Guia 15 LGN y TCL

Agustín Muñoz González

15/6/2020

Preparamos el entorno

```
rm(list=ls())
library(ggplot2)
library(tidyr)
library(gganimate)
```

1. Comprobación empírica de la LGN: generaremos muestras aleatorias de una distribución dada, de tamaño cada vez mayor, y calcularemos las medias muestrales. Veremos que estos promedios muestrales se acercan al valor verdadero de la media cuando el tamaño muestral crece.

(a) Probamos primero con una distribución $\mathcal{U}(0,1)$. Indique cuál es el valor verdadero de la media μ .

Resolución:

El verdadero valor de la media es $\frac{1+0}{2} = \frac{1}{2}$.

(b) Genere una muestra de tamaño N = 1000 y para cada k entre 1 y 1000 calculemos el promedio de las primeras k observaciones. Realizamos un scatterplot de k vs. los promedios obtenidos. Incluya una linea horizontal en el valor de y correspondiente a la verdadera media μ.

Resolución:

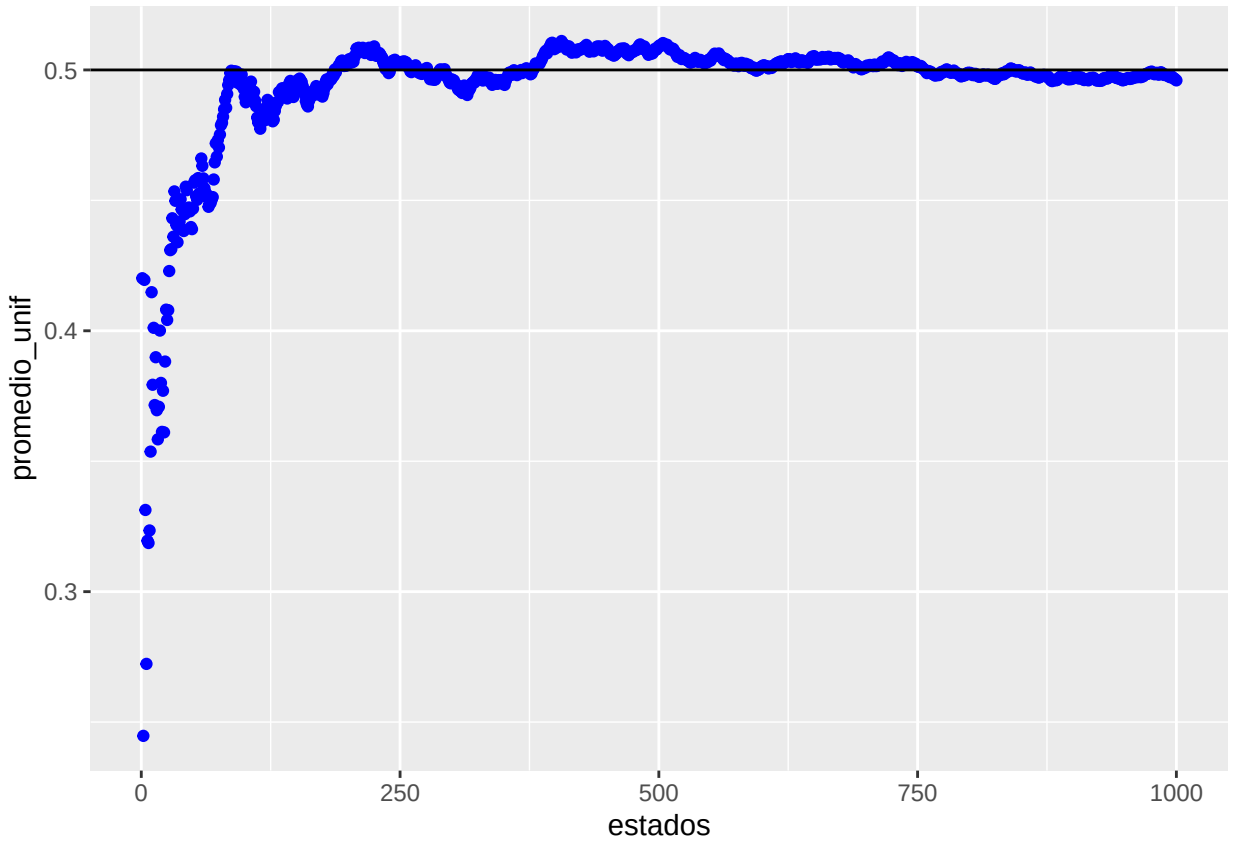
Generamos los datos.

```
Nrep=1000
muestra_unif=runif(Nrep,0,1)
promedio_unif=c()
for(k in 1:Nrep){
  promedio_unif=c(promedio_unif,mean(muestra_unif[1:k]))
}
promedio_unif=as.data.frame(promedio_unif)

# le agregamos otra variable al data.frame para hacer el gif
promedio_unif$estados=1:1000
```

Ploteamos.

```
promedio_unif %>%
  ggplot()+
  geom_point(aes(x=estados,y=promedio_unif),col='blue')+
  geom_hline(aes(yintercept=0.5), col='black')
```



Progresión.

```
anim_unif=promedio_unif %>%
  ggplot()+
  geom_point(aes(x=estados,y=promedio_unif,group = seq_along(estados)),col='blue')+
  geom_hline(aes(yintercept=0.5), col='black')+
  transition_reveal(estados)
# animate(anim_unif,
#         width = 400, height = 400,
#         nframes = 480, fps = 24)
# anim_save("unif.gif", anim_unif)
```

(c) Repetimos b) comenzando con otra semilla y superpongamos el nuevo gráfico utilizando un color diferente. Comparamos los gráficos obtenidos. ¿ Qué podemos concluir?

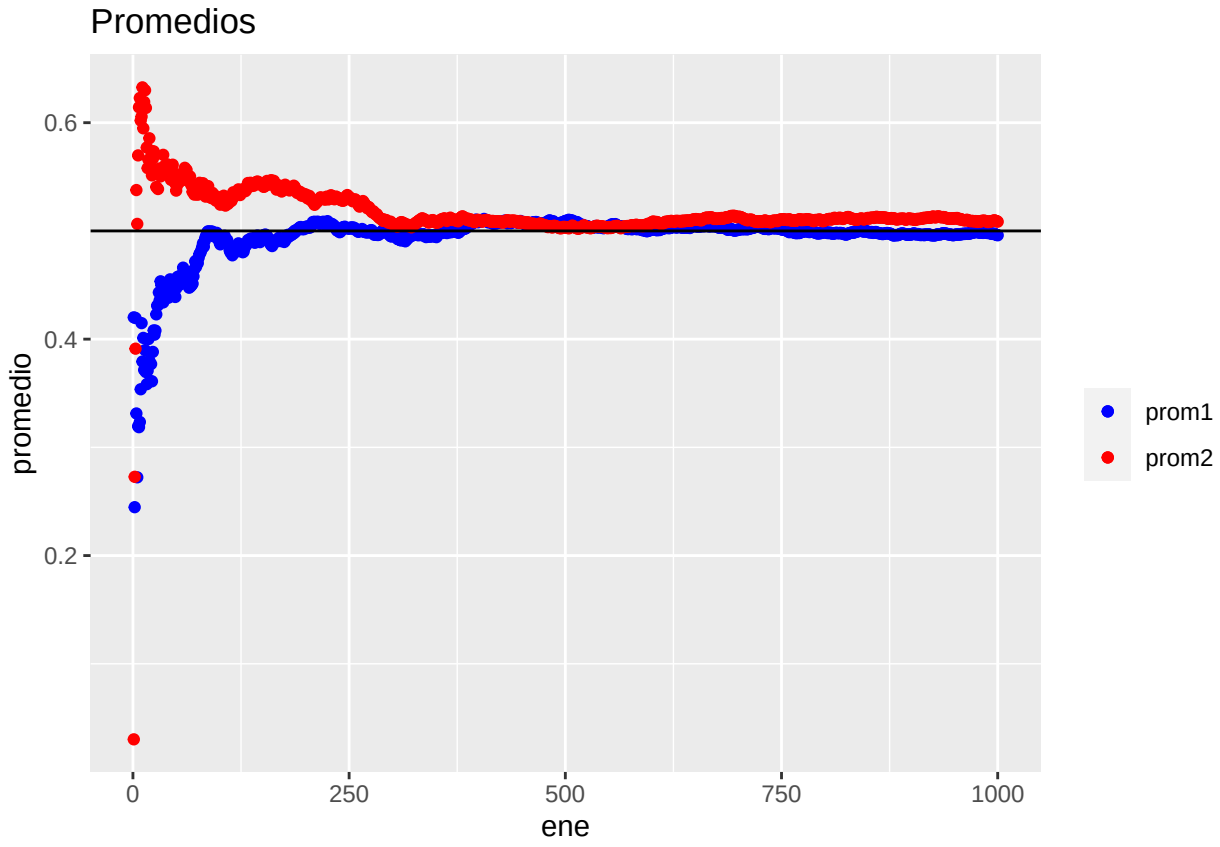
Resolución:

