}$$

$$= \sqrt{\frac{1}{(m-1)} \sum_{i=1}^{m} (x_i \cdot 0.1 + 15.000 - (\bar{x}_{old} \cdot 0.1 + 15.000)^2}$$

$$= \sqrt{\frac{1}{(m-1)} \sum_{i=1}^{m} (x_i \cdot 0.1 - \bar{x}_{old} \cdot 0.1)^2}$$

$$= \sqrt{\frac{1}{(m-1)} \sum_{i=1}^{m} (x_i - \bar{x}_{old})^2 \cdot 0.1^2}$$

$$= \sqrt{\frac{1}{(m-1)} \sum_{i=1}^{m} (x_i - \bar{x}_{old})^2} \cdot 0.1$$

$$= sd_{old} \cdot 0.1$$

- Mediana (n par)

$$M_{old} = \frac{x_{\frac{n}{2}} + x_{\frac{n}{2}+1}}{2}$$

$$M_{new} = \frac{x_{\frac{n}{2}} \cdot 0.1 + 15.000 + x_{\frac{n}{2}+1} \cdot 0.1 + 15.000}{2}$$

$$= \frac{x_{\frac{n}{2}} \cdot 0.1 + x_{\frac{n}{2}+1} \cdot 0.1}{2} + 15.000$$

$$= \frac{x_{\frac{n}{2}} + x_{\frac{n}{2}+1}}{2} \cdot 0.1 + 15.000$$

$$= M_{old} \cdot 0.1 + 15.000$$

- Mediana (n impar)

$$M_{old} = x_{\frac{n+1}{2}}$$

$$M_{new} = x_{\frac{n+1}{2}} \cdot 0.1 + 15.000$$

$$= M_{old} \cdot 0.1 + 15.000$$

- Mediana

$$M_{new} = M_{old} \cdot 0.1 + 15.000$$

Pregunta 6: Suponga que debe responder un examen sorpresa de 10 preguntas, con 5 alternativas por cada pregunta. ¿Cual es la probabilidad de obtener mas de 5 alternativas correctas si responde de forma aleatoria todo el examen?.

Nota: Puede resolver el ejercicio desarrollándolo a mano o utilizando código en R.

Respuesta Aquí

Calculando el binomio de Newton , podemos obtener de cuántas maneras sacamos más de 5 correctas

```
cant_list <- c(1, 10, 45, 120, 210)
win_prob <- 1/5
total <- 0
for (i in 0:4){
  total <- total + cant_list[i+1]*((win_prob**(10 - i))*((1- win_prob)**(i)))
}
print(total)
```

```
## [1] 0.006369382
```

Pregunta 7: Supongamos que el 10% de los alumnos del curso utilizan Macintosh, el 60% utiliza Windows y el 30% utiliza Linux. Supongamos que el 50% de los usuarios de Mac, el 78% de los usuarios de Windows y el 20% de los usuarios de Linux han sucumbido bajo un terrible virus. Al seleccionar una persona al azar nos enteramos de que su sistema está infectado por el virus. ¿Cuál es la probabilidad de que sea un alumno con Windows?.

Por el teroema de Bayes se tiene que

$P(windows|virus) = \frac{P(virus|windows)\cdot P(windows)}{P(virus)}$

$= \frac{0.78\cdot 0.6}{0.1\cdot 0.5+0.6\cdot 0.78+0.3\cdot 0.2}$

$= \frac{0.78\cdot 0.6}{0.578}$

$= 0.8096886$

Pregunta 8: Señale si las siguientes declaraciones son verdaderas o falsas respecto a las variables aleatorias:

- [F] Como las variables aleatorias son funciones que nos permiten obtener valores de probabilidad, siempre podemos obtener $\mathbb{P}(X = x) > 0$ evaluando en una $f(x)$ continua y discreta. Justificación: En particular, las funcinones continuas valen 0 evaluando en un punto particular. La manera correcta de obtener un valor es calcular la suma sobre un rango de valores.
- [F] Una PDF bien definida solo puede tener valores menores a 1 y un área debajo de la curva igual a 1. Justificación: Una PDF bien definida sí puede tener valores mayores a 1.
- [V] La CDF(cummulative distribuitive f) puede ser representada como la integral de la PDF(probability dens. f) y PMF (probability mass func.).
- [V] Una CDF es definida para todo x, continua hacia la derecha y no es decreciente.

  Respuesta Aquí

Pregunta 9: Una famosa fabrica de dulces señala que solo el 5% de sus dulces contienen menos de 350 gramos. Si los dulces elaborados por la fabrica distribuyen de forma normal, con media μ y desviación estándar 11.2. Responda las siguientes preguntas:

- a) Encuentre la media del producto.
- b) Señale el porcentaje de dulces que se encuentran sobre los 390 gramos.

Nota: Puede ser útil https://www.statskingdom.com/z_table.html

a) Se normalizan los datos mendiante:

$Z = \frac{x-\mu}{\sigma}$

Luego, se busca en la tabla cual debe ser el valor de Z para que la probabilidad de que un dato sea mayor sea 5%.

$\mathbb{P}(z \leq -1.644854) = 0.05$

Por lo tanto, se tiene que

$-1.644854 = \frac{350-\mu}{11.2}$

$\mu = 1.644854 \cdot 11.2 + 350$

$\mu = 368.4224$

a) Se normalizan los datos

$Z = \frac{x-\mu}{\sigma} = \frac{390-368.4224}{11.2} = 1.926571$

Luego se busca el valor en la tabla

$\mathbb{P}(z \leq 1.93) = 0.9725710503$

$\mathbb{P}(z > 1.93) = 1 - \mathbb{P}(z \leq 1.93)$

$\mathbb{P}(z > 1.93) = 0.02742895$

Por lo tanto, existe un 2,7% de probabilidades de que esté sobre los 390.

Segunda Parte: Elaboración de Código

En la siguiente sección deberá resolver cada uno de los experimentos computacionales a través de la programación en R. Para esto se le aconseja que cree funciones en R, ya que le facilitará la ejecución de gran parte de lo solicitado.

Pregunta 1: Visualización de Datos

Para esta pregunta usted deberá trabajar en base al conjunto de datos `hearth_database.csv`, el cual esta compuesto por las siguientes variables:

- target: Señala si el paciente tuvo un infarto.
- sex: Sexo de los sujetos de prueba.
- fbs: Azúcar en la sangre con ayunas. Esta variable señala solo si se encuentra <=120 o >120.
- exang: Angina de pecho inducida por el ejercicio.
- cp: Tipo de dolor de pecho.
- restecg: Resultados electrocardiográficos en reposo.
- slope: Pendiente del segmento ST máximo de ejercicio.
- ca: Número de buques principales.
- thal: Thalassemia.
- age: Edad en años.
- trestbps: Presión arterial en reposo.
- chol: colesterol sérico en mg/dl.
- thalach: Frecuencia cardíaca máxima alcanzada.
- oldpeak: Depresión del ST inducida por el ejercicio en relación con el reposo.

En base al dataset propuesto realice un análisis exploratorio de los datos (EDA). Para su análisis enfoquen el desarrollo en las siguientes tareas:

- ☐ Obtenga la media, mediana, quintiles y valores máximos desde los datos que componen el dataset.
- ☐ Obtenga la Matriz de correlación de Pearson y visualice la relación entre las variables numéricas.
- ☐ Visualice los boxplot para las variables numéricas.
- ☐ Visualice a través de un histograma como distribuyen las variables respecto a los TARGET.

Respuesta

```
#install.packages("corrplot")
library(corrplot)
```

```
## corrplot 0.90 loaded
```

```
data <- read.csv('hearth_database.csv', header = TRUE, sep= ",", quote = '\"')
numeric_columns <- data[, 7:14]

media <- mapply(mean, numeric_columns)
mediana <- mapply(median, numeric_columns)
quintiles <- mapply(quantile, numeric_columns, MoreArgs = list(seq(0, 1, 1/5)))
maximos <- mapply(max, numeric_columns)
print(media)
```

```
##       slope          ca        thal         age    trestbps        chol
##   1.3993399   0.7293729   2.3135314  54.3663366 131.6237624 246.2640264
##     thalach     oldpeak
## 149.6468647   1.0396040
```

```
print(mediana)
```

```
##    slope       ca     thal      age trestbps     chol  thalach  oldpeak
##      1.0      0.0      2.0     55.0    130.0    240.0    153.0      0.8
```

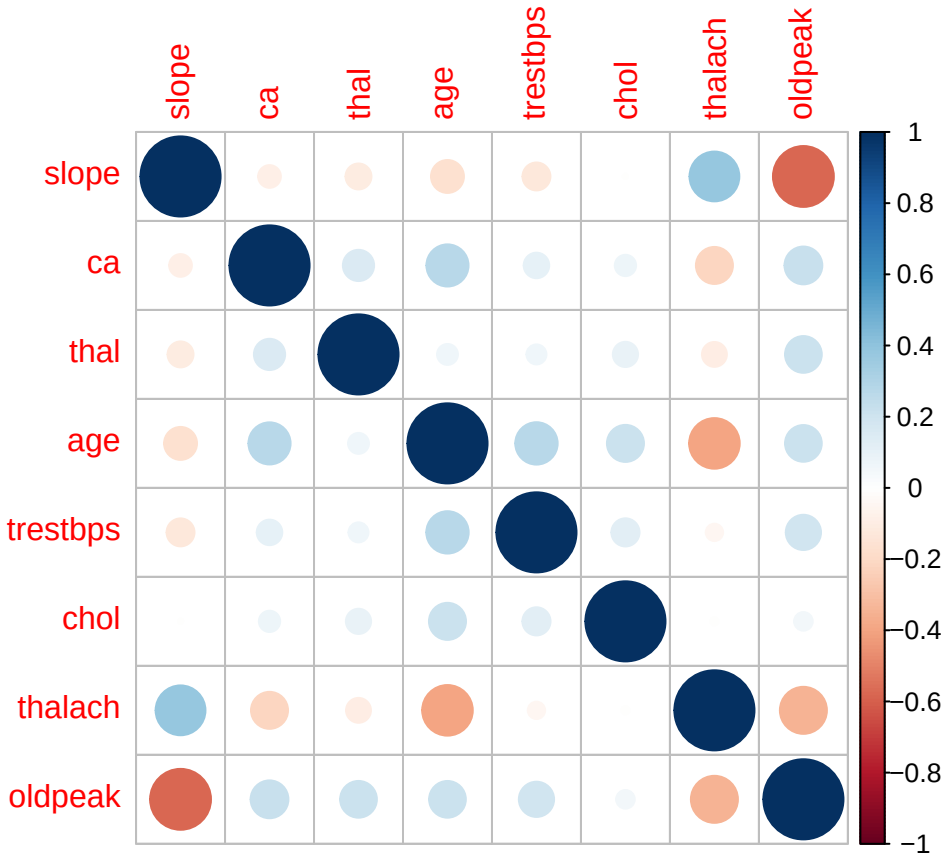
```
print(quintiles)
```

```
##      slope ca thal age trestbps  chol thalach oldpeak
## 0%       0  0    0  29       94 126.0      71    0.00
## 20%      1  0    2  45      120 204.0     130    0.00
## 40%      1  0    2  53      126 230.0     146    0.38
## 60%      2  1    2  58      134 254.0     159    1.12
## 80%      2  2    3  62      144 285.2     170    1.90
## 100%     2  4    3  77      200 564.0     202    6.20
```

```
print(maximos)
```

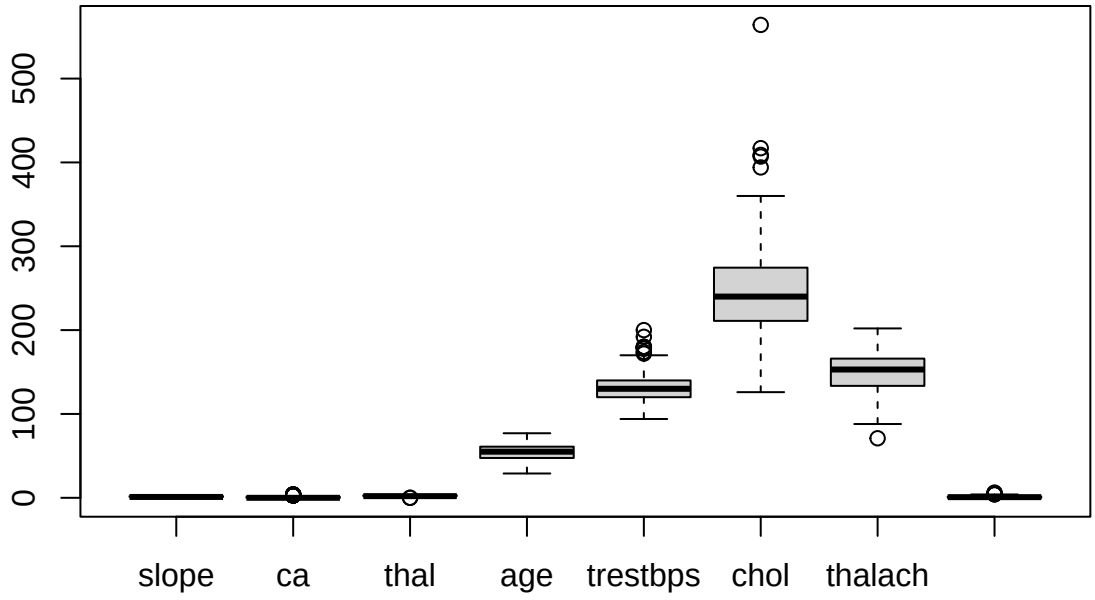
```
##    slope       ca     thal      age trestbps     chol  thalach  oldpeak
##      2.0      4.0      3.0     77.0    200.0    564.0    202.0      6.2
```

```
numeric_columns.cor <- cor(numeric_columns)
corrplot(numeric_columns.cor)
```