Generamos los nuevos datos.

```
set.seed(7777)
muestra_unif_2=runif(Nrep,0,1)
promedio_unif_2=c()
for(k in 1:Nrep){
  promedio_unif_2=c(promedio_unif_2,mean(muestra_unif_2[1:k]))
}
promedio_unif_2=as.data.frame(promedio_unif_2)
promedios_unif=cbind(promedio_unif,promedio_unif_2)
```

Ploteamos.

```
promedios_unif %>%
  ggplot()+
  geom_point(aes(x=estados,y=promedio_unif,col='prom1'))+
  geom_point(aes(x=estados,y=promedio_unif_2,col='prom2'))+
  geom_hline(aes(yintercept=0.5), col='black')+
  scale_colour_manual("",
                      breaks = c("prom1", "prom2"),
                      values = c("blue", "red")) +
  xlab("ene") +
  scale_y_continuous("promedio") +
  labs(title="Promedios")
```



Ploteamos con progresión.

```
anim_unif_2=promedios_unif %>%
  ggplot()+
  geom_point(aes(x=estados,y=promedio_unif,
                 group = seq_along(estados),
                 col='prom1'))+
  geom_point(aes(x=estados,y=promedio_unif_2,
                 group = seq_along(estados),
                 col='prom2'))+
  geom_hline(aes(yintercept=0.5), col='black')+
  transition_reveal(estados)+
  scale_colour_manual("",
                      breaks = c("prom1", "prom2"),
                      values = c("blue", "red")) +
```

```
  xlab("ene") +
  scale_y_continuous("promedio") +
  labs(title="Promedios")
# animate(anim_unif_2,
#         width = 400, height = 400,
#         nframes = 480, fps = 24)
# anim_save("unif.gif", anim_unif_2)
```

Generamos muestras de tamaño k cada vez mayor k entre 1 y 1000, calculamos sucesivamente los promedios y realizamos un scatter plot de k vs. los promedios obtenidos.

(d) Probamos ahora con una distribución normal $\mathcal{N}(3,4)$. Generamos muestras de tamaño 10^k cada vez mayor, k = 0 a 7, calculamos sucesivamente los promedios y realizamos un scatter plot de k vs. los promedios obtenidos.

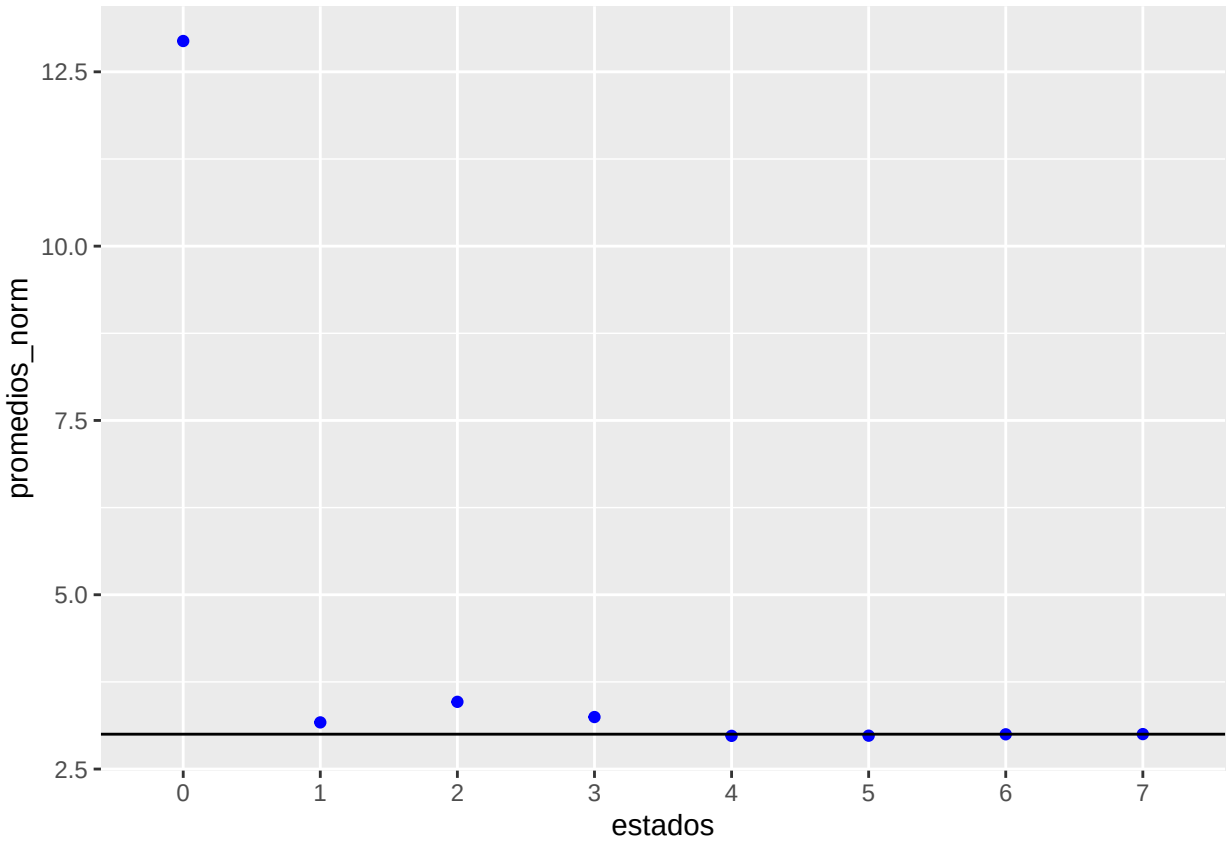
Resolución:

Generamos los datos.

```
K=7
promedios_norm=estados=c()
for(i in 0:K){
  aux=rnorm(10^i,3,4)
  promedios_norm=c(promedios_norm,mean(aux))
  estados=c(estados,i)
}
datos=as.data.frame(cbind(promedios_norm,estados))
datos$estados=as.factor(datos$estados)
```

Ploteamos.

```
datos %>%
  ggplot()+
  geom_point(aes(x=estados,y=promedios_norm),col='blue')+
  geom_hline(aes(yintercept=3),color='black')
```



(e) Repetimos el item anterior comenzando con otra semilla, superponiendo el gráfico con otro color, y comparamos los gráficos obtenidos. ¿ Qué podemos concluir?

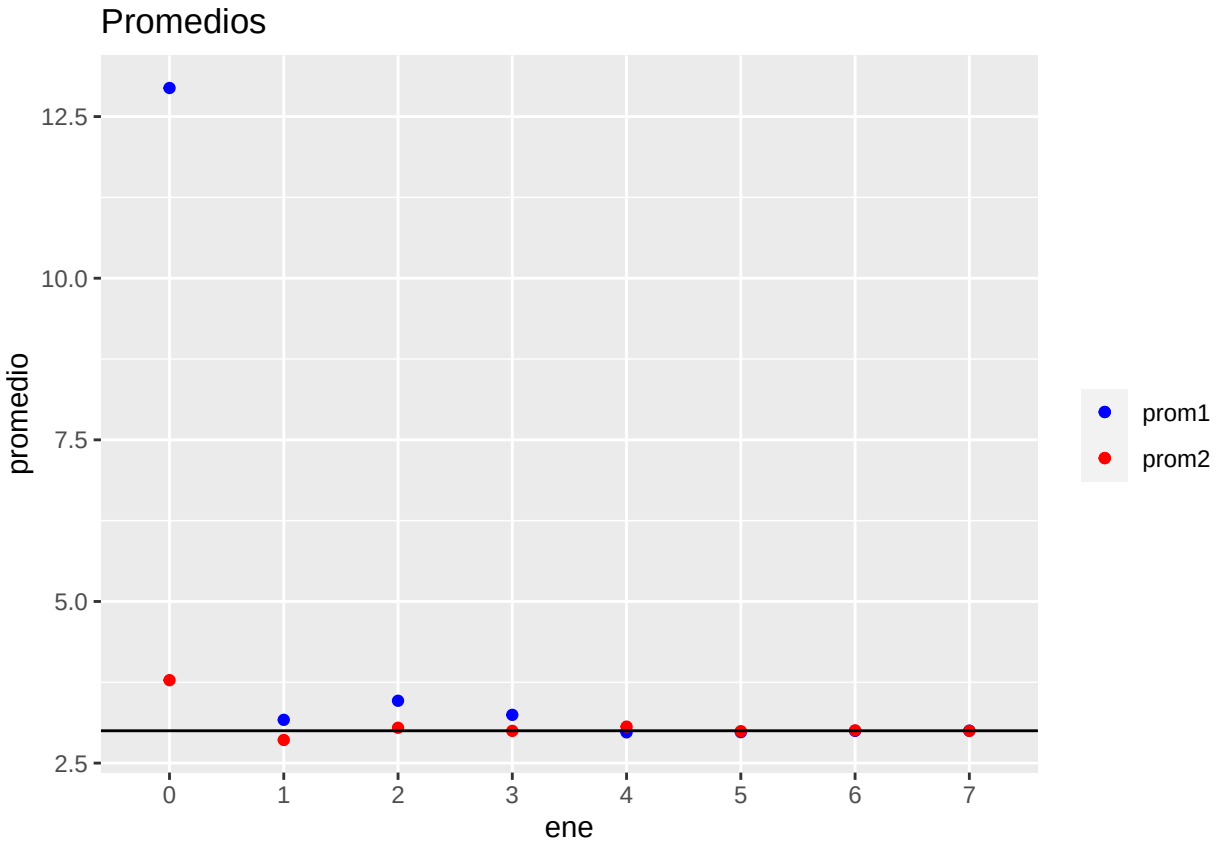
Resolución:

Generamos los datos.

```
promedios_norm_2=c()
set.seed(8888)
for(i in 0:K){
  aux=rnorm(10^i,3,4)
  promedios_norm_2=c(promedios_norm_2,mean(aux))
  }
datos=as.data.frame(cbind(datos,promedios_norm_2))
```

Ploteamos.

```
datos %>%
  ggplot()+
  geom_point(aes(x=estados,y=promedios_norm,col='prom1'))+
  geom_point(aes(x=estados,y=promedios_norm_2,col='prom2'))+
  geom_hline(aes(yintercept=3),color='black')+
  scale_colour_manual("",
                      breaks = c("prom1", "prom2"),
                      values = c("blue", "red")) +
  xlab("ene") +
  scale_y_continuous("promedio") +
  labs(title="Promedios")
```



(f) Generamos ahora datos con una distribución de Cauchy. Recordemos que la distribución de Cauchy corresponde a una distribución t de student con un grado de libertad, con densidad dada por

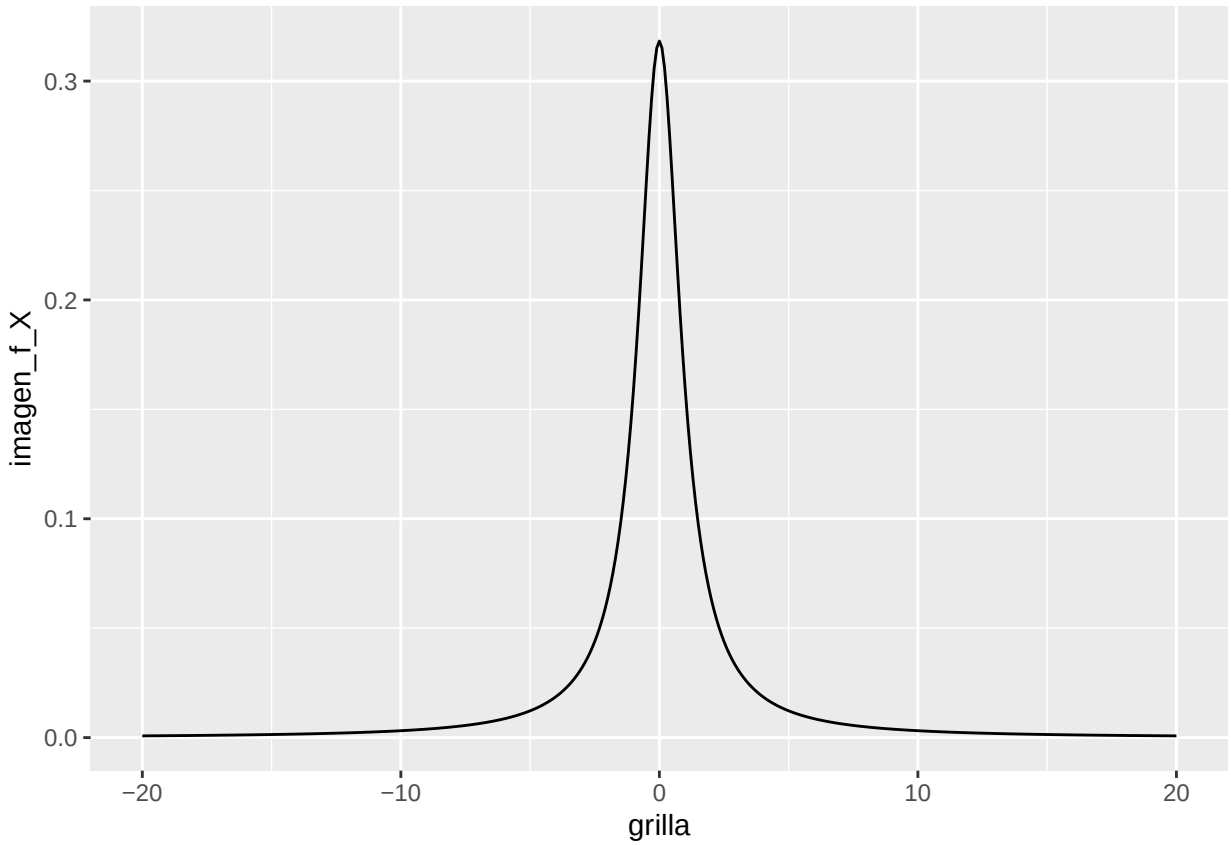
$$f_X(x) = \frac{1}{\pi(1+x^2)},$$

que es una densidad simétrica alrededor del cero, con colas que acumulan más probabilidad que la normal estándar, y que no tiene esperanza ni varianza finitas. Generamos muestras de tamaño cada vez mayor 10^k , k = 0 a 7, calculamos sucesivamente los promedios y realizamos un scatter plot de k vs. los promedios obtenidos. Comparemos con los resultados obtenidos en los ítems anteriores.

Resolución:

Grafiquemos primero f_X.

```
f_X=function(x){
  1/(pi*(1+x^2))
}
grilla=seq(-20,20,0.1)
imagen_f_X=f_X(grilla)
plot_f_X=data.frame(cbind(grilla,imagen_f_X))
ggplot(plot_f_X,aes(x=grilla,y=imagen_f_X),col='blue')+
  geom_line()
```

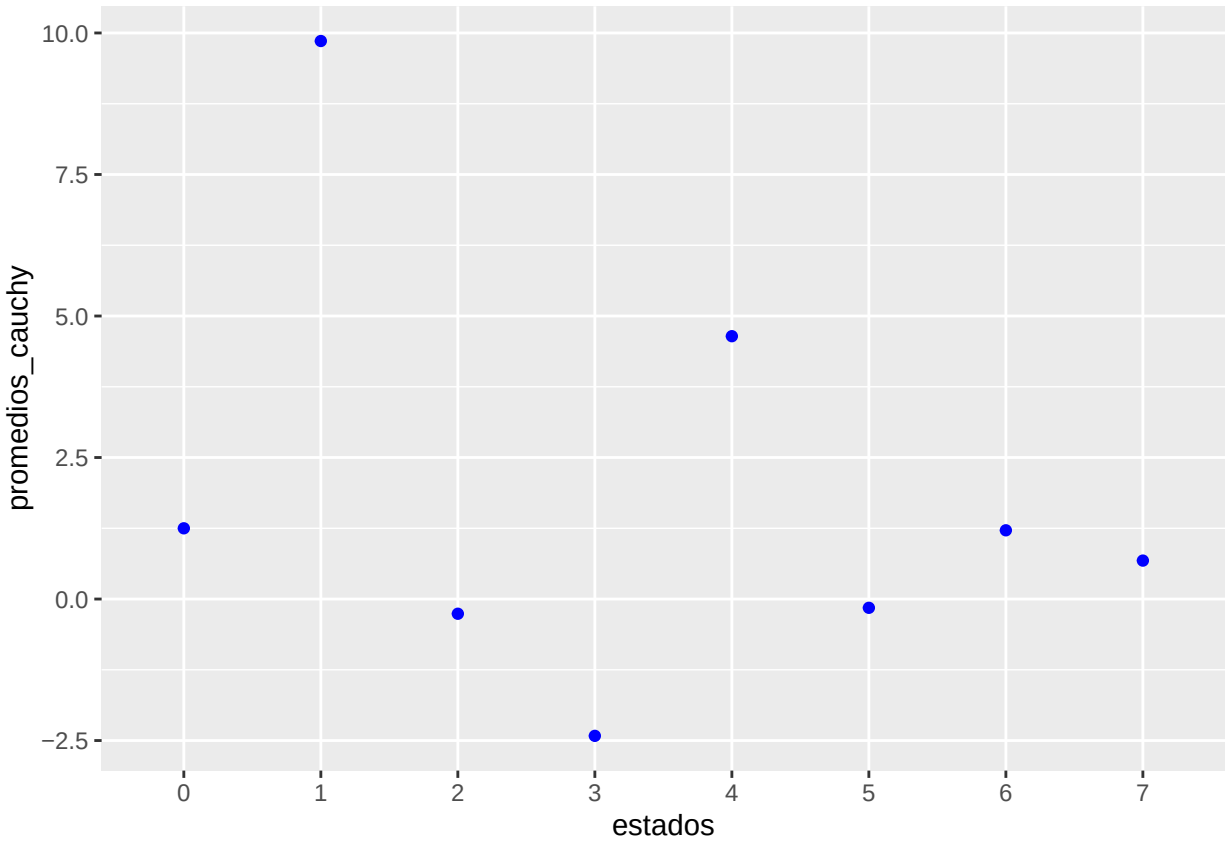


Generamos los datos teniendo en cuenta que la función f_X es la función de densidad de una variables $\mathcal{C}(0,1)$.

```
K=7
promedios_cauchy=estados=c()
for(i in 0:K){
  aux=rcauchy(10^i,0,1)
  promedios_cauchy=c(promedios_cauchy,mean(aux))
  estados=c(estados,i)
}
datos_cauchy=as.data.frame(cbind(promedios_cauchy,estados))
datos_cauchy$estados=as.factor(datos_cauchy$estados)
```