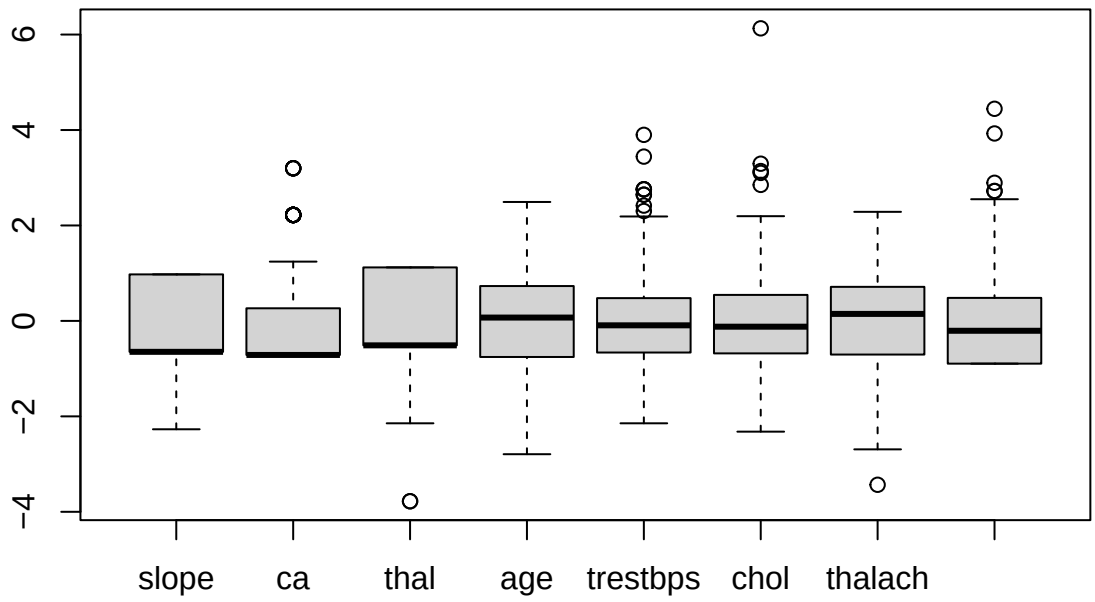


```
# Se observa que hay una relación inversa notable entre slope y oldpeak, y unas menores entre thalach y
```

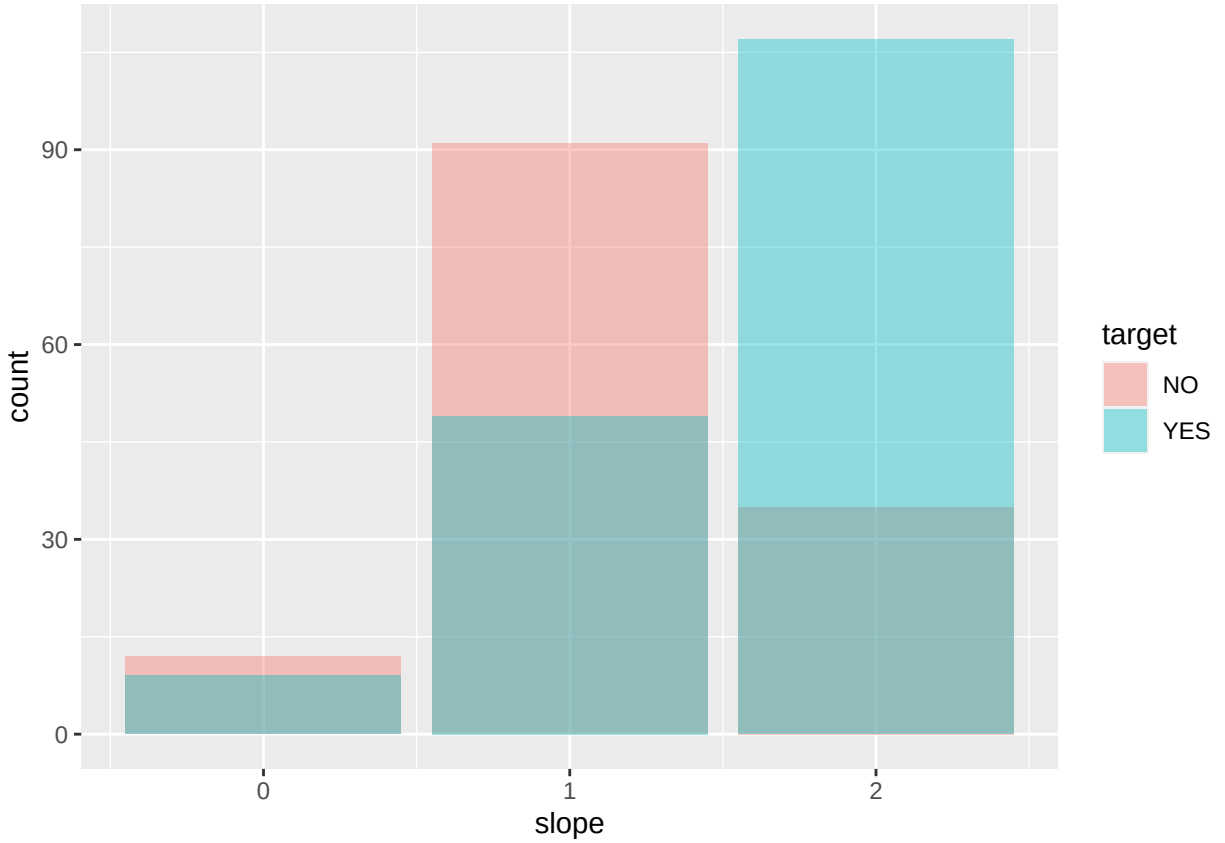
```
#NORMAL
boxplot(numeric_columns)
```



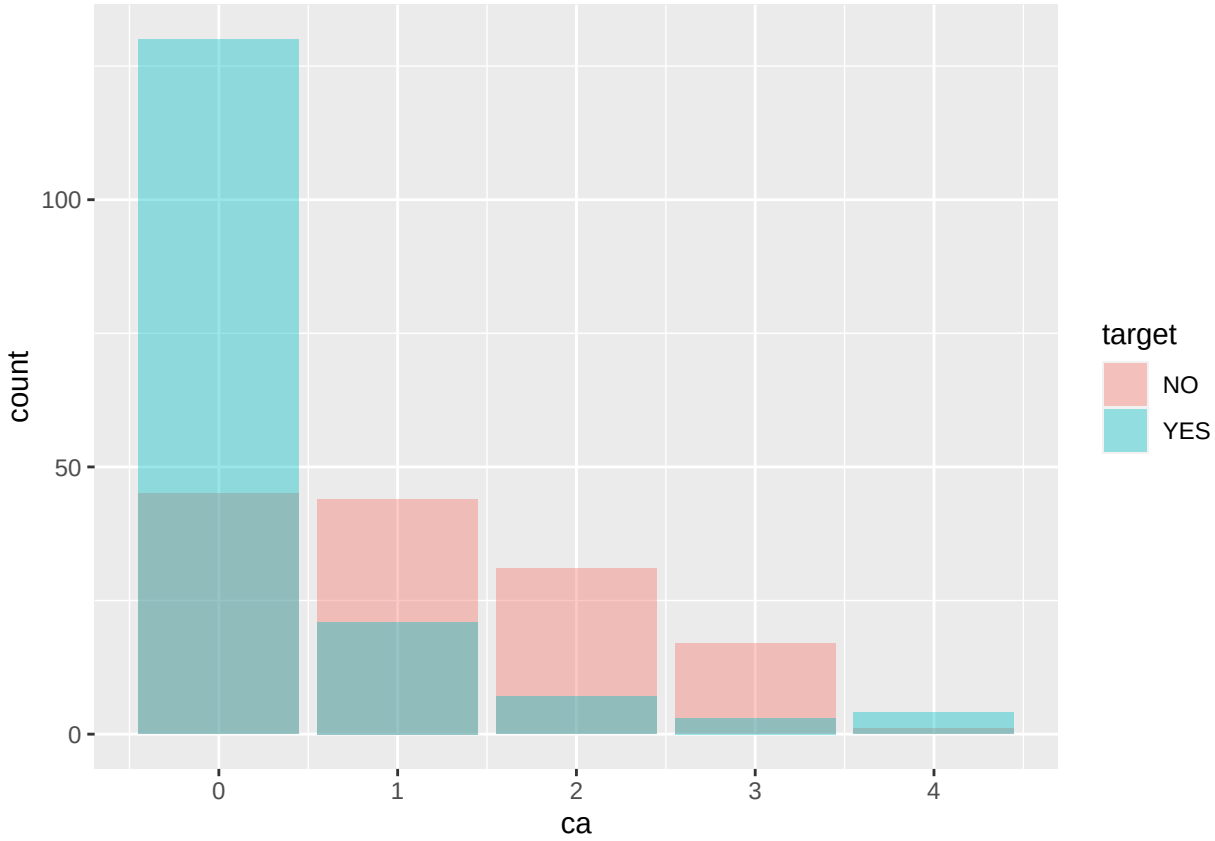

```
#NORMALIZE BY Z-SCORE
boxplot(scale(numeric_columns))
```



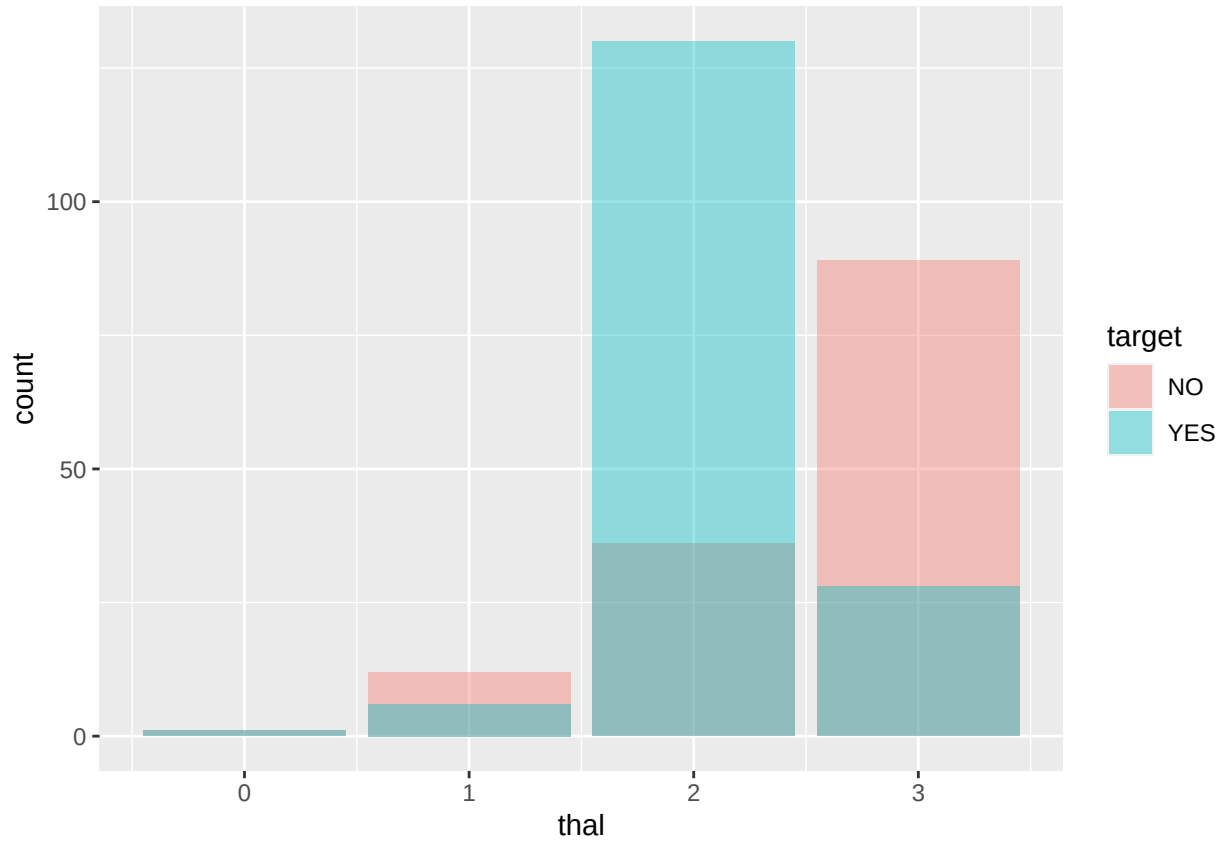

```
#install.packages("ggplot2")
library(ggplot2)


#yes_frame <- data[data[, c(1)] == "YES",]
#no_frame  <- data[data[, c(1)] == "NO", ]

#qplot(yes_frame$slope, geom="histogram")

#dat <- data.frame(xx = c(runif(100,20,50),runif(100,40,80),runif(100,0,30)),yy = rep(letters[1:3],each
data_f <-as.data.frame(data)

ggplot(data_f, aes(x = slope, fill = target)) + geom_bar(position = "identity", alpha = 0.4)
```