Ploteamos.

```
datos_cauchy %>%
  ggplot()+
  geom_point(aes(x=estados,y=promedios_cauchy),col='blue')
```


10.0
7.5
5.0
2.5
0.0
−2.5
promedios_cauchy
0
1
2
3
4
5
6
7
estados

2. En este ejercicio estudiaremos la distribución del promedio $\overline{X}_n$ de variables $X_1, \ldots, X_n$ independientes e idénticamente distribuidas (i.i.d.), definido por n

$$\overline{X}_n = \frac{1}{n} \sum_i X_i.$$

A través de los correspondientes histogramas analizaremos el comportamiento de la distribución del promedio $\overline{X}_n$, a medida que promediamos un número creciente de variables aleatorias (n aumentando). Es decir, trataremos de validar empíricamente los resultados de la Ley de los Grandes Números y el Teorema Central del Límite. Para ello, fijado n, generaremos datos correspondientes a una muestra $X_1, \ldots, X_n$ i.i.d. de variables aleatorias distribuídas como X, con una distribución dada y luego calcularemos el promedio de cada conjunto de datos. Repetimos este procedimiento $N_{rep} = 1000$ veces. A partir de las $N_{rep} = 1000$ replicaciones realizaremos un histograma con los promedios generados, para obtener una aproximación de la densidad o la función de probabilidad de $\overline{X}_n$.

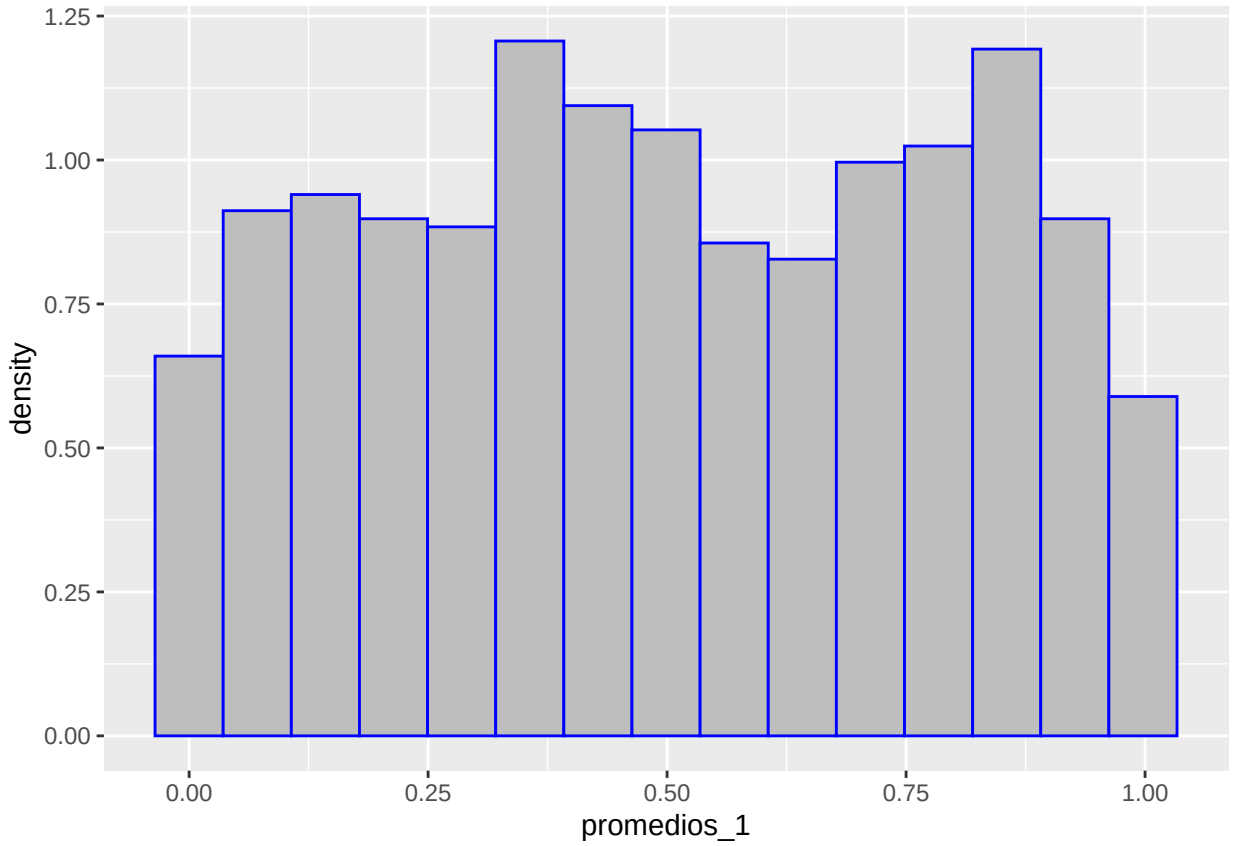
(a) Consideremos n = 1: la variable coincide con el promedio. Generamos entonces N rep = 1000 datos correspondientes a $X \sim \mathcal{U}(0;1)$ y luego hacemos un histograma. ¿A qué densidad se parece el histograma obtenido?

Resolución:

Defino primero una función que simula Nrep valores de n variables $\mathcal{U}(0,1)$.

```
var.gen=function(n,Nrep){
   tabla=c()
   for(i in (1:n)){
      tabla=cbind(tabla,runif(Nrep,0,1))
   }
   data.frame(tabla)
}
var.gen.norm=function(n,Nrep){
   tabla=c()
   for(i in (1:n)){
      tabla=cbind(tabla,rnorm(Nrep,3,4))
   }
   data.frame(tabla)
}
```

```
Nrep=1000
muestra_1=var.gen(1,Nrep)
promedios_1=data.frame('promedios_1'=apply(muestra_1,1,mean))
ggplot(promedios_1)+
  geom_histogram(aes(x=promedios_1,y=..density..),
                 colour='blue',fill='grey',
                 bins=15)
```




```
                    #binwidth = 0.05)
```

(b) Consideramos n = 2 variables aleatorias X_1 y X_2 independientes con distribución $\mathcal{U}(0;1)$ y el promedio de ambas, es decir,

$$\overline{X}_2 = \frac{X_1 + X_2}{2}.$$

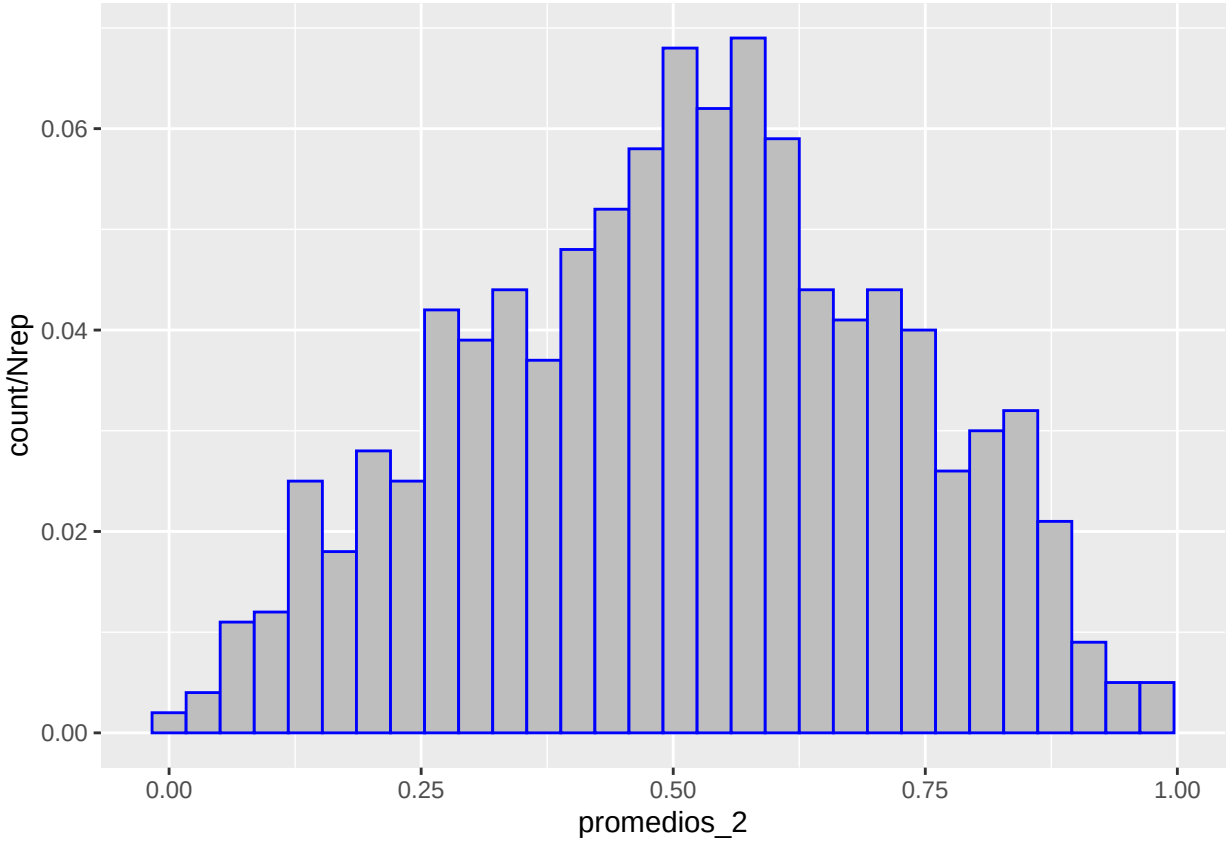
Genermos n = 2 datos (independientes) correspondientes a una variable aleatorias con distribución $\mathcal{U}(0;1)$ y computamos el promedio. Replicamos N rep = 1000 veces y realizamos un histograma con los N rep = 1000 promedios obtenidos. ¿Qué características tiene este histograma?

Resolución:

Simulamos los datos y ploteamos.

```
muestra_2=var.gen(2,Nrep)
promedios_2=data.frame('promedios_2'=apply(muestra_2,1,mean))
ggplot(promedios_2,aes(x=promedios_2,y=stat(count)/Nrep))+
  geom_histogram(colour='blue',fill='grey')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



El histograma se va pareciendo a una campana.

(c) Aumentamos a n = 5 las variables promediadas. Consideramos ahora 5 variables aleatorias uniformes independientes, es decir $X_1, \ldots, X_5$ i.i.d. con $X_i \sim \mathcal{U}(0;1)$ y definimos elpromedio de las mismas

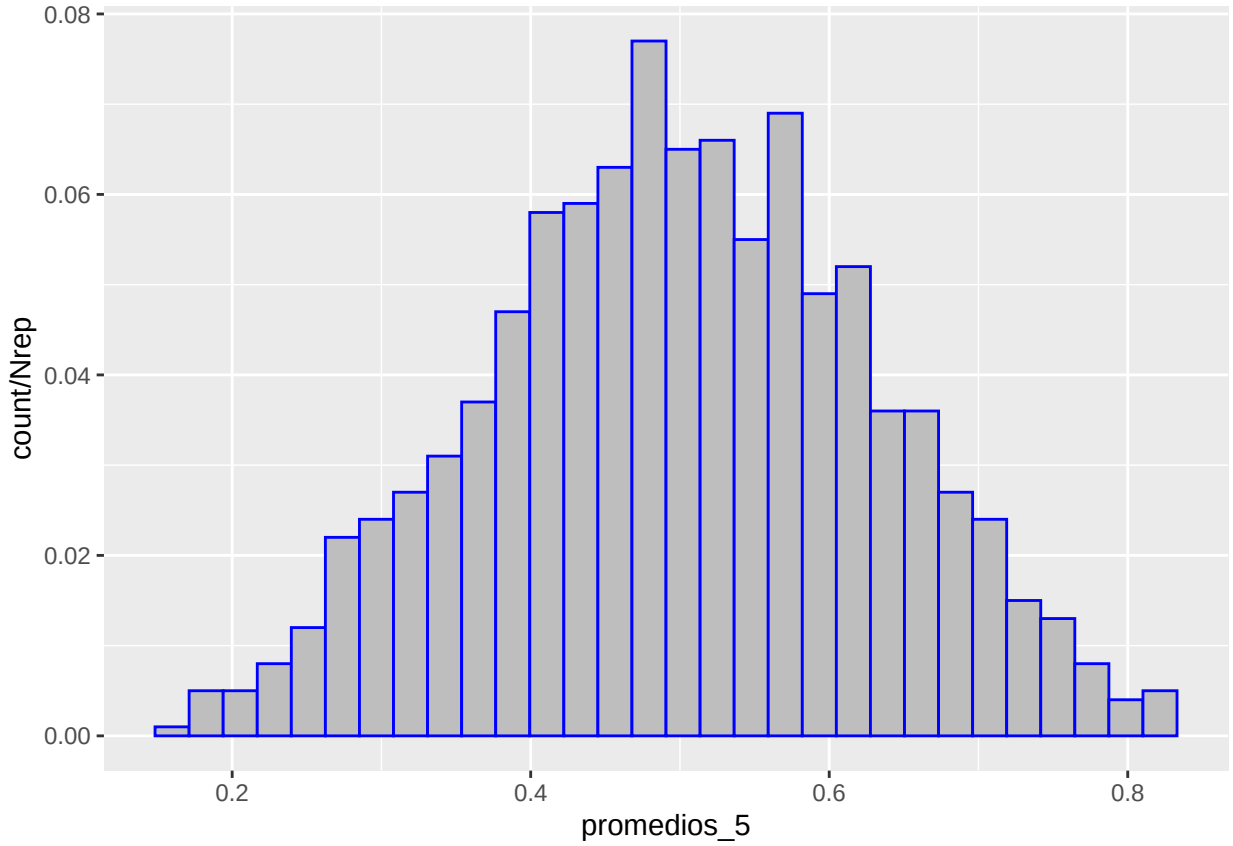
$$\overline{X}_5 = \frac{X_1 + \cdots + X_5}{5}.$$

Generamos datos de 5 variables aleatorias con distribución $\mathcal{U}(0;1)$ computamos el promedio. Repetimos N rep = 1000 veces y realizamos un histograma para los valores obtenidos. Comparamos con el histograma anterior. ¿Qué se observa?

Resolución:

```
muestra_5=var.gen(5,Nrep)
promedios_5=data.frame('promedios_5'=apply(muestra_5,1,mean))
ggplot(promedios_5,aes(x=promedios_5,y=stat(count)/Nrep))+
  geom_histogram(colour='blue',fill='grey')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