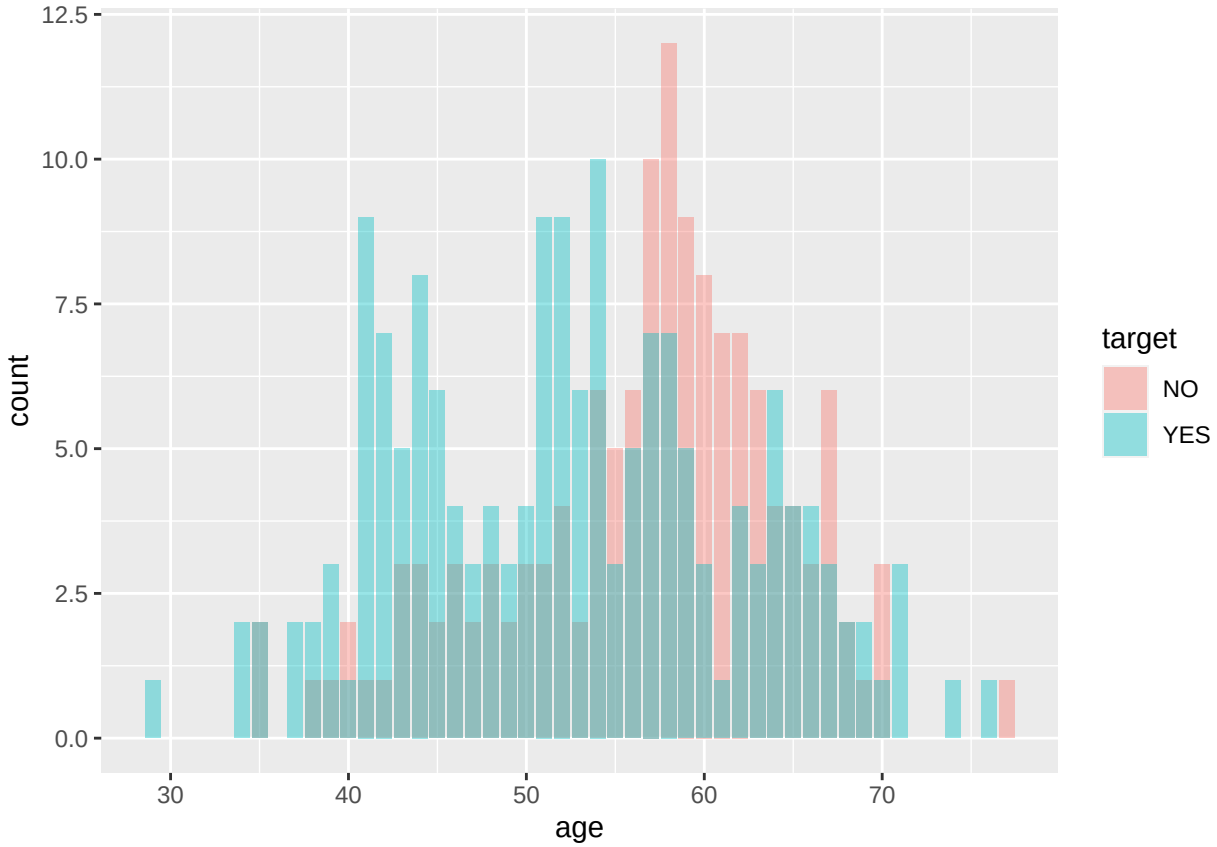

```
ggplot(data_f, aes(x = ca, fill = target)) + geom_bar(position = "identity", alpha = 0.4)
```



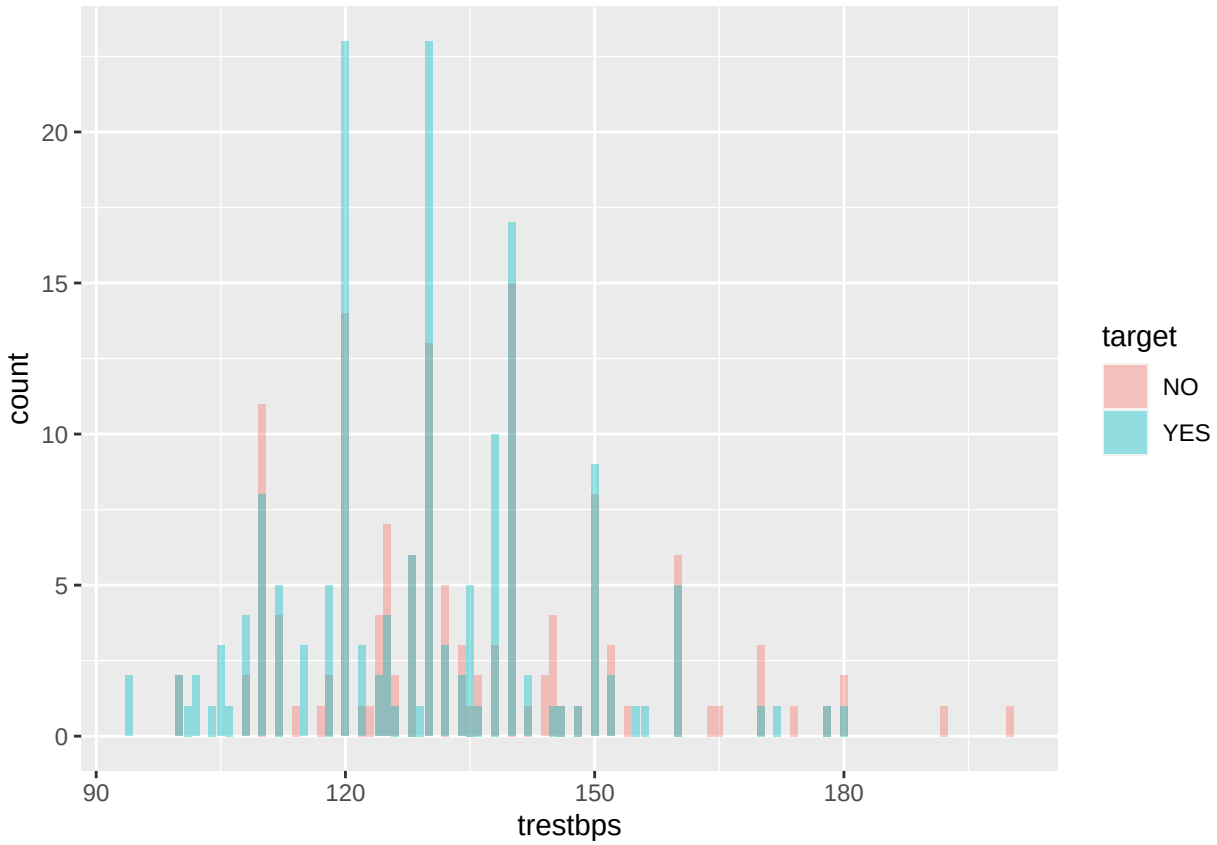

```
ggplot(data_f, aes(x = thal, fill = target)) + geom_bar(position = "identity", alpha = 0.4)
```



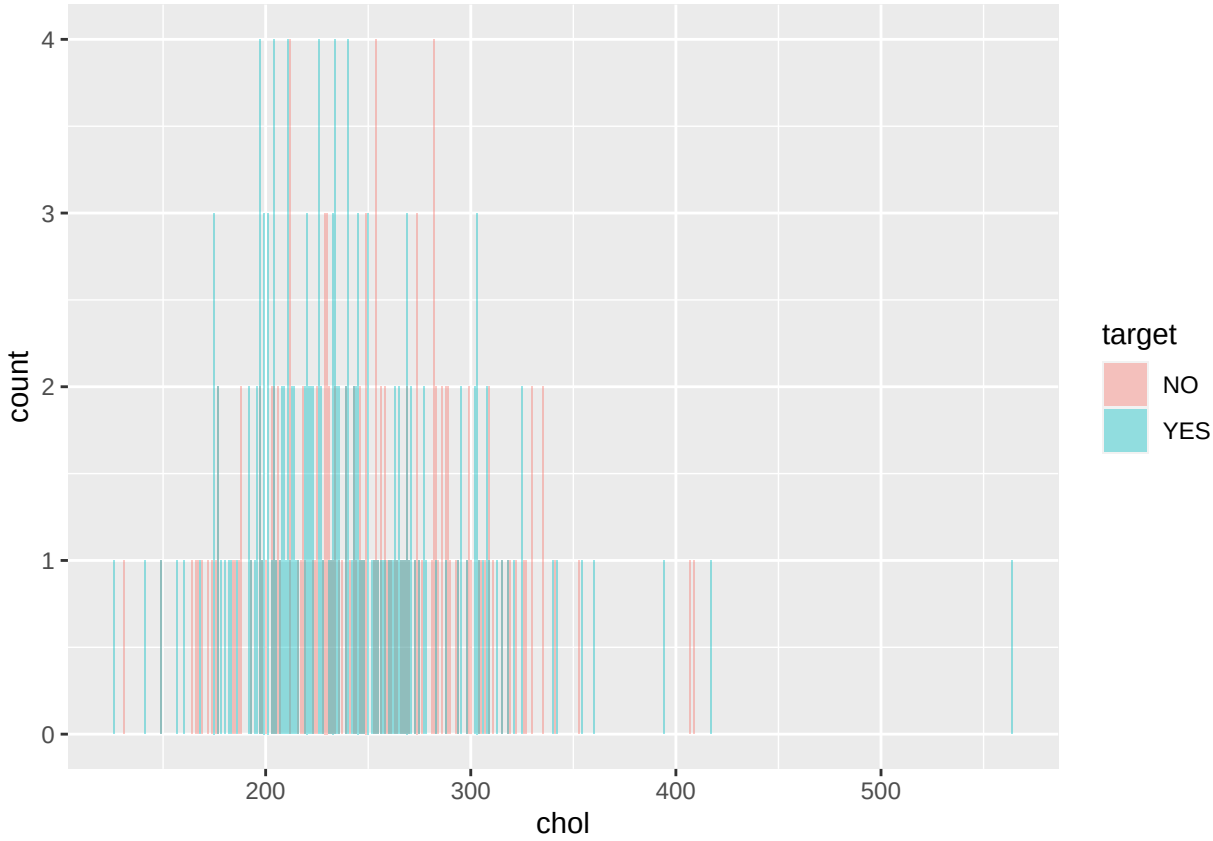

```
ggplot(data_f, aes(x = age, fill = target)) + geom_bar(position = "identity", alpha = 0.4)
```



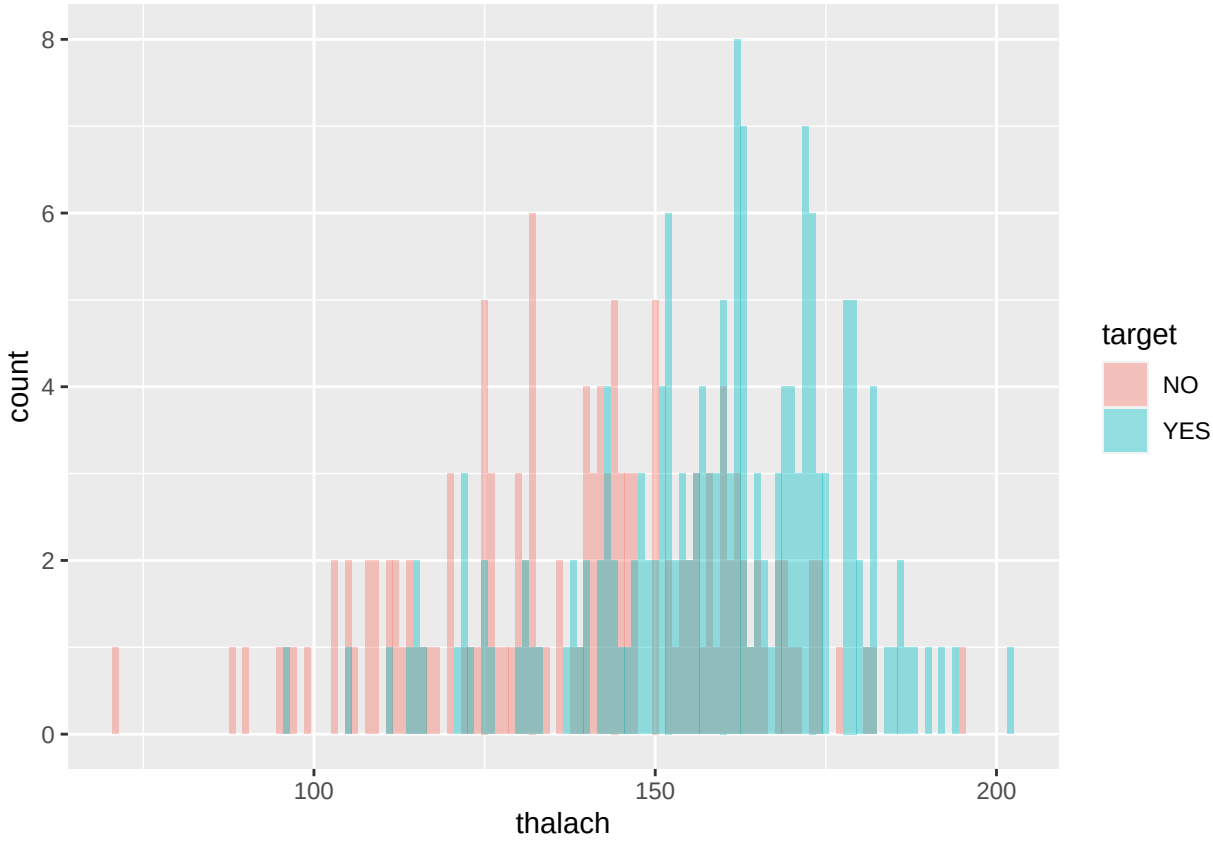

```
ggplot(data_f, aes(x = trestbps, fill = target)) + geom_bar(position = "identity", alpha = 0.4)
```