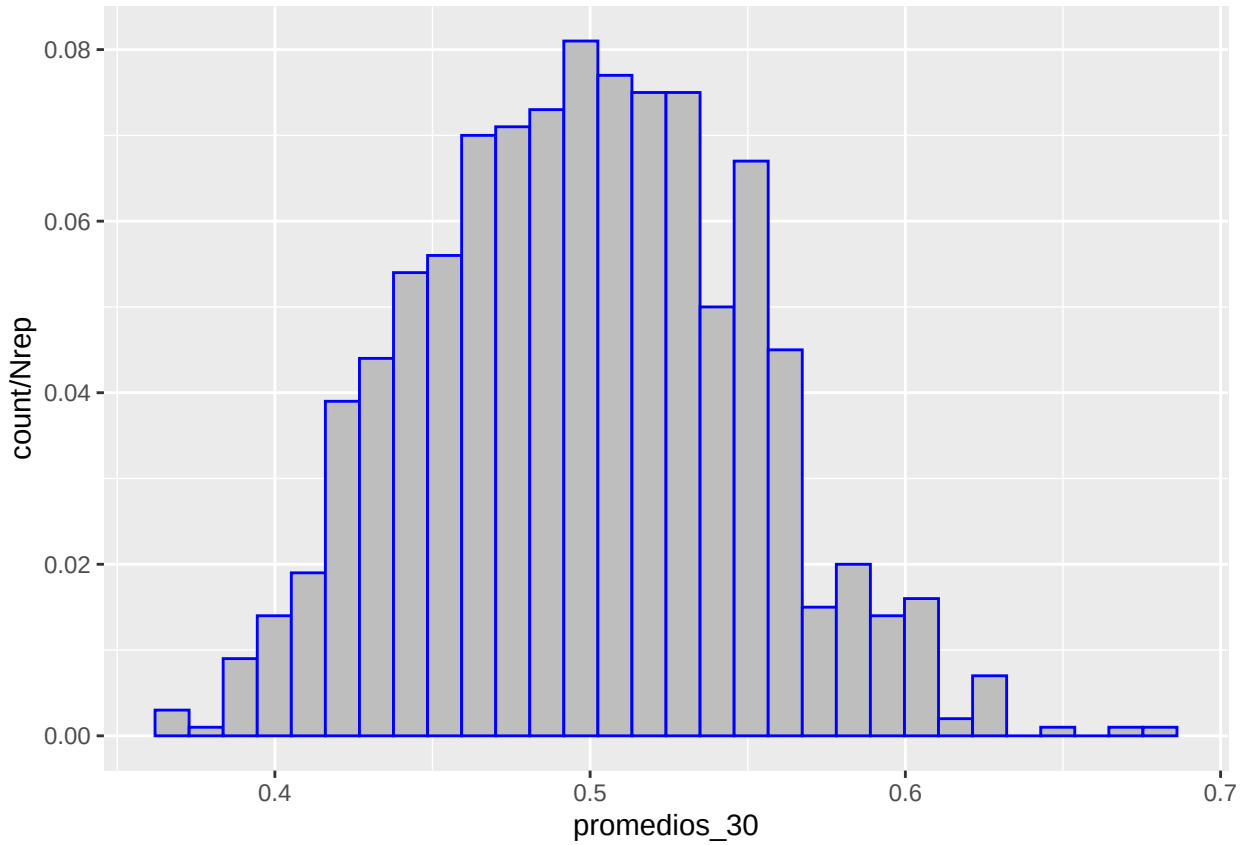


(d) Aumentamos aún más la cantidad de variables promediadas. Generando n = 30 datos con distribución $\mathcal{U}(0;1)$ repetimos el ítem anterior. ¿Qué se observa?

Resolución:

```
muestra_30=var.gen(30,Nrep)
promedios_30=data.frame('promedios_30'=apply(muestra_30,1,mean))
ggplot(promedios_30,aes(x=promedios_30,y=stat(count)/Nrep))+
  geom_histogram(colour='blue',fill='grey')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



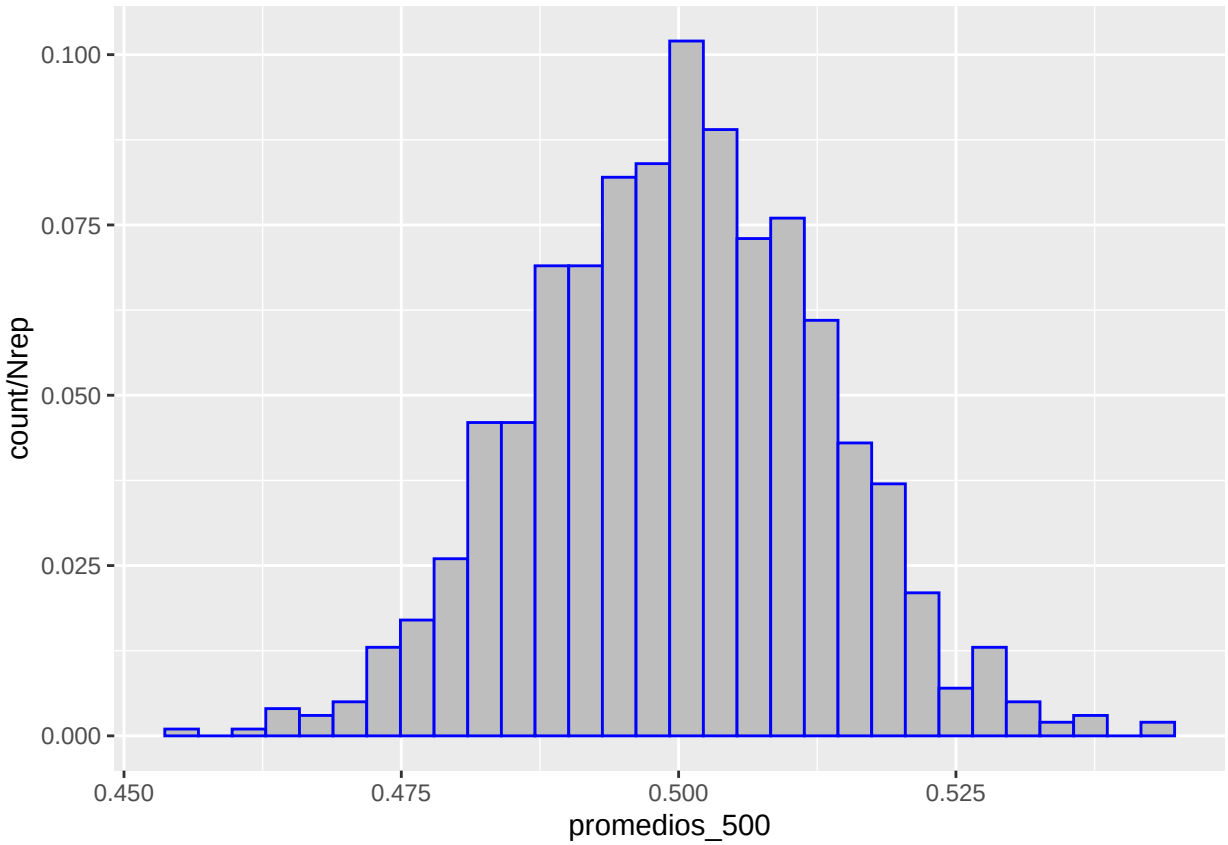
(e) Idem anterior generando n = 500 datos. ¿Qué pasa si aumentamos el tamaño de la muestra?

Resolución:

Graficamos solo este caso.

```
muestra_500=var.gen(500,Nrep)
promedios_500=data.frame('promedios_500'=apply(muestra_500,1,mean))
ggplot(promedios_500,aes(x=promedios_500,y=stat(count)/Nrep))+
  geom_histogram(colour='blue',fill='grey')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Vamos a graficar ahora la progresión para n en c(1,2,5,30,500). Lo haremos de dos formas, una generando nuevos datos y otra juntando y usando los datos anteriores.

Con lo datos de antes.

Una forma:

```
enes=c(1,2,5,30,500)
promedios=matrix()
estado=c()
for(i in enes){
  aux=get(paste('promedios_',i,sep=""))
  promedios=cbind(promedios,aux)
  estado=c(estado,rep(i, Nrep))
}
# le saco la columna 0 que tiene NAs
promedios=promedios[,1:length(enes)+1]
promedios=stack(promedios)[1]
names(promedios)[1]="promedios"
promedios$estado=estado
promedios$estado=as.factor(promedios$estado)
```

Muestro otra forma de hacerlo a partir de los datos. Solo la pongo porque queria saber como se hacia pero no sirve para este caso porque como hay que ponerle Nrepˆenes etiquetas el programa tarda mucho e incluso se cuelga.

```
# muestras=matrix()
# estado=c()
# for(i in enes){
```

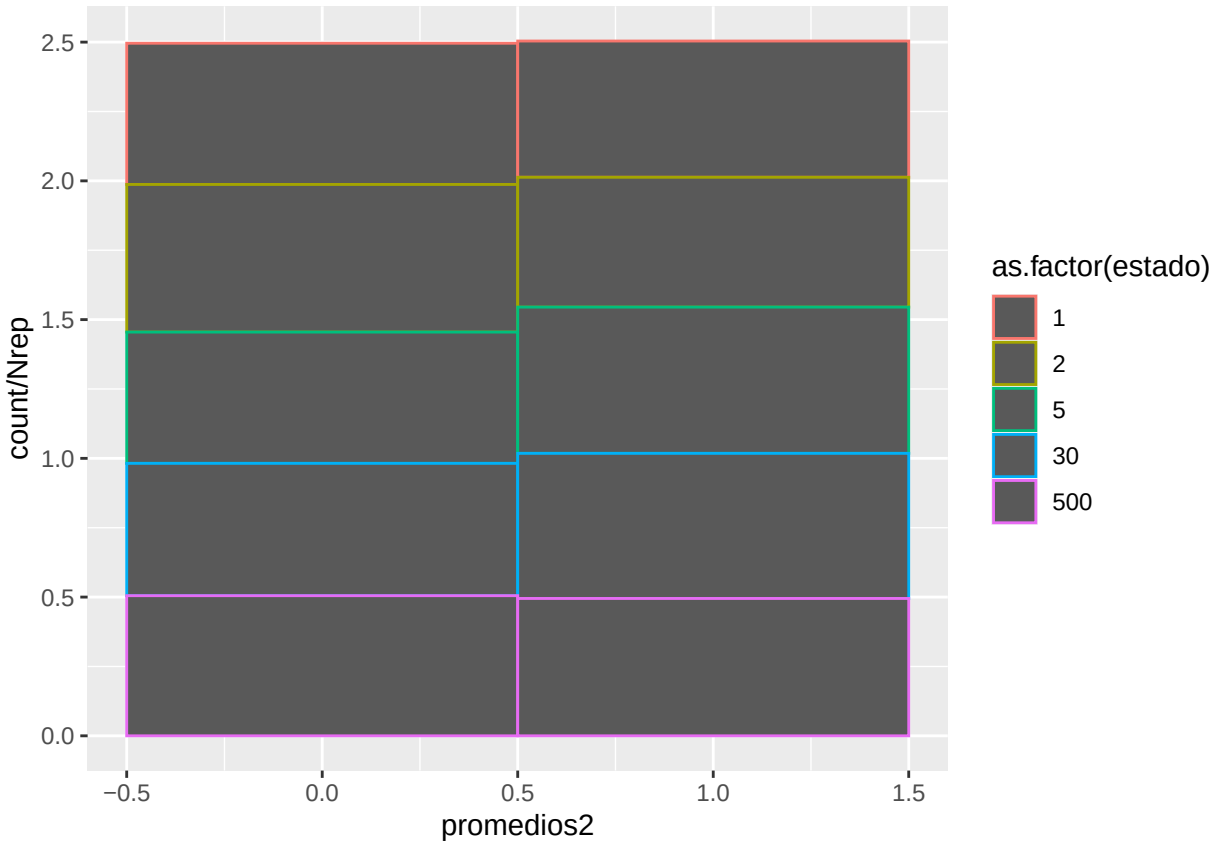
```
#   aux=get(paste('muestra_',i,sep=""))
#   muestras=cbind(muestras,aux)
#   estado=c(estado,rep(i, Nrep))
# }
# le saco la columna 0 que tiene NAs
# muestras=muestras[,1:length(enes)+1]
# muestras=stack(muestras)[1]
# muestras$estado=rep(enes,Nrep^enes)
# muestras$estado=as.factor(muestras$estado)
# for(i in enes){
#   promedio=c(promedio,apply(muestra[muestra$estado==i],1,mean))
# }
```

Ploteamos.

```
anim_prom <- promedios %>%
   ggplot()+
   geom_histogram(aes(x=promedios,y=stat(count)/Nrep),
                  colour ='blue',fill='gray',binwidth=1 )+
   transition_states(estado, transition_length = 1, state_length = 1)
# animate(anim_prom,
#         width = 400, height = 400,
#         nframes = 480, fps = 24)
# anim_save("promedios.gif", anim_prom)
```

Con nuevos datos.

```
enes=c(1,2,5,30,500)
promedios2=estado=c()
for(i in enes){
  aux=var.gen(i,Nrep)
  promedios2=c(promedios2,apply(aux,1,mean))
  estado=c(estado,rep(i, Nrep))
}
promedios2=data.frame('promedios2'=promedios2)
promedios2$estado=estado
anim_prom2 <- promedios2 %>%
   ggplot()+
   geom_histogram(aes(x=promedios2,y=stat(count)/Nrep),
                  colour ='blue',fill='gray',binwidth=1,bins=30 )+
   transition_states(estado, transition_length = 1, state_length = 1)
promedios2 %>%
ggplot(aes(color=as.factor(estado)))+
geom_histogram(aes(x=promedios2,y=stat(count)/Nrep),
binwidth=1 )
```




```
# animate(anim_prom2,
#         width = 400, height = 400,
#         nframes = 480, fps = 24)
# anim_save("promedios2.gif", anim_prom2)
```

XQ SI LOS HISTOGRAMAS TIENEN ESA FORMA CUANDO HAGO LA TRANSICION ME DIBUJA OTRA COSA?

Observemos que para poder comparar los histogramas de los distintos conjuntos de datos será necesario tenerlos dibujados en la misma escala tanto para el eje horizontal como para el vertical. Por eso, en general es más cómodo hacer boxplots para comparar distintos conjuntos de datos.

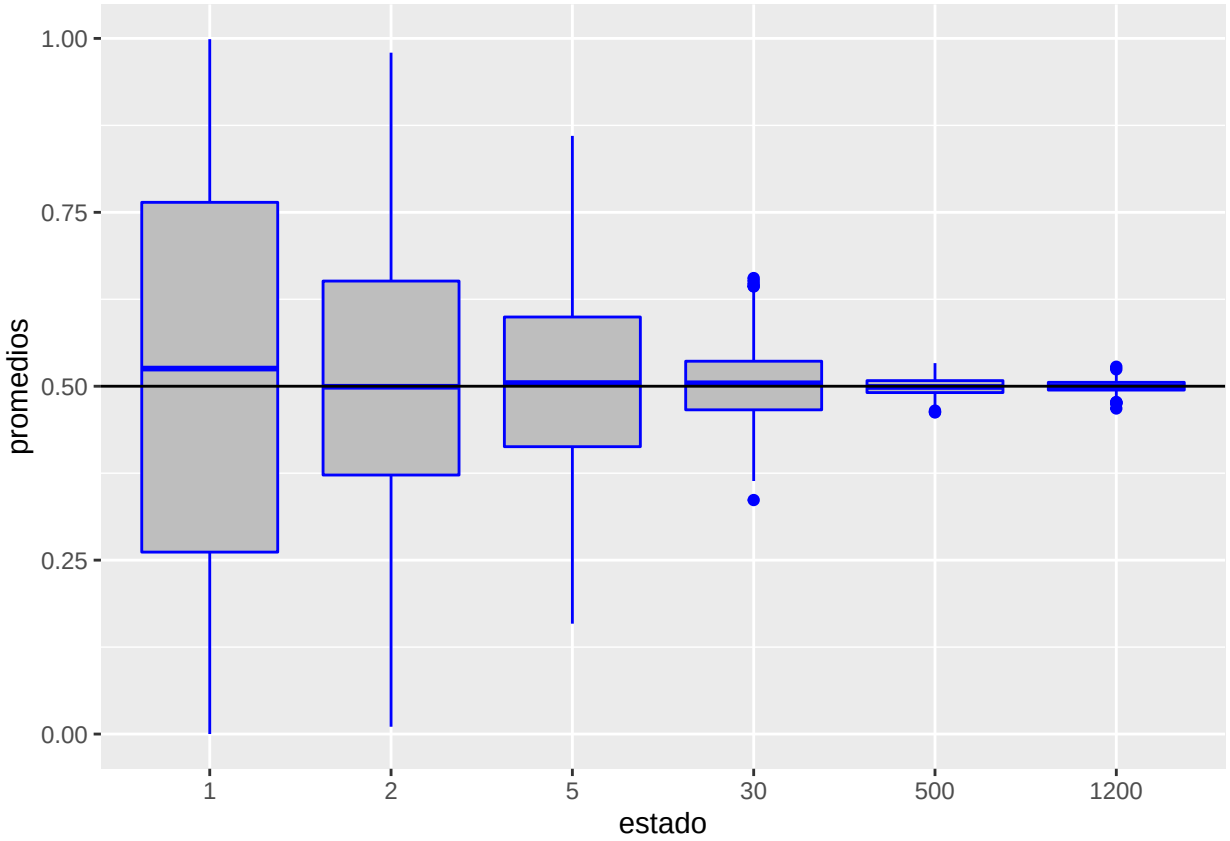
(f) Finalmente, lo hacemos también para n = 1200, y graficamos un boxplot de los 6 conjuntos de datos en el mismo gráfico. En este gráfico se verá que a medida que aumenta el n, o sea el tamaño muestral, los valores de los promedios tienden a concentrarse, ¿alrededor de qué valor? Calculamos media y varianza muestral para cada conjunto de datos. ¿A qué valores teóricos deberían parecerse?

Resolución:

Generamos los datos y ploteamos.

```
enes=c(1,2,5,30,500,1200)
promedios=estado=c()
for(i in enes){
  aux=var.gen(i,Nrep)
  promedios=c(promedios,apply(aux,1,mean))
  estado=c(estado,rep(i, Nrep))
}
promedios=data.frame('promedios'=promedios)
promedios$estado=estado
```

```r
promedios$estado=as.factor(promedios$estado)
ggplot(promedios,aes(y=promedios,x=estado))+
  geom_boxplot(colour ='blue',fill='gray')+
  geom_hline(aes(yintercept=1/2),color='black')
```