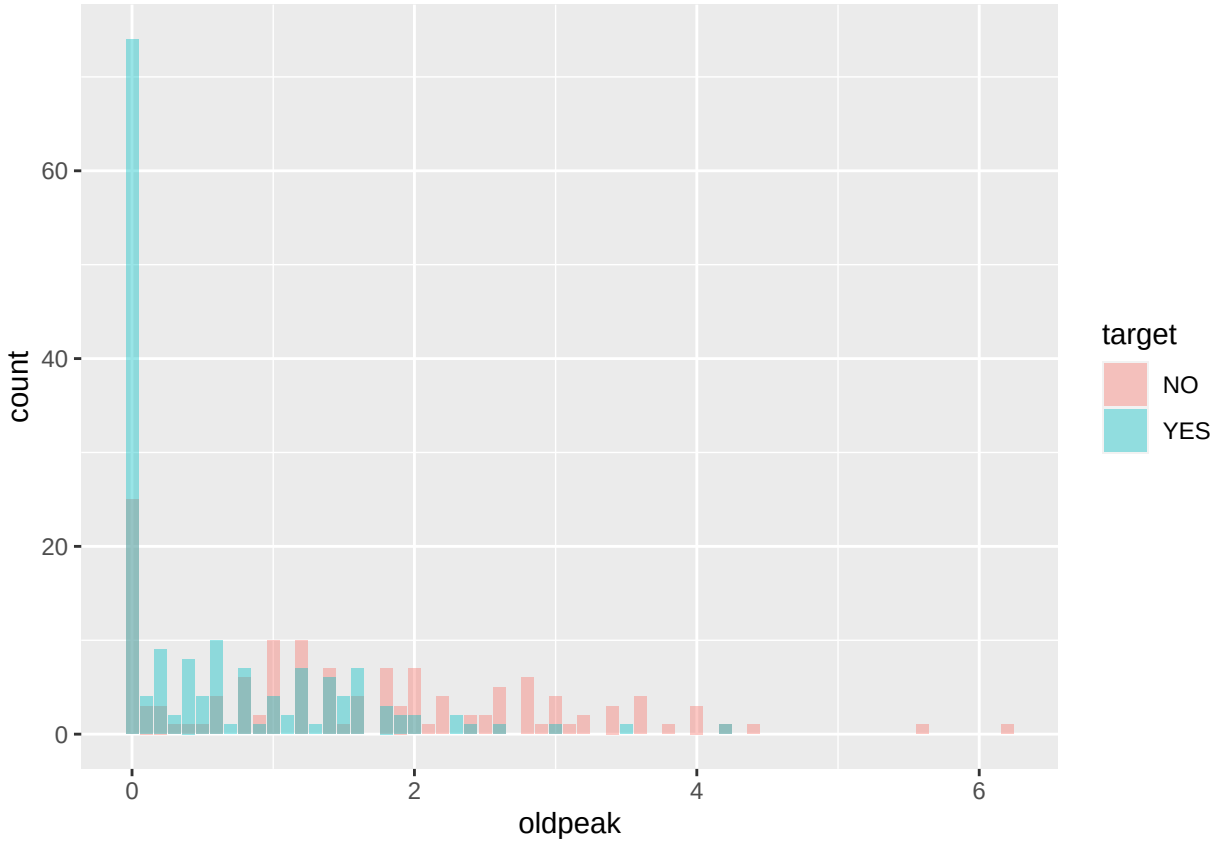

```
ggplot(data_f, aes(x = chol, fill = target)) + geom_bar(position = "identity", alpha = 0.4)
```




```
ggplot(data_f, aes(x = thalach, fill = target)) + geom_bar(position = "identity", alpha = 0.4)
```




```
ggplot(data_f, aes(x = oldpeak, fill = target)) + geom_bar(position = "identity", alpha = 0.4)
```



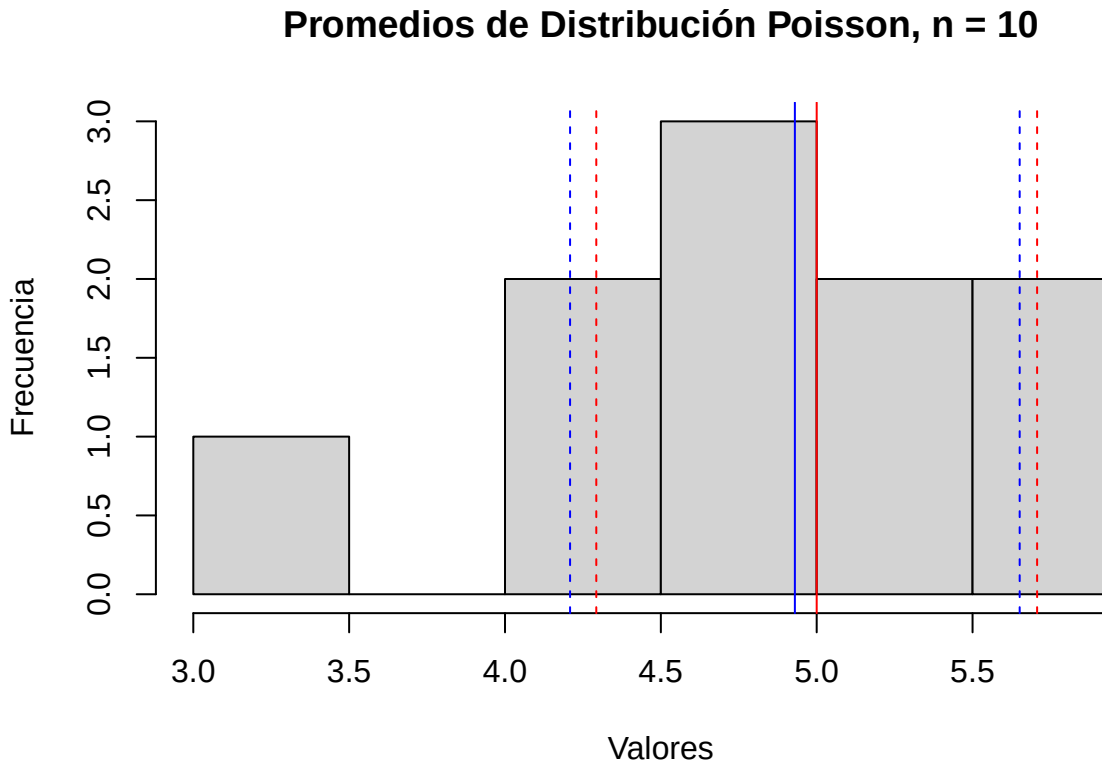
Pregunta 2: Teorema Central del Limite

Pruebe el teorema central del limite aplicando un muestreo de la media en las distribuciones Poisson, Exponencial y una a su elección. Grafique los resultados obtenidos y señale aproximadamente el numero de muestreos necesarios para obtener el resultado esperado, pruebe esto con las siguientes cantidades de muestreo $\{10, 100, 1000, 5000\}$. ¿El efecto ocurre con el mismo número de muestreo para todas las distribuciones?.

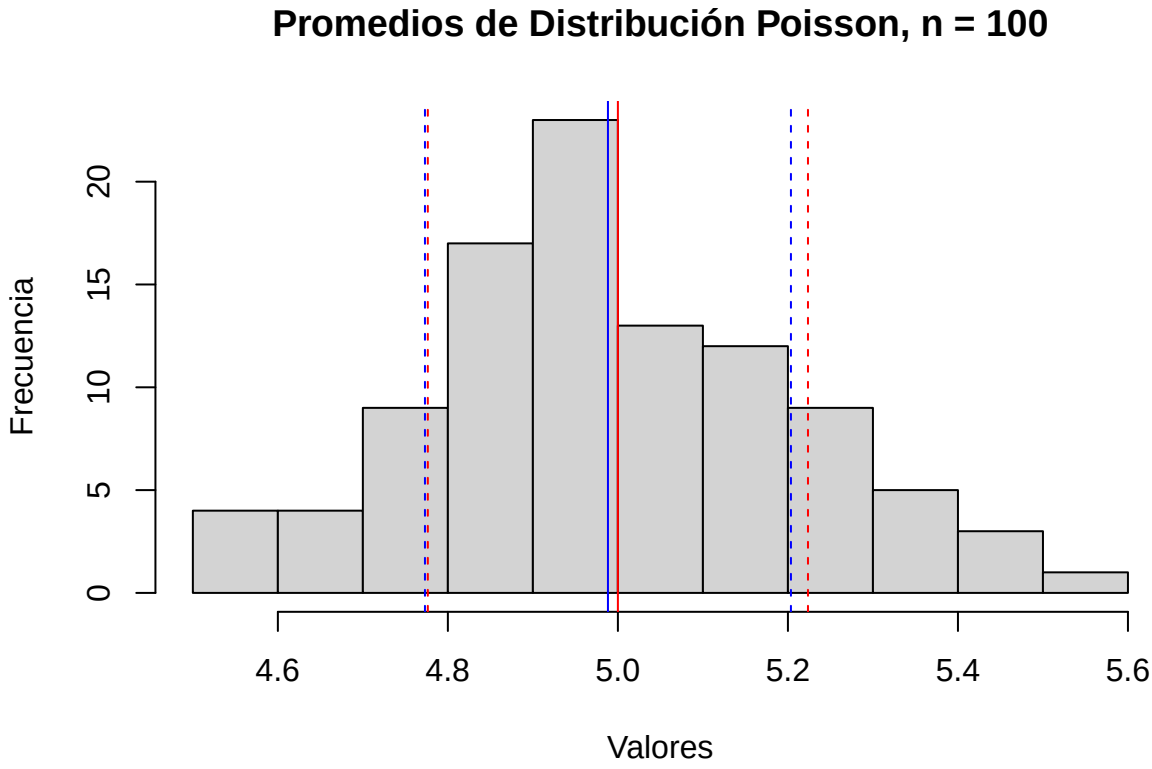
Por el terorema central del limite (TLC) aplicado a promdeios se tiene que los promedios de una muestra n formarán una distribución normal de $\mu = \mu$ y $\sigma^2 = \frac{\sigma^2}{n}$. A continuación se demuestra esto para distintas distribuciones y número de muestras.

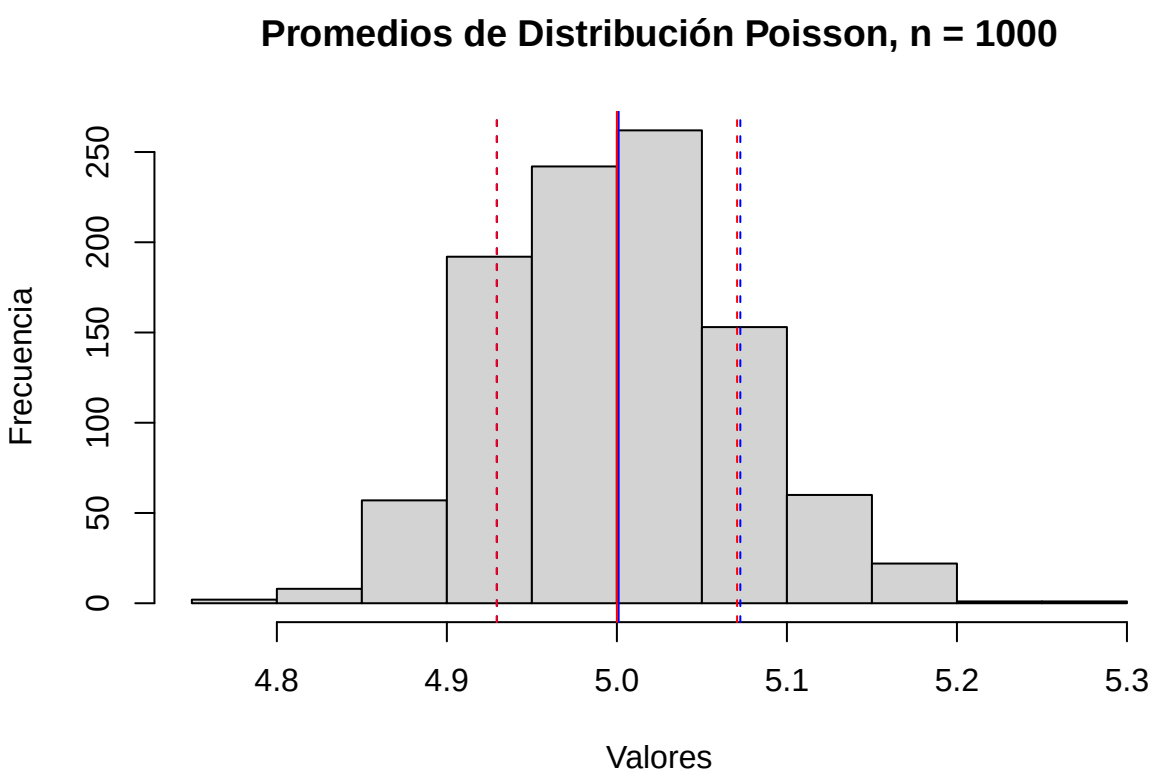
```
# Definición de variables o estructuras necesarias para el muestreo.
n<- 100
lambda_dp = 5

mean_dp <- lambda_dp
var_dp <- lambda_dp

for(n in c(10,100,1000,5000)){
  means_dp <- vector("numeric",n)
  vars_dp <- vector("numeric",n)

  for(i in 1:n){
    dp = rpois(n, lambda_dp)
```

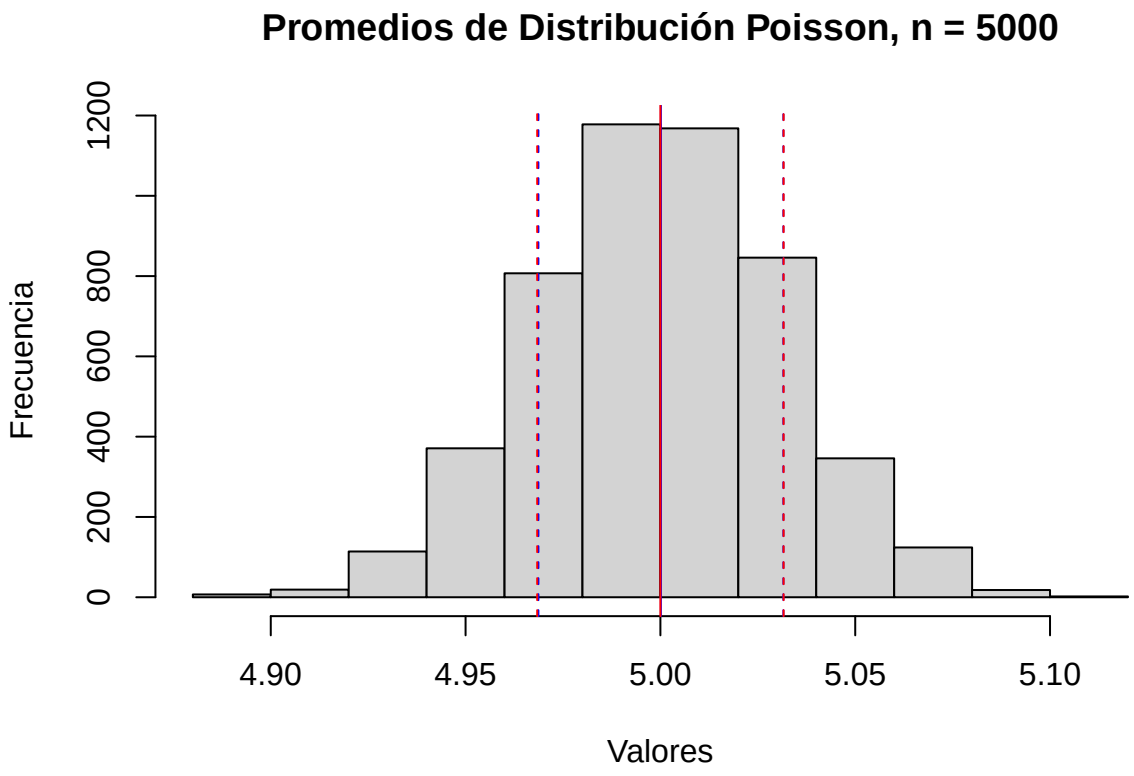
```
    means_dp[i] <- mean(dp)
    vars_dp[i] <- var(dp)
  }

  hist(means_dp, main = paste("Promedios de Distribución Poisson, n =", n, sep=" "), ylab = "Frecuencia
  abline(v = mean(means_dp), col='blue')
  abline(v = mean(means_dp)+sqrt(var(means_dp)),col='blue', lty = 2)
  abline(v = mean(means_dp)-sqrt(var(means_dp)),col='blue', lty = 2)

  abline(v = mean_dp, col='red')
  abline(v = mean_dp + sqrt(var_dp/n), col='red',  lty = 2)
  abline(v = mean_dp - sqrt(var_dp/n), col='red',  lty = 2)

}
```



Dsitribución de Poisson

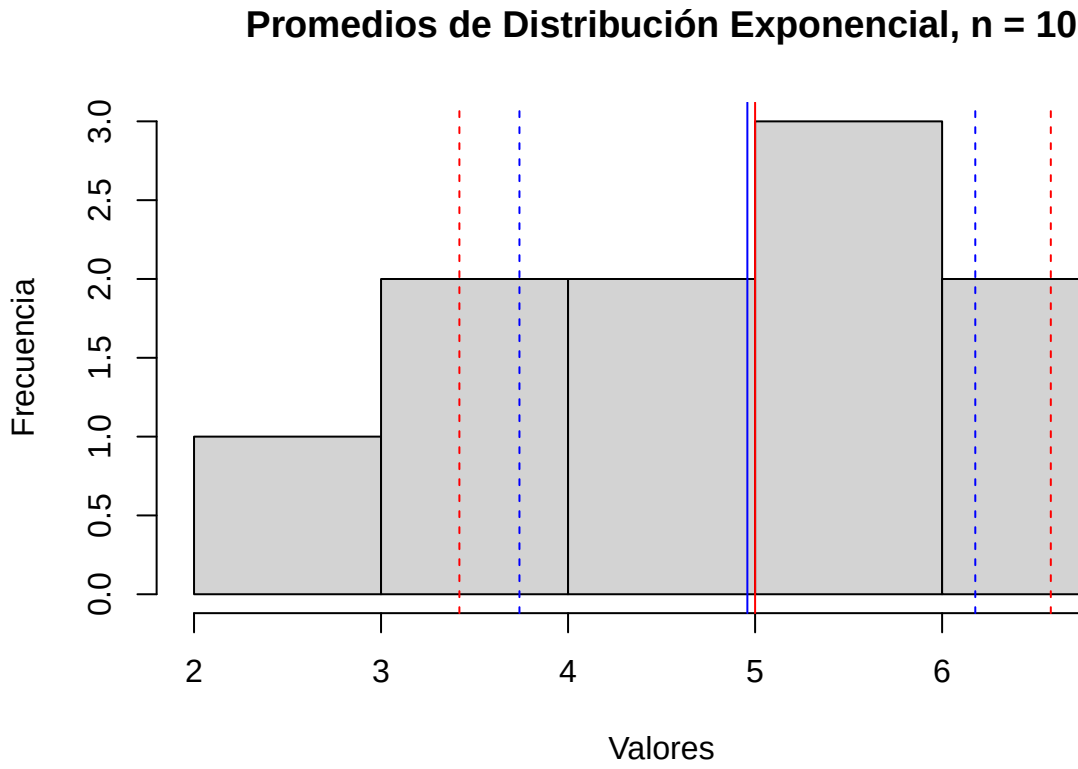



Promedios de Distribución Poisson, n = 1000
Frecuencia
0
50
100
150
200
250
4.8
4.9
5.0
5.1
5.2
5.3
Valores

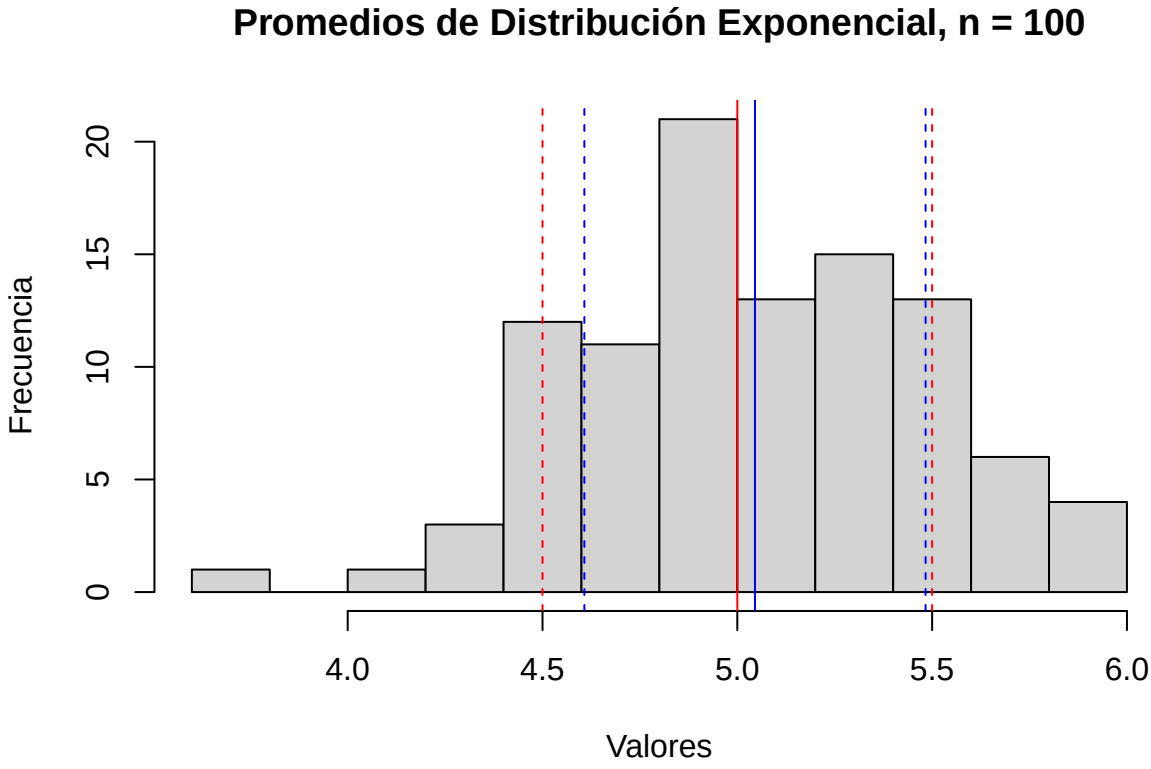


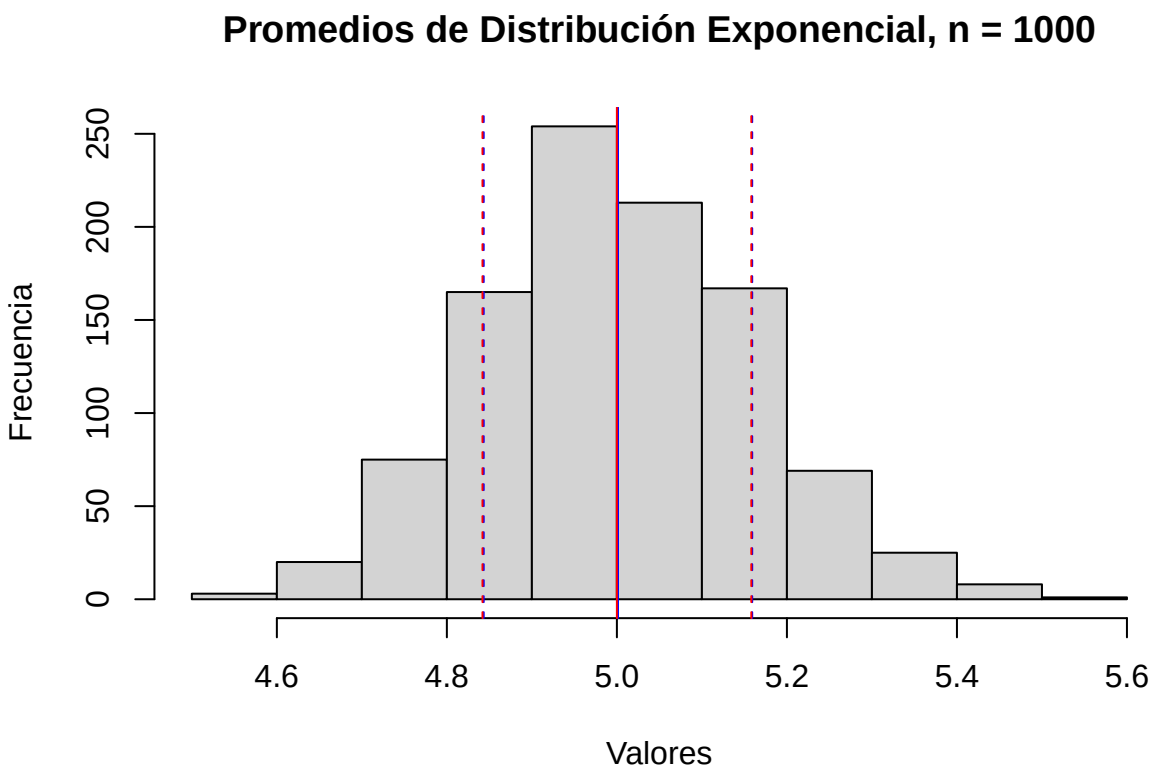

```r
# Definición de variables o estructuras necesarias para el muestreo.
n<- 100
lambda_de = 0.2

mean_de <- lambda_de^(-1)
var_de <- lambda_de^(-2)

for(n in c(10,100,1000,5000)){
  means_de <- vector("numeric",n)
  vars_de <- vector("numeric",n)

  for(i in 1:n){
    de = rexp(n, lambda_de)

    means_de[i] <- mean(de)
    vars_de[i] <- var(de)
  }

  hist(means_de, main = paste("Promedios de Distribución Exponencial, n =", n, sep=" "), ylab = "Frecuer
  abline(v = mean(means_de), col='blue')
  abline(v = mean(means_de)+sqrt(var(means_de)),col='blue', lty = 2)
  abline(v = mean(means_de)-sqrt(var(means_de)),col='blue', lty = 2)

  abline(v = mean_de, col='red')
```

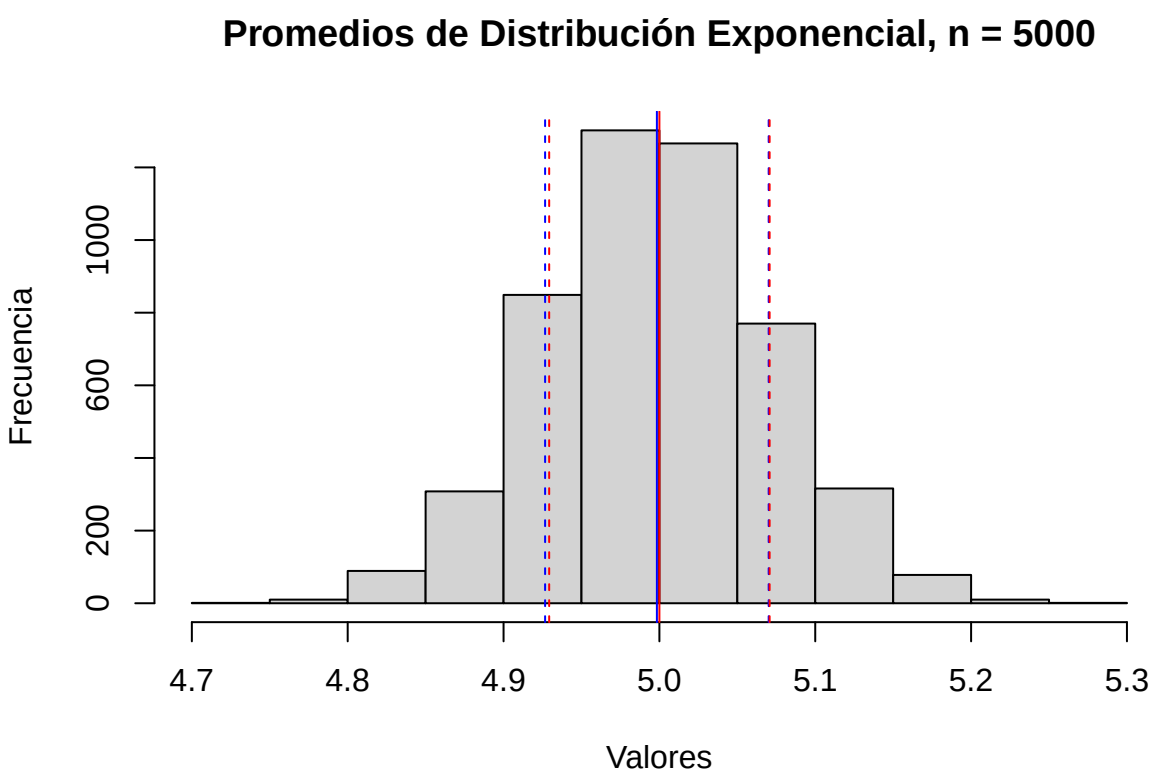
```
  abline(v = mean_de + sqrt(var_de/n), col='red',  lty = 2)
  abline(v = mean_de - sqrt(var_de/n), col='red',  lty = 2)
}
```



Dsitribución Exponencial




Promedios de Distribución Exponencial, n = 1000
Frecuencia
0
50
100
150
200
250
4.6
4.8
5.0
5.2
5.4
5.6
Valores




```r
# Definición de variables o estructuras necesarias para el muestreo.

mean_dn <- 0
var_dn <- 1

for(n in c(10,100,1000,5000)){
  means_dn <- vector("numeric",n)
  vars_dn <- vector("numeric",n)

  for(i in 1:n){
    dn = rnorm(n)

    means_dn[i] <- mean(dn)
    vars_dn[i] <- var(dn)
  }

  hist(means_dn, main = paste("Promedios de Distribución Poisson, n =", n, sep=" "), ylab = "Frecuencia
  abline(v = mean(means_dn), col='blue')
  abline(v = mean(means_dn)+sqrt(var(means_dn)),col='blue', lty = 2)
  abline(v = mean(means_dn)-sqrt(var(means_dn)),col='blue', lty = 2)

  abline(v = mean_dn, col='red')
  abline(v = mean_dn + sqrt(var_dn/n), col='red',  lty = 2)
  abline(v = mean_dn - sqrt(var_dn/n), col='red',  lty = 2)
```

```
}
```

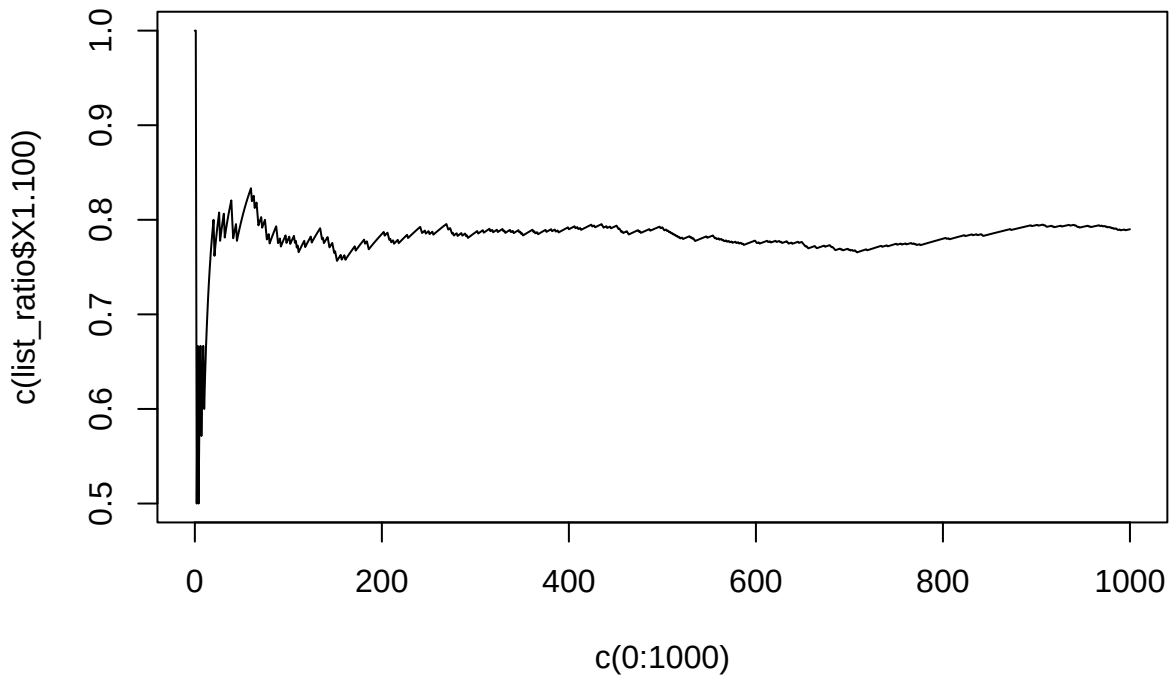
Dsitribución Normal

Pregunta 3: Ley de los Grandes Numeros.

Lanzamiento de monedas Realice el experimento de lanzar una moneda cargada 1000 veces y observe el comportamiento que tiene la probabilidad de salir cara. Para realizar el experimento considere que la moneda tiene una probabilidad de 4/5 de salir cara. Grafique el experimento para las secuencias de intentos que van desde 1 a 1000, señalando el valor en que converge la probabilidad de salir cara.

Respuesta

```
# Simular lanzamientos

lanzamientos <- runif(1000, 1, 100)
comportamiento <- lanzamientos >= 20
list_ratio <- data.frame(1:1000)
suma_true <- 0
suma_falsa <- 0
for (tirada  in comportamiento){
  if (tirada){suma_true  <- suma_true  + 1}
  else{       suma_falsa <- suma_falsa + 1}
  list_ratio[1+suma_falsa+suma_true, 1] <- suma_true/(suma_falsa + suma_true)
}
plot(c(0:1000), c(list_ratio$X1.100), type = "l")
```



```r
# Gráfico de la convergencia
```

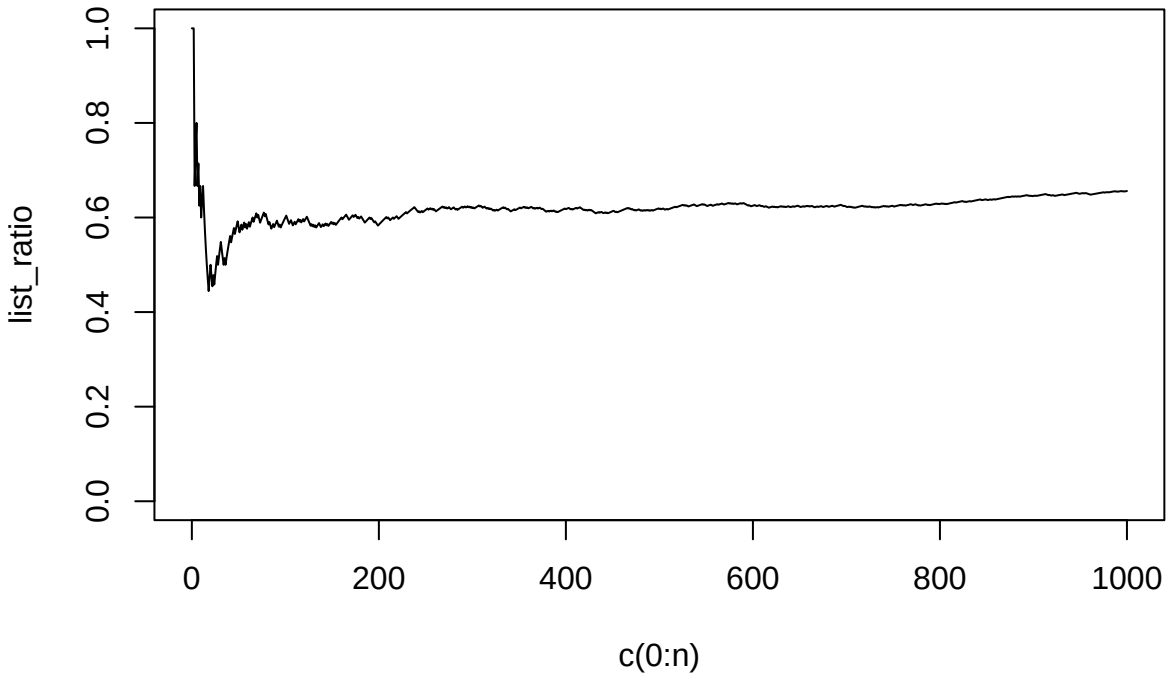
El problema de Monty Hall Remontándonos en la televisión del año 1963, en USA existía un programa de concursos donde los participantes debían escoger entre 3 puertas para ganar un premio soñado. El problema del concurso era que solo detrás de 1 puerta estaba el premio mayor, mientras que detrás de las otras dos habían cabras como "premio".

Una de las particularidades de este concurso, es que cuando el participante escogía una puerta, el animador del show abría una de las puertas que no fue escogida por el participante (Obviamente la puerta abierta por el animador no contenía el premio). Tras abrir la puerta, el animador consultaba al participante si su elección era definitiva, o si deseaba cambiar la puerta escogida por la otra puerta cerrada.

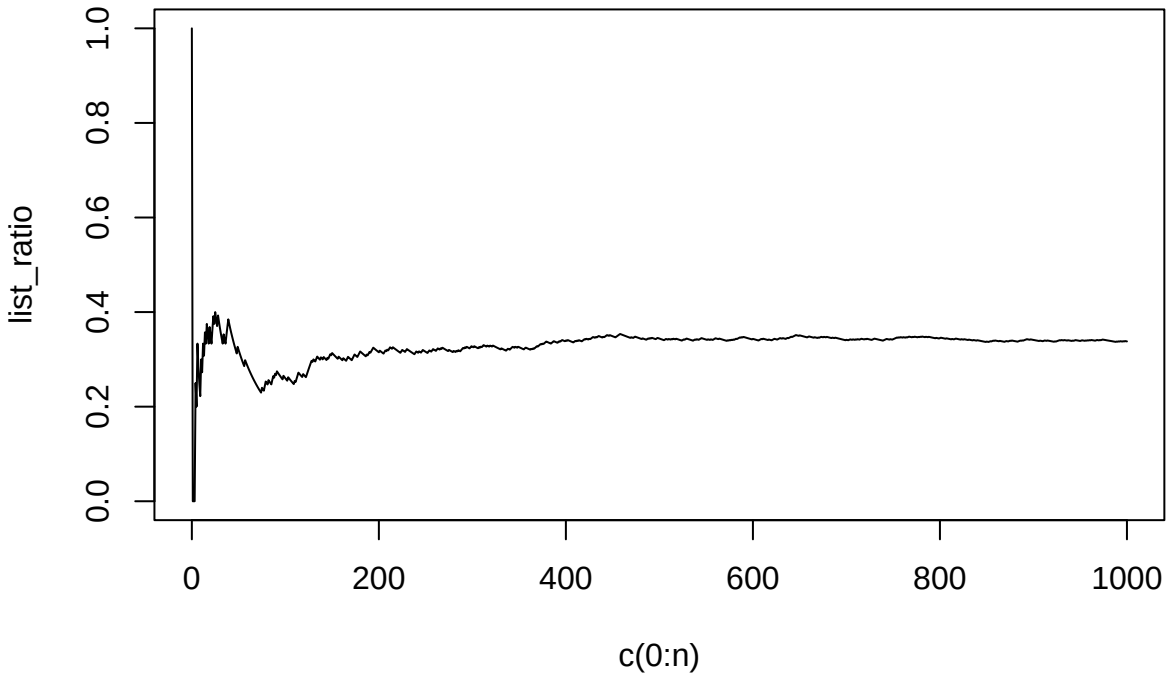
Imagine que usted es participante del concurso y desea calcular la probabilidad de ganar el gran premio **si cambia de puerta** en el momento que el animador se lo ofrece. Utilizando listas/arrays/vectores simule las puertas del concurso, dejando aleatoriamente el premio en alguna posición del array. Hecho esto, genere un numero de forma aleatoria para escoger una de las puerta (posiciones de la estructura), para luego ver si cambiando de posición tendrá mayores posibilidades de ganar el premio. Genere N veces el experimento y grafique cada una de las iteraciones, tal como se hizo en el ejercicio de las monedas.

Respuesta:

```r
# Creamos una función que simule el juego
montyhall <- function(cambiar = TRUE){
  Puertas <- sample(1:3,3)             #Puertas donde la posición que tiene el 3 es el premio
  first_choose <- sample(1:3,1)            #Elección del participante.
  TRUE
  if (which(Puertas == 3) == first_choose){# le achunta a la primera
    return (!cambiar)

  }
  else{# no le achunta a la primera
    return (cambiar)
  }
  # Retornamos la elección, esta puede que tenga el premio o no
}

# Función que simula N juegos
n_juegos <- function(n = 1000 ,cambiar_puerta = TRUE){
  res <- c()
  for (i in (1:n)){
    res <- rbind(res, montyhall(cambiar = cambiar_puerta))
  }
  comportamiento <- res
  list_ratio <- c(1:n)
  suma_true <- 0
  suma_falsa <- 0
  for (tirada  in comportamiento){
    if (tirada){suma_true  <- suma_true  + 1}
    else{       suma_falsa <- suma_falsa + 1}
    list_ratio[1+suma_falsa+suma_true] <- suma_true/(suma_falsa + suma_true)
  }
  plot(c(0:n), list_ratio, type = "l", ylim=c(0,1))

}
n_juegos(1000, TRUE)
```




```
n_juegos(1000, FALSE)
```



Pregunta 4: ¿Independencia?

Ustedes disponen de los dados D1 y D2, los cuales no están cargados y son utilizados para comprobar que $\mathbb{P}(AB) = \mathbb{P}(A)\mathbb{P}(B)$ cuando el evento A es independiente del B. Para estudiar la independencia considere que los eventos A y B se definen de la siguiente manera; sea A el evento dado por los valores obtenidos en el lanzamiento del dado D1, este está compuesto por $A = \{D1 = 1, D1 = 2, D1 = 6\}$. Por otro lado, el evento B viene dado por los valores obtenidos con el dado D2, el que está conformado por $B = \{D2 = 1, D2 = 2, D2 = 3, D2 = 4\}$. Con esto, tendremos un $\mathbb{P}(A) = 1/2$, $\mathbb{P}(B) = 2/3$ y $\mathbb{P}(AB) = 1/3$. Compruebe de forma gráfica que al realizar 1000 lanzamientos (u otro valor grande que usted desea probar) se visualiza que $\mathbb{P}(AB) = \mathbb{P}(A)\mathbb{P}(B)$.

Hecho lo anterior, compruebe el comportamiento de un segundo grupo de eventos, dados por el lanzamiento de solo el dado D1. Donde, los eventos para D1 quedan definidos como: $A = \{D1 = 1, D1 = 2, D1 = 6\}$ y $B = \{D1 = 1, D1 = 2, D1 = 3\}$. ¿Se observa independencia en este experimento?.

Se le aconseja que para simular los lanzamientos de dados utilice la función `sample()` para generar valores aleatorios entre 1 y 6. Compruebe los números generados por la función con los casos favorables de cada uno de los eventos a ser estudiados.

** Lanzamiento de dados **

```
N_lan <- 1000 # Numero de lanzamientos

vec_D1 <- sample(1:6, size = N_lan, replace = TRUE)
vec_D2 <- sample(1:6, size = N_lan, replace = TRUE)
```

** Eventos independientes **

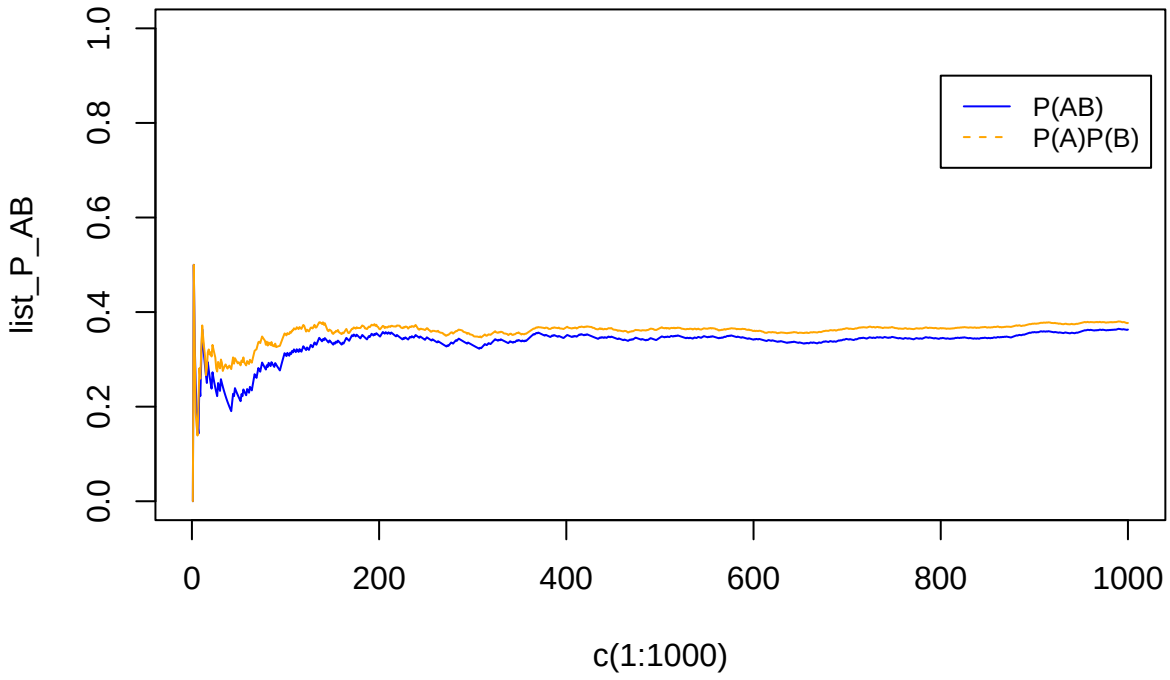
```r
cond_A <- vec_D1 %in% c(1,2,6)
cond_B <- vec_D2 %in% c(1,2,3,4)

L_A  <- sum(cond_A , na.rm=TRUE)          # Lanzamientos favorables A = c(1, 2, 6)
L_B  <- sum(cond_B , na.rm=TRUE)          # Lanzamientos favorables B = c(1, 2, 3, 4)
L_AB <- sum(cond_A & cond_B, na.rm=TRUE)  # Lanzamientos favorables AB = c(1, 2)

P_A <- L_A/N_lan
P_B <-  L_B/N_lan

P_AB <- L_AB/N_lan
PA_PB <- P_A*P_B

message("P(AB): ", P_AB)
```

```
## P(AB): 0.363
```

```r
message("P(A)P(B): ", PA_PB)
```

```
## P(A)P(B): 0.377024
```

```r
list_P_AB <- c(1:1000)  # probabilidad de ambas
list_PA_PB <- c(1:1000) # producto

suma_true_A <- 0
suma_true_B <- 0
suma_true_AB <- 0
count <- 0

for (i in 1:N_lan){
  if (cond_A[i]){suma_true_A  <- suma_true_A  + 1}
  if (cond_B[i]){suma_true_B  <- suma_true_B  + 1}
  if (cond_A[i] & cond_B[i]){suma_true_AB  <- suma_true_AB  + 1}
  count <- count + 1

  P_A <- suma_true_A/(count)
  P_B <- suma_true_B/(count)
  P_AB <- suma_true_AB/(count)
  PA_PB <- P_A * P_B

  list_P_AB[count] <- P_AB
  list_PA_PB[count] <- PA_PB
}
plot(c(1:1000), list_P_AB, type = "l", col = "blue", xlim=c(1,1000), ylim=c(0,1))
lines(c(1:1000), list_PA_PB, type = "l", col = "orange")

legend(800, 0.9, legend=c("P(AB)", "P(A)P(B)"),
       col=c("blue", "orange"), lty=1:2, cex=0.8)
```



** Eventos dependientes **

```
cond_A <- vec_D1 %in% c(1,2,6)
cond_B <- vec_D1 %in% c(1,2,3,4)

L_A <- sum(cond_A , na.rm=TRUE) # Lanzamientos favorables A = c(1, 2, 6)
L_B <- sum(cond_B , na.rm=TRUE) # Lanzamientos favorables B = c(1, 2, 3, 4)
L_AB <- sum( cond_A & cond_B, na.rm=TRUE) # Lanzamientos favorables AB = c(1, 2)

P_A <- L_A/N_lan
P_B <- L_B/N_lan
P_AB <- L_AB/N_lan
PA_PB <- P_A*P_B

message("P(AB): ", P_AB)
```

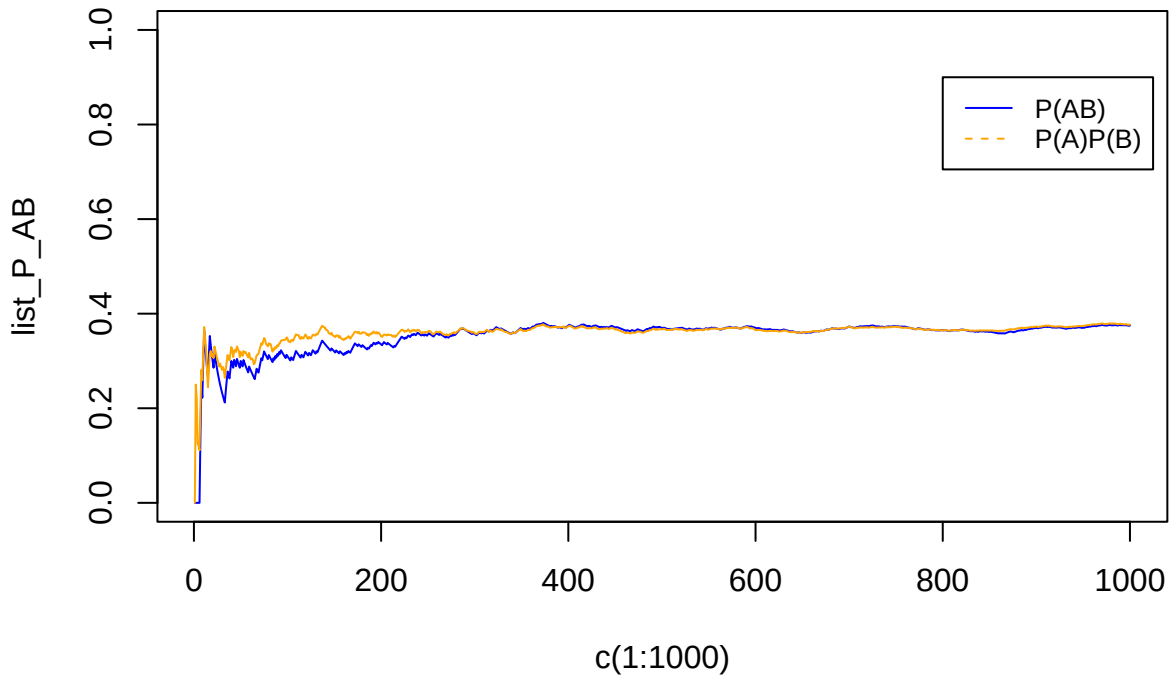
```
## P(AB): 0.375
```

```
message("P(A)P(B): ", PA_PB)
```

```
## P(A)P(B): 0.377024
```

```
list_P_AB <- c(1:1000)  # probabilidad de ambas
list_PA_PB <- c(1:1000) # producto

suma_true_A <- 0
suma_true_B <- 0
suma_true_AB <- 0
```

```r
count <- 0

for (i in 1:N_lan){
  if (cond_A[i]){suma_true_A  <- suma_true_A  + 1}
  if (cond_B[i]){suma_true_B  <- suma_true_B  + 1}
  if (cond_A[i] & cond_B[i]){suma_true_AB  <- suma_true_AB  + 1}
  count <- count + 1

  P_A <- suma_true_A/(count)
  P_B <- suma_true_B/(count)
  P_AB <- suma_true_AB/(count)
  PA_PB <- P_A * P_B

  list_P_AB[count] <- P_AB
  list_PA_PB[count] <- PA_PB
}
plot(c(1:1000), list_P_AB, type = "l", col = "blue", xlim=c(1,1000), ylim=c(0,1))
lines(c(1:1000), list_PA_PB, type = "l", col = "orange")

legend(800, 0.9, legend=c("P(AB)", "P(A)P(B)"),
       col=c("blue", "orange"), lty=1:2, cex=0.8)
```



Pregunta 5: La Ruina del Jugador

Un amigo ludópata suyo le comenta que el truco de jugar en el casino esta en no parar de apostar y apostando lo mínimo posible. Ya que así, tienes mas probabilidades de ganar el gran pozo que acumula el juego. Usted sabiendo la condición de su amigo, decide no creer en su conjetura y decide probar esto a través de un experimento.

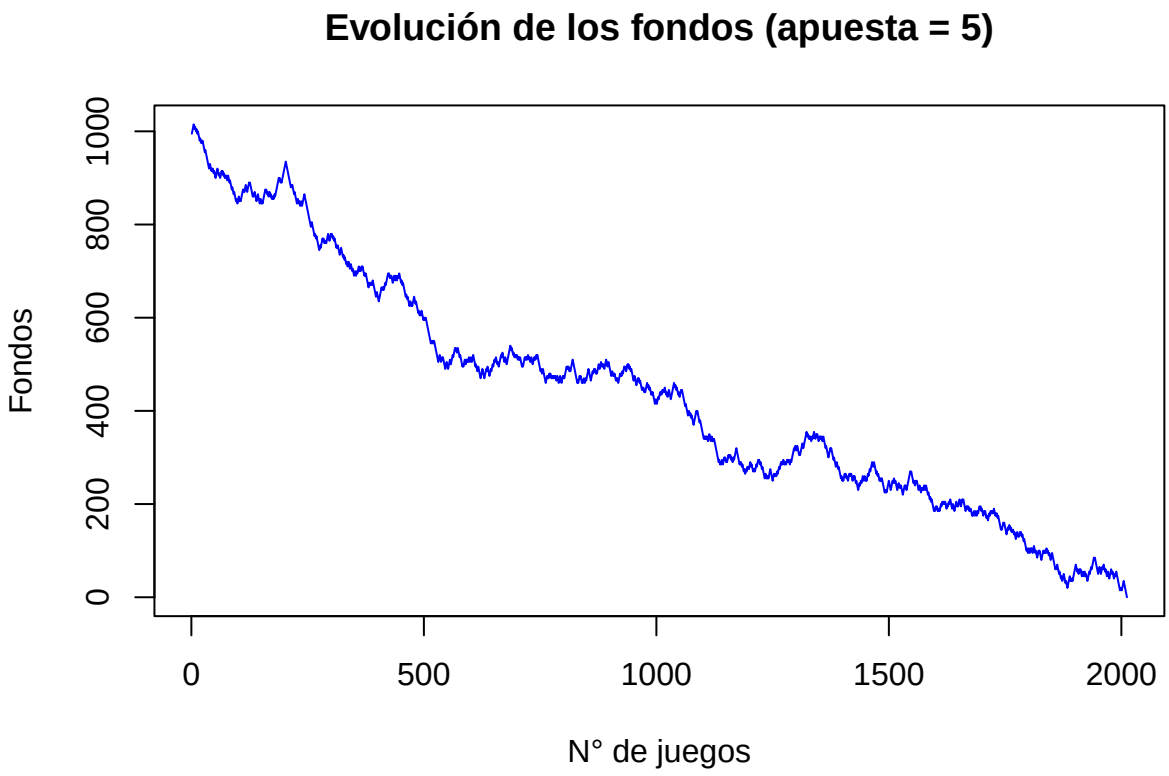
Para realizar el experimento usted decide asumir las siguientes declaraciones, bajo sus observaciones:

- La probabilidad de ganar en un juego del casino es 9/19
- Sabe que su amigo posee fondos en el rango de 0 a 200 dolares.
- Las apuestas como mínimo deben ser igual a 5 dolares.
- El monto de las apuestas no cambia y son siempre igual a la primera. Por ejemplo, si su amigo apuesta 50 dolares, todos los próximos juegos apuesta 50 hasta que se acaba su dinero.
- Asuma que al momento de ganar el jugador anexa el valor apostado a sus fondos.

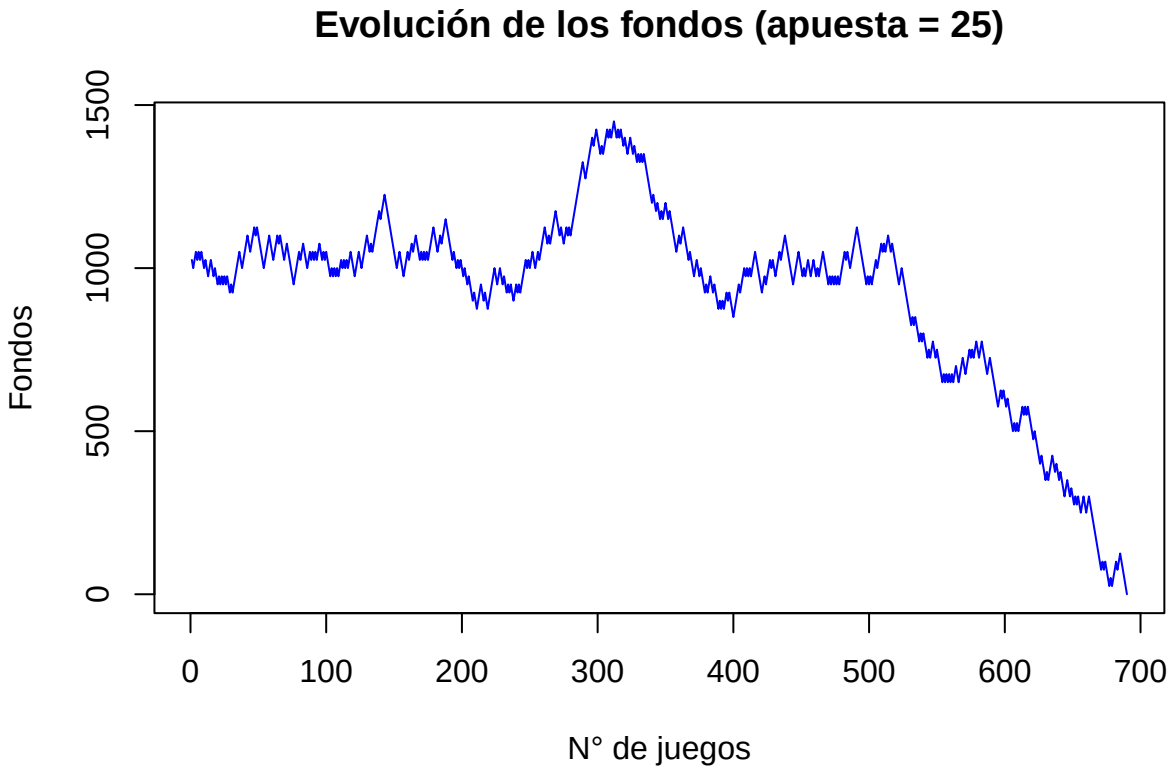
En el experimento deberá obtener la evolución de los fondos hasta que el jugador se queda sin fondos para jugar. Puede ser útil seguir la lógica de una moneda cargada para realizar esto. Pruebe esto con una apuesta igual a 5, 25 y 50 graficando los resultados. Comente los resultados obtenidos.

Respuesta

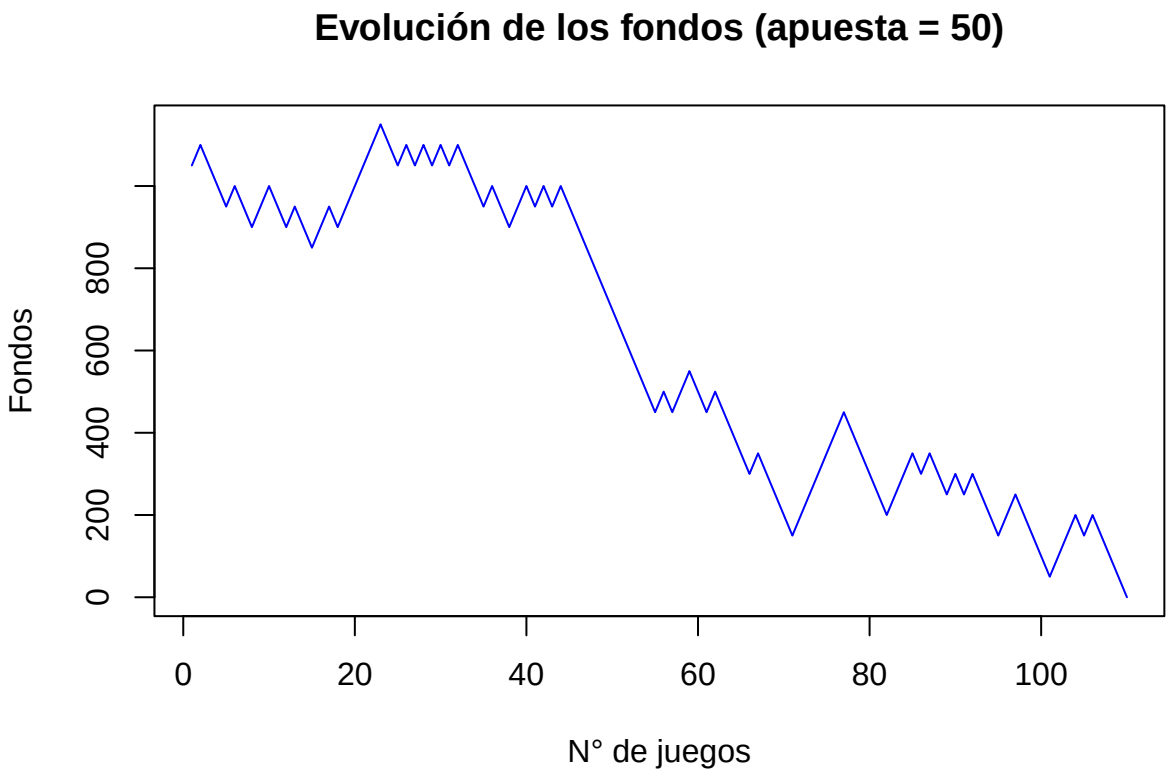
```
# Función para obtener el desarrollo de las apuestas
ruina <- function(fondos = 1000, apuesta = 5){
  win_prob <- 9/19
  vec_fondos <- c()
  while (0<fondos & fondos<2000) {

    dado <- runif(1, 0, 1)
    fondos <- fondos + ((dado<win_prob) - (dado>=win_prob))*apuesta
    vec_fondos <- rbind(vec_fondos, fondos)

  }
  return(vec_fondos) # Devuelve un vector con el desarrollo de los fondos
}

plot(ruina(), type="l", col="blue", xlab="N° de juegos", ylab="Fondos", main="Evolución de los fondos (a
```




```
plot(ruina(apuesta = 25), type="l", col="blue", xlab="N° de juegos", ylab="Fondos", main="Evolución de l
```




```
plot(ruina(apuesta = 50), type="l", col="blue", xlab="N° de juegos", ylab="Fondos", main="Evolución de l
```



Viendo los resultados, y repitiendo el experimento varias veces y con diferentes variables, podemos observar que si la probabilidad de ganar la apuesta es menor a un 50%, es altamente probable que ha medida que vayan ocurriendo más juegos la persona irá perdiendo más dinero.

A work by CC6104