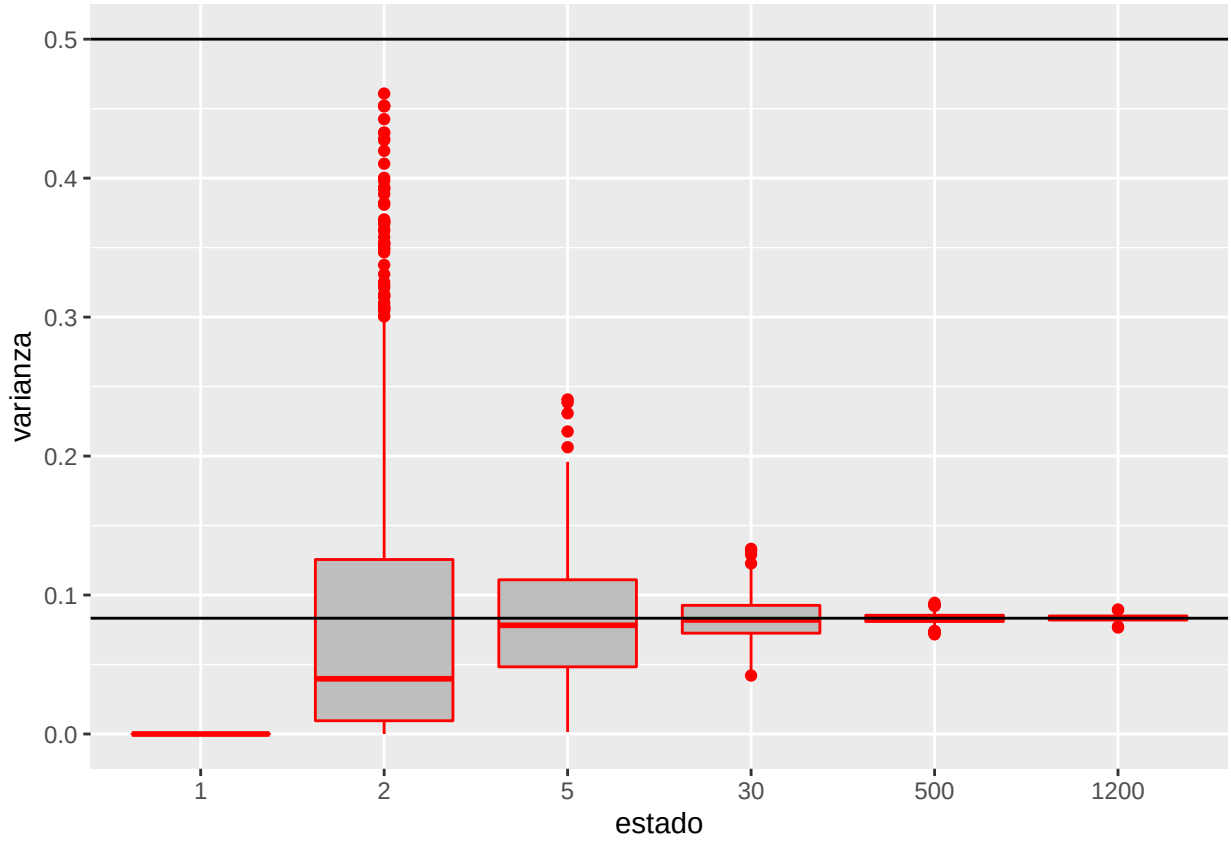


Los datos tienden a concentrarse alrededor del valor $y = 1/2$ que es precisamente el valor de la media de la distribución $\mathcal{U}(0, 1)$.

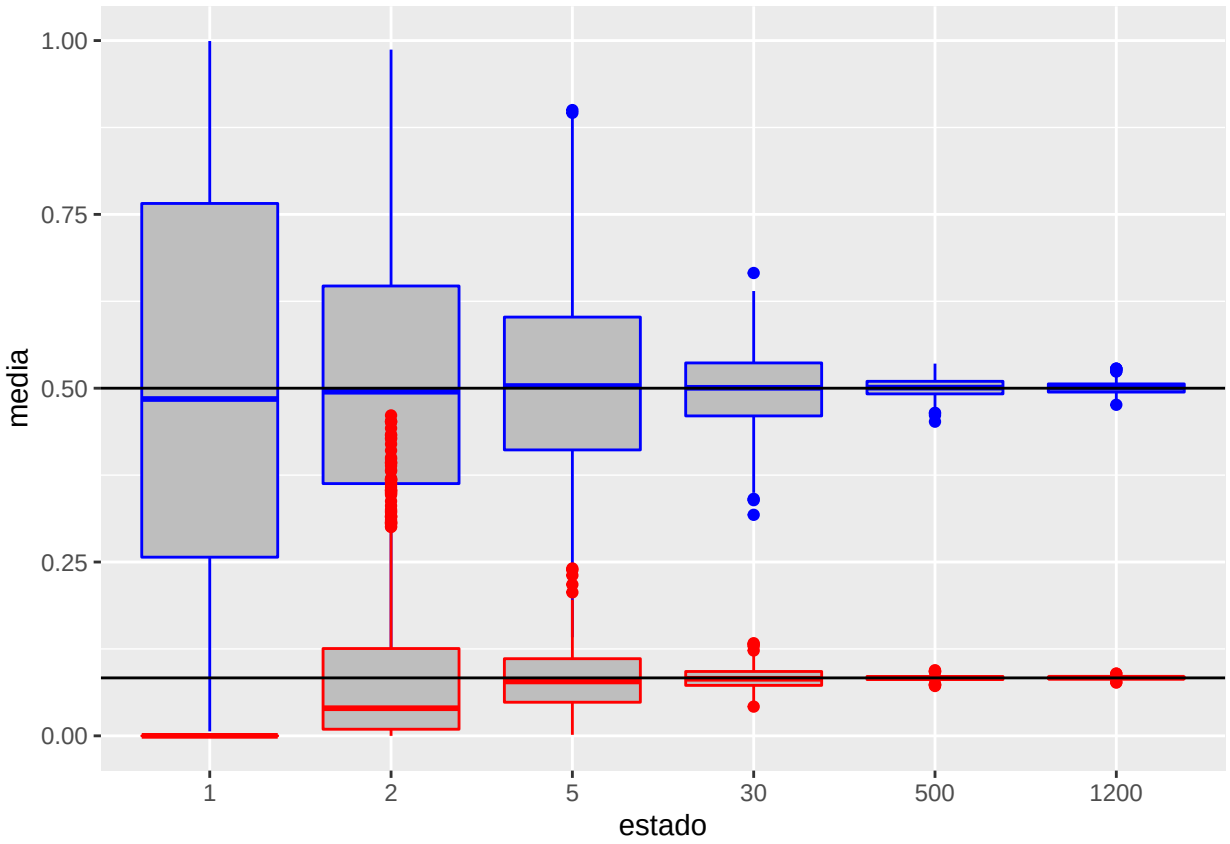
Calculamos media y varianza muestral del conjunto de datos. Es decir, para cada n calculamos media y varianza de la muestra simulada. La media de hecho es el promedio muestral que calculamos y ploteamos antes, lo unico que falta es la varianza.

```r
media_muestral=varianza_muestral=estado=c()
for(i in enes){
  aux=var.gen(i,Nrep)
  media_muestral=c(media_muestral,apply(aux,1,mean))
  if(i==1){
    varianza_muestral=c(varianza_muestral,rep(0,Nrep))
  }else{varianza_muestral=c(varianza_muestral,apply(aux,1,var))}
  # cuando tengo 1 sola v.a. la varianza es 0
  estado=c(estado,rep(i, Nrep))
}
media_muestral=data.frame('media'=media_muestral)
varianza_muestral=data.frame('varianza'=varianza_muestral)
datos_muestral=cbind(media_muestral,varianza_muestral)
datos_muestral$estado=estado
datos_muestral$estado=as.factor(datos_muestral$estado)
datos_muestral %>%
```

```
  ggplot()+
  geom_boxplot(aes(x=estado,y=varianza),colour ='red',fill='gray')+
  geom_hline(aes(yintercept=1/2),color='black')+
  geom_hline(aes(yintercept=1/12),color='black')
```




```
datos_muestral %>%
  ggplot()+
  geom_boxplot(aes(x=estado,y=media),colour ='blue',fill='gray')+
  geom_boxplot(aes(x=estado,y=varianza),colour ='red',fill='gray')+
  geom_hline(aes(yintercept=1/2),color='black')+
  geom_hline(aes(yintercept=1/12),color='black')
```



Estos datos deberían parecerse a la media y varianza de la distribución $\mathcal{U}(0,1)$ que son $\mu = \frac{1+0}{2} = \frac{1}{2} = 0.5$ y $\sigma^2 = \frac{(1-0)^2}{12} = \frac{1}{12} = 0.083$. Efectivamente vemos que los boxplot se concentran en el valor 0.5 y 0.083 (las líneas negras).

(g) El Teorema Central del Límite nos dice que cuando hacemos la siguiente transformación con los promedios,

$$\frac{\overline{X}_n - E(X_1)}{(Var(\overline{X}_n))^{1/2}} = \frac{\overline{X}_n - E(X_1)}{(Var(\overline{X}_1)/n)^{1/2}},$$

la distribución de estas variables aleatorias se aproxima a la de la normal estándar, cuando n es suficientemente grande. Para comprobarlo empíricamente, hagamos esta transformación en los 6 conjuntos de datos (es razonable hacerlo para valores de n suficientemente grandes, lo realizaremos en todos los casos para comparar) y luego comparemos los datos transformados mediante histogramas y boxplots.

Resolución:

(h) Repitamos los ítems anteriores generando ahora variables con distribución Cauchy C(0;1): Comparemos los resultados obtenidos.

Resolución: