Tarea 1 Inferencia Estadística

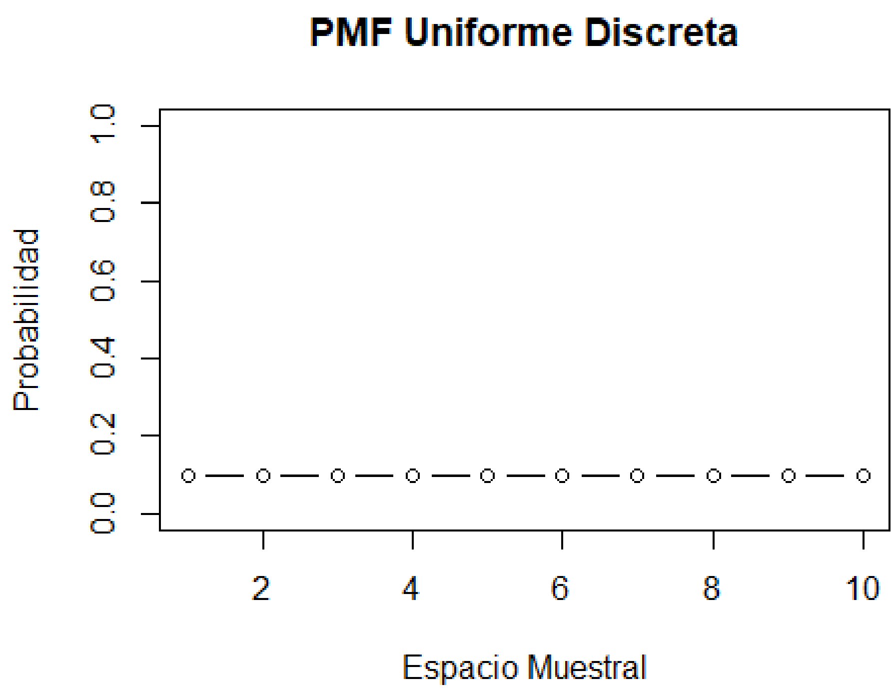
Hairo Ulises Miranda Belmonte

28 de agosto de 2018

5. Para el siguiente ejercicio es necesario el programa R.

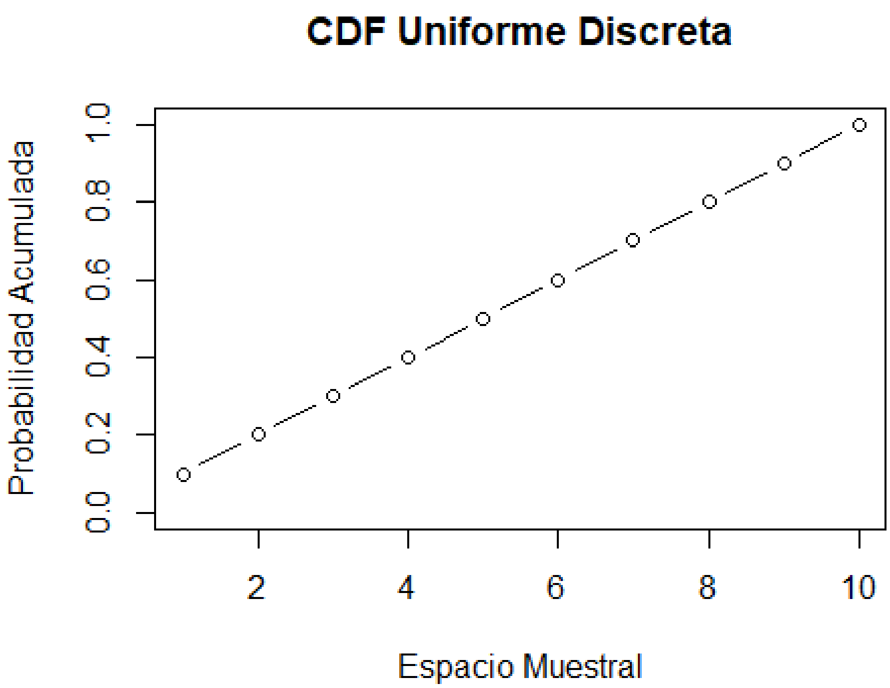
a) Escriba un programa en R que reproduzca las gráficas de las funciones de distribución acumulada y de masa de la distribución uniforme que aparecen en las notas del curso. Las gráficas deben verse similares a las figuras de la Figura 1.

```
rm(list=ls())
pmf<-0
for(i in seq(1,10,1)){ #generando los lanzamientos
  pmf[i]<-dunif(i,min = 0,max = 10, log = FALSE)
}
SampleSpace<-1:10
plot(SampleSpace,pmf, type = "b", main="PMF Uniforme Discreta", xlab =
"Espacio Muestral",
     ylab = "Probabilidad",ylim=c(0, 1))
```

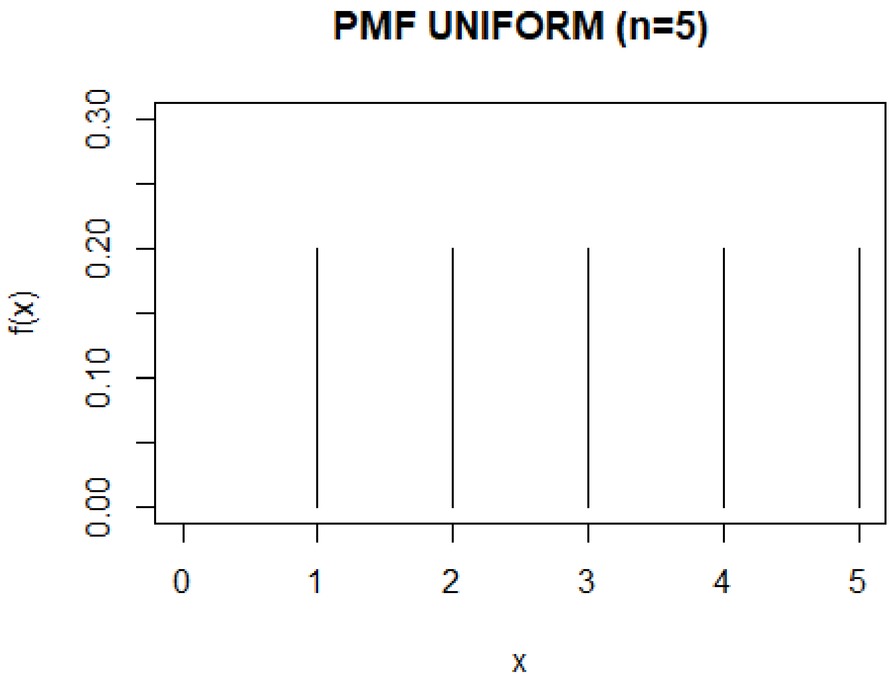



```
cdf<-0
for(i in seq(1,10,1)){#función de distribución
  cdf[i]<-punif(i,min = 0,max = 10, log = FALSE)
}
```

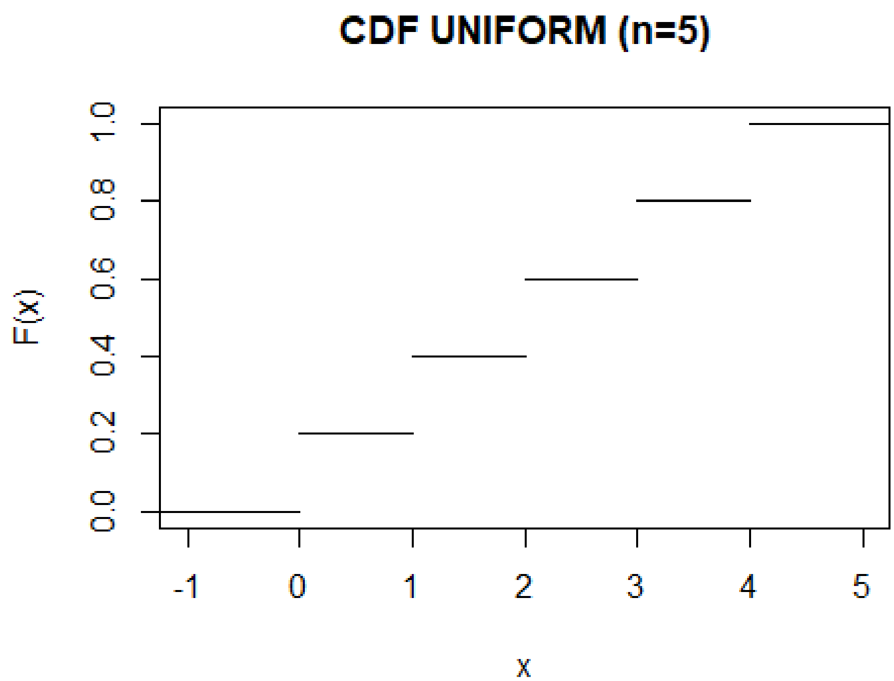
```r
SampleSpace<-1:10
plot(SampleSpace,cdf, type = "b", main="CDF Uniforme Discreta", xlab =
"Espacio Muestral",
     ylab = "Probabilidad Acumulada",ylim=c(0, 1))
```



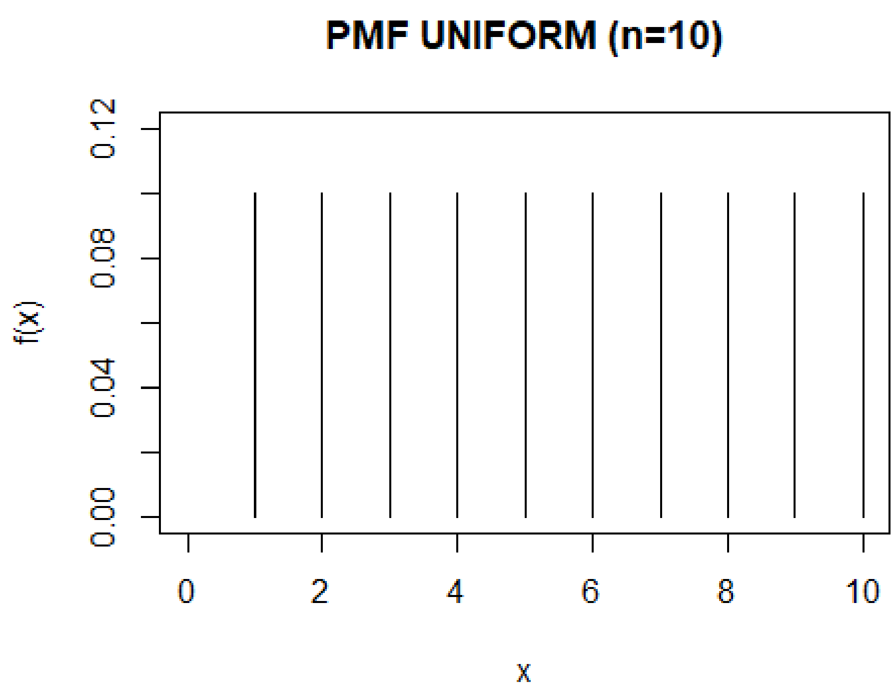

```r
# Diapositiva número 14.
plot(dunif(0:5, 0, 5), type="h", ylim=c(0,.3), xlim=c(0,5), main="PMF
UNIFORM (n=5)", ylab="f(x)", xlab="x") # n = 5 pmf
```



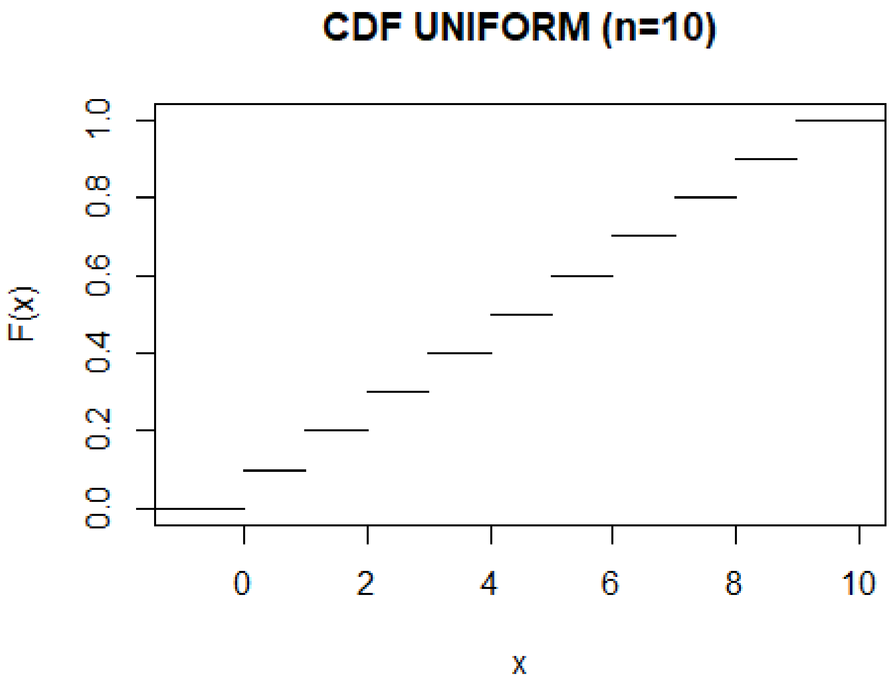

```r
  t <- c(0:4)
x <- punif(0:5, 0, 5) # CDF n=5
plot(stepfun(t, x), xlab="x", ylab="F(x)", main="CDF UNIFORM (n=5)",
     do.points = FALSE, pch = 16,verticals = FALSE)
```



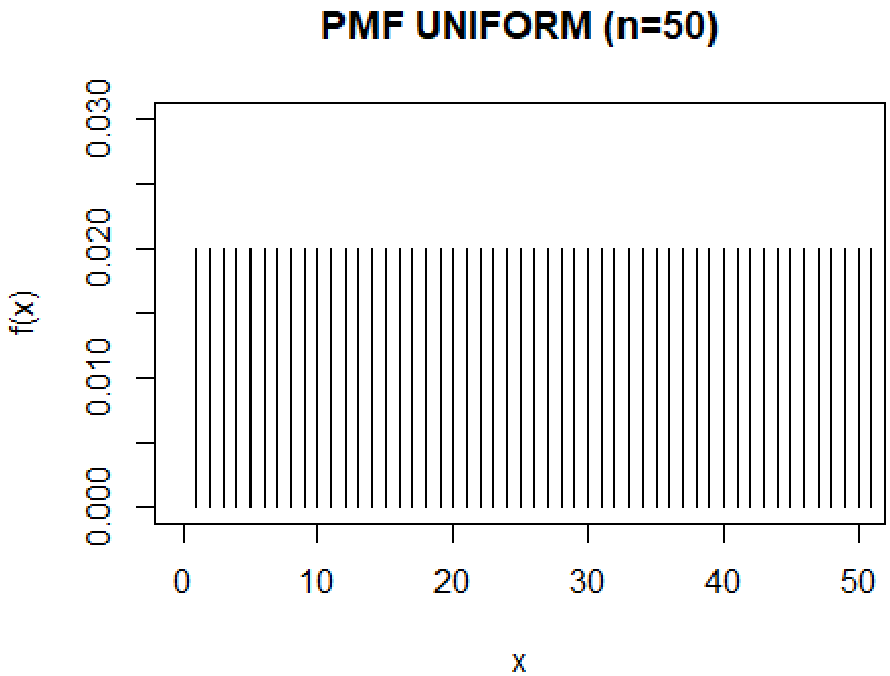

```r
# Diapositiva número 15.
plot(dunif(0:10, 0, 10), type="h", ylim=c(0,.12), xlim=c(0,10), main="PMF
UNIFORM (n=10)", ylab="f(x)", xlab="x") # n = 10 pmf
```



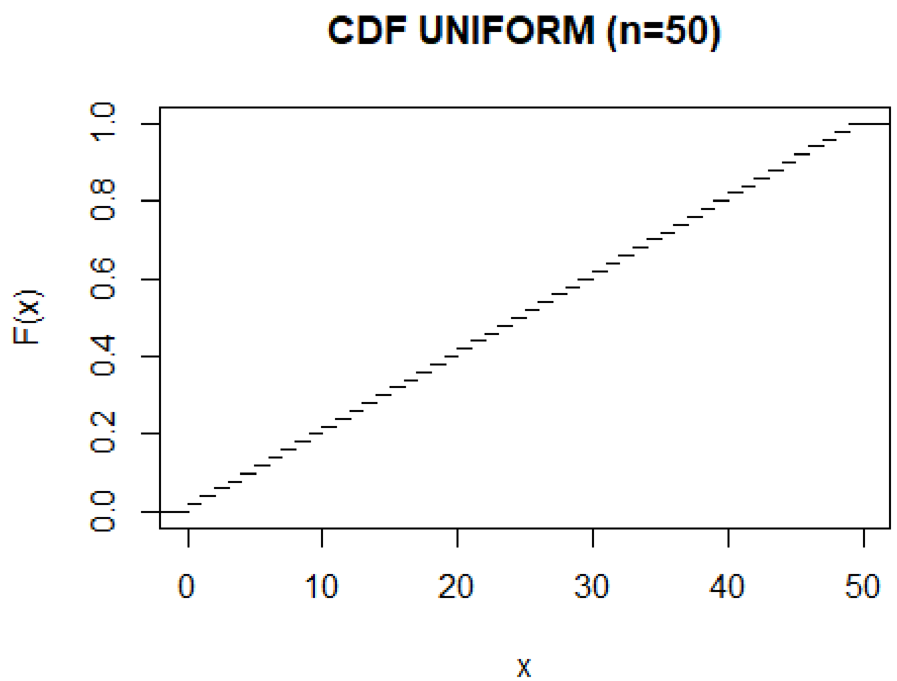
```r
t <- c(0:9)
x <- punif(0:10, 0, 10) # CDF n=10
plot(stepfun(t, x), xlab="x", ylab="F(x)", main="CDF UNIFORM (n=10)",
     do.points = FALSE, pch = 16,verticals = FALSE)
```




```r
# Diapositiva número 16.
plot(dunif(0:50, 0, 50), type="h", ylim=c(0,.030), xlim=c(0,50),
main="PMF UNIFORM (n=50)", ylab="f(x)", xlab="x") # n = 50 pmf
```




```r
t <- c(0:49)
x <- punif(0:50, 0, 50) # CDF n=50
plot(stepfun(t, x), xlab="x", ylab="F(x)", main="CDF UNIFORM (n=50)",
     do.points = FALSE, pch = 16,verticals = FALSE, xlim=c(0,50))
```



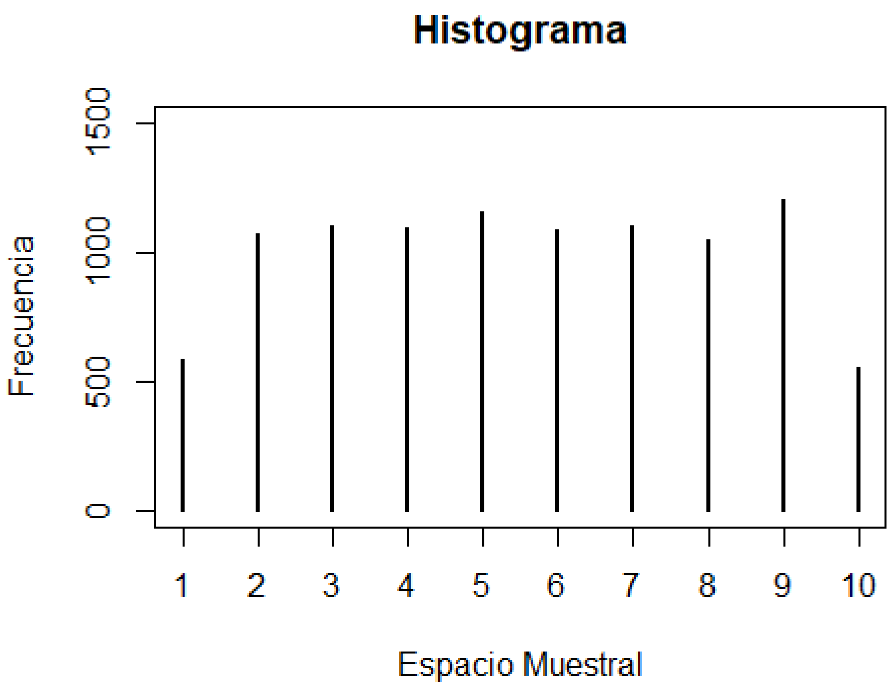
b) Lea en la documentaciónde R, o en cualquier otra fuente de informacion confiable, la explicacion de la funcion sample(x, size, replace=FALSE, prob=NULL). (No es necesario entregar algo para este ejercicio).

c) Usando la funcion sample simule una muestra de tamaño 10000 de la distribucion U(1,.....,10). Fijando la semilla en 13 (set.seed(13)), muestre los resultados de la simulacion en una tabla de frecuencia y calcule la media y la varianza. Sugerencia: Use la funcion table.

```
rm(list=ls())
set.seed(13)#fijando la semilla en 13
uniform<-sample(round(runif(10000,1,10)))
table(uniform)#frecuencia

## uniform
##    1    2    3    4    5    6    7    8    9   10
##  585 1073 1104 1093 1156 1089 1102 1043 1200  555

#La media es de:
mean(uniform)

## [1] 5.5137

#La varianza es de:
var(uniform)

## [1] 6.985311
```

d) Grafique las frecuencias de la simulacion anterior.

```
plot(table(uniform), type="h", main="Histograma", xlab = "Espacio
Muestral",
     ylab = "Frecuencia", ylim=c(0, 1500))
```



6. Para el siguiente ejercicio también es necesario R.

a) Usando la funciónn sample, simule 10 lanzamientos de una moneda equilibrada y cuente el número de aguilas que obtiene. Repita este proceso 10^6 veces y muestre sus primeros 3 resultados. Grafique las frecuencias del numero de aguilas obtenidas en los 10^6 experiementos. Tambien graque las proporciones (probabilidades) del numero de aguilas obtenidas.

b) Usando la funcion dbinom grafique la funcion de masa de una distribucion B(10; 0:5) sobre la grafica de las proporciones que hizo en el inciso anterior. Que observa?

```
rm(list=ls())
count3<-0
set.seed(10)#fijando la semilla en 10
LanzamientoMoneda3<-sample(0:1,10,.5)#guardando los lanzamientos
```

Se realizan diez lanzamientos de una moneda con probabilidad de 0.5 porceinto, en el cual, el 0 representa sello y el uno aguila.

```
LanzamientoMoneda3#resultados de los lanzamientos

##  [1] 1 0 0 1 0 0 0 0 1 0
```

El numero de aguilas que se obtiene es de:

```
count3<-(length(which(LanzamientoMoneda3==1)))
count3

## [1] 3
```

A continuación se repite el proceso 10 a la 6 veces con el comando sample y se saca su función de densidad. Una vez eso, se compara con la función de densidad que proporciona el comando “dbnom” generando una pmf de una binomial.

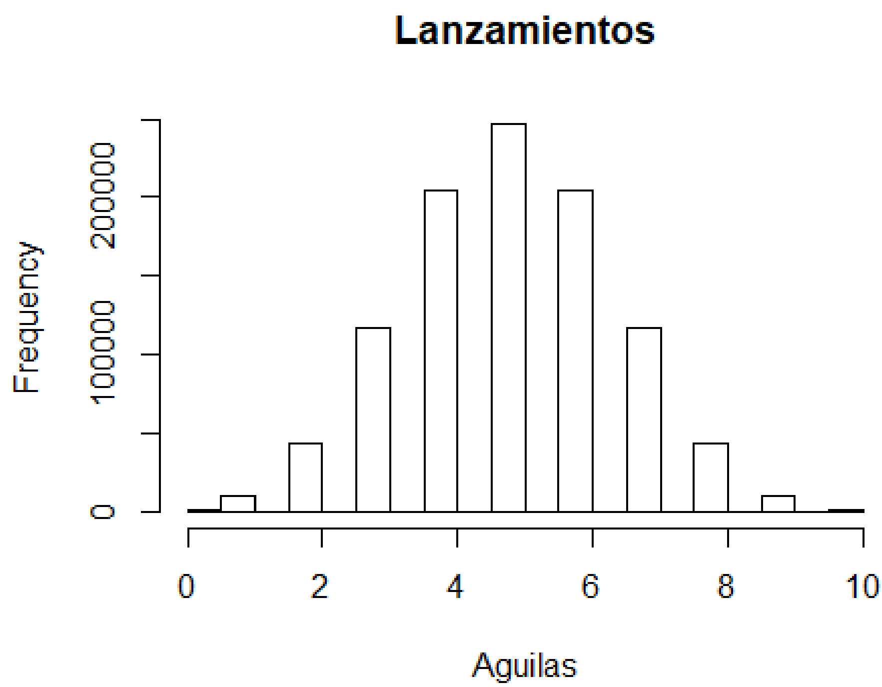
```
rm(list=ls())
count4<-rep(0,1000000)
pb<-txtProgressBar(1, 1000000,1)
for (i in seq(1,1000000,1)){
  LanzamientoMoneda4<-sample(0:1,10,.5)
  count4[i]<-(length(which(LanzamientoMoneda4==1)))
  setTxtProgressBar(pb,i)
}

##
========================================================================
==
```

Se imprimen los primeros tres resultados

```
count4[1:3]

## [1] 4 6 7
```

Se gráfica el número de aguilas que se presentaron en los lanzamientos.

```
hist(count4, xlab="Aguilas", main="Lanzamientos")
```

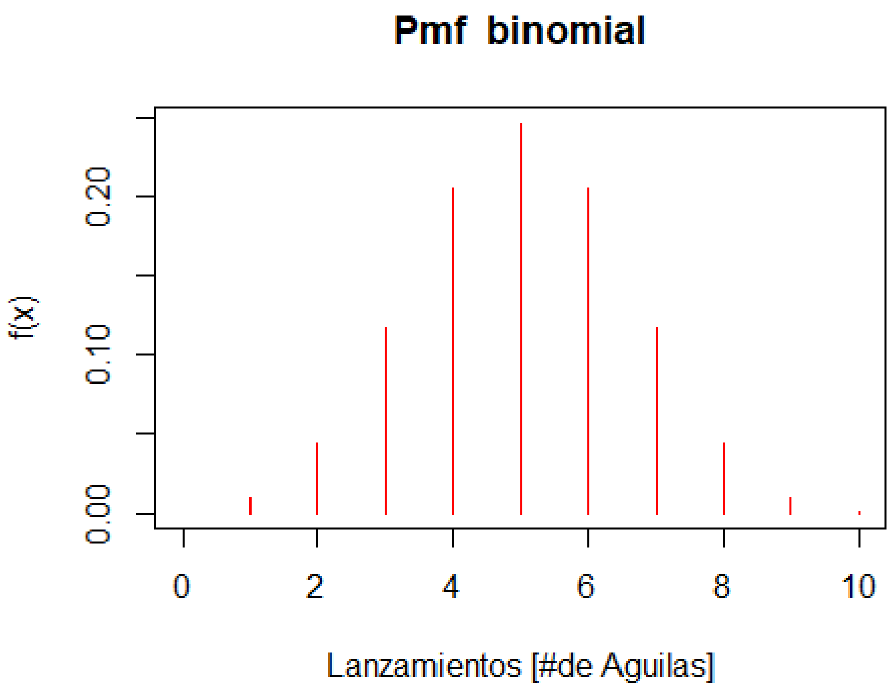


Calculamos las probabilidades y realizamos un plot a la función de densidad

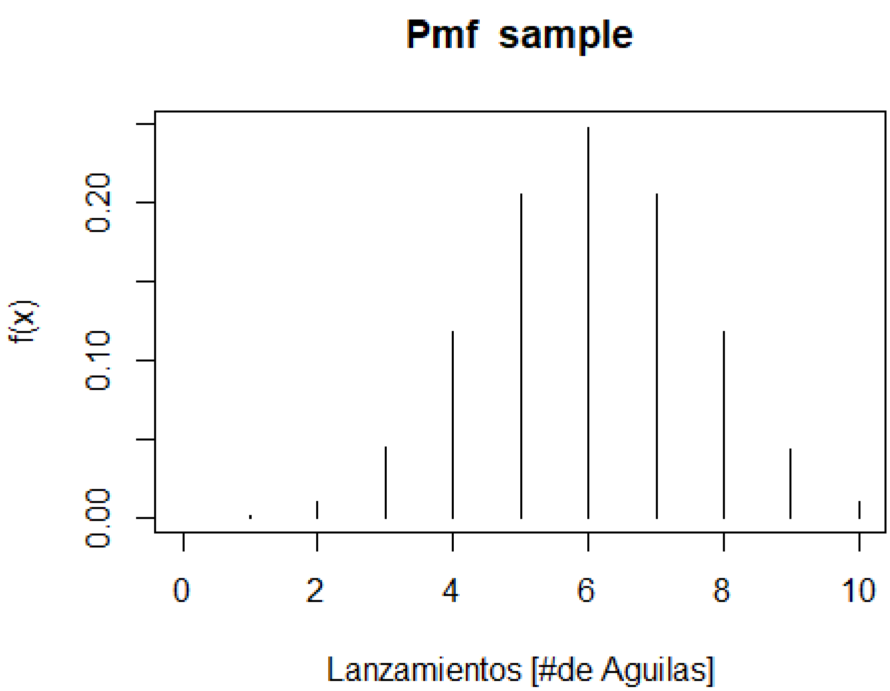
```
b<-table(count4)
c<-rep(0,11)
for (i in seq(1,length(b),1)){
  c[i]<-b[[i]]/1000000
}
```

Se gráfica la distribución de densidad (pmf) de una binomial y se empalma junto la simulación de la densidad anterior.

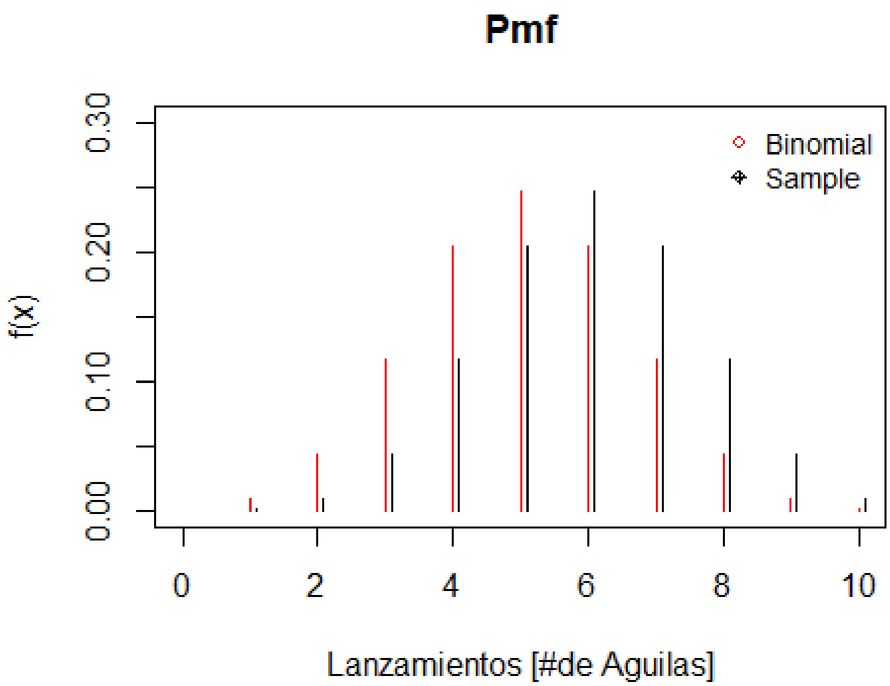
```
a<-rep(0,10)
for (i in seq(0,10,1)){
  a[i]<-dbinom(i, 10,.5)
}

plot(a, type="h", xlab="Lanzamientos [#de Aguilas]", ylab="f(x)", main =
"Pmf  binomial", xlim = c(0,10), col="red")
```




```
plot(c[1:10], type="h", xlab="Lanzamientos [#de Aguilas]", ylab="f(x)" ,
main= "Pmf  sample", xlim = c(0,10))
```



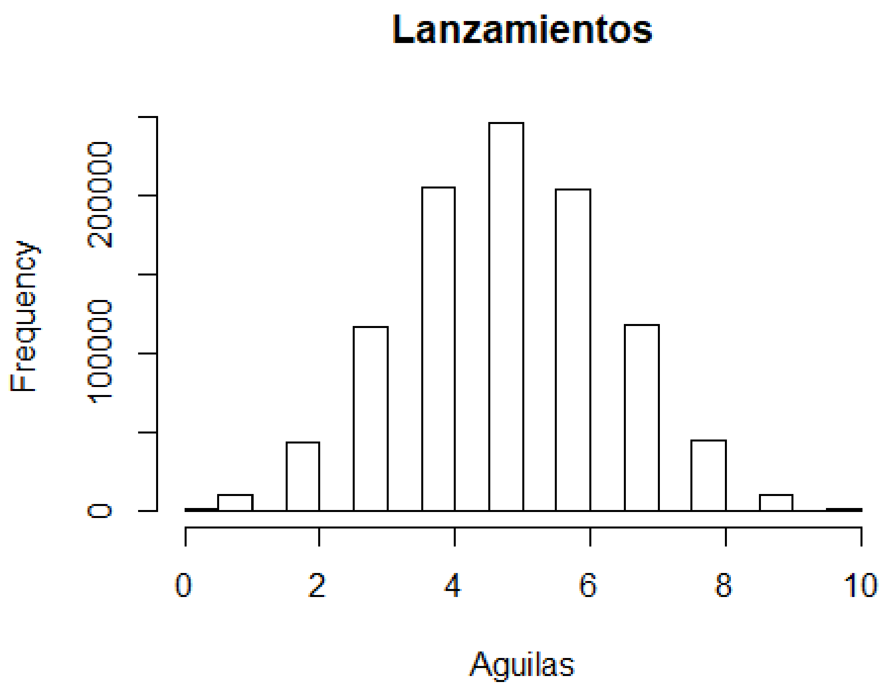
```r
#comparando las PMF
plot(a, type="h", xlab="Lanzamientos [#de Aguilas]", ylab="f(x)" , main=
"Pmf", xlim = c(0,10), ylim=c(0,.3), col="red")
lines(1:10+.09,c[1:10], type="h", ylim=c(0,.25))
legend("topright", legend=c("Binomial", "Sample"), pch=c(1,10),
       col=c("red", "black"),
       horiz=FALSE, bty='n', cex=0.9)
```



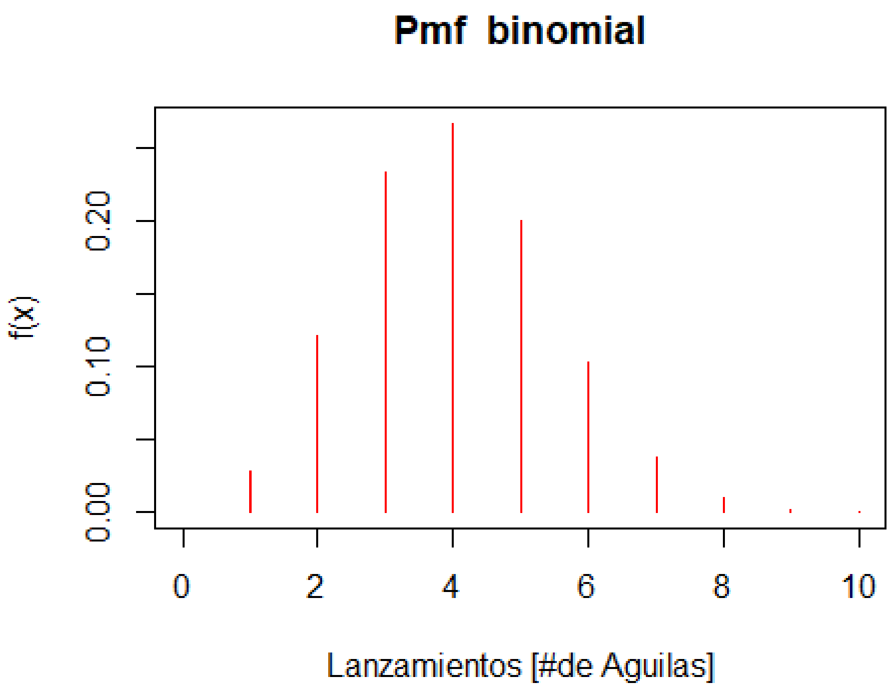
c) Repita los dos incisos anteriores para una moneda desequilibrada que tiene probabilidad p = 0:3 de obtener un aguila cuando se lanza. Que observa?

```r
rm(list=ls())
count2<-rep(0,1000000)
pb<-txtProgressBar(1, 1000000,1)
for (i in seq(1,1000000,1)){
  LanzamientoMoneda2<-sample(0:1,10,.3)
  count2[i]<-(length(which(LanzamientoMoneda2==1)))
  setTxtProgressBar(pb,i)
}

##
=========================================================================
==

count2[1:3]

## [1] 4 6 4
```

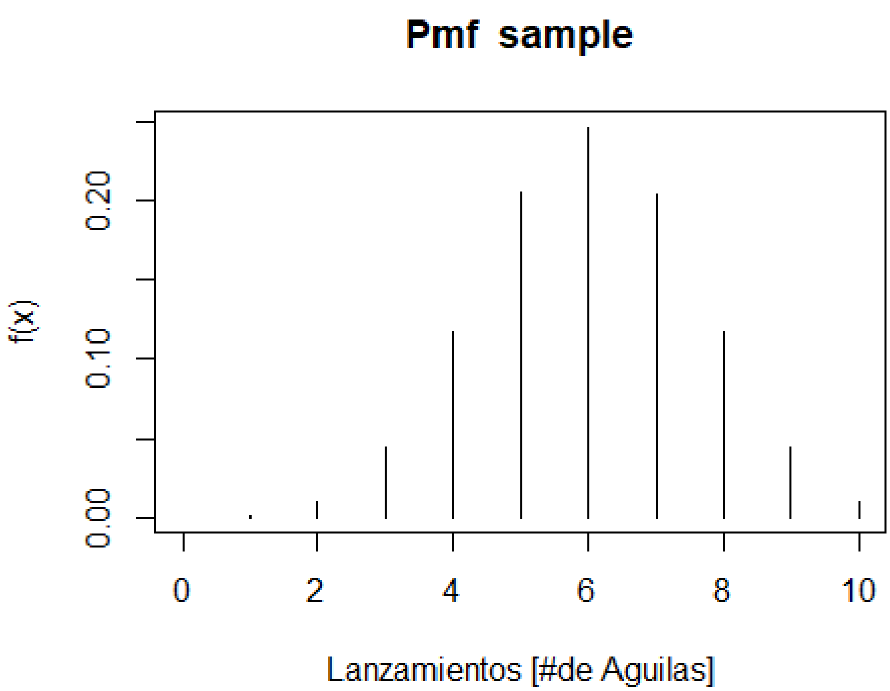
```r
hist(count2, xlab="Aguilas", main="Lanzamientos")
```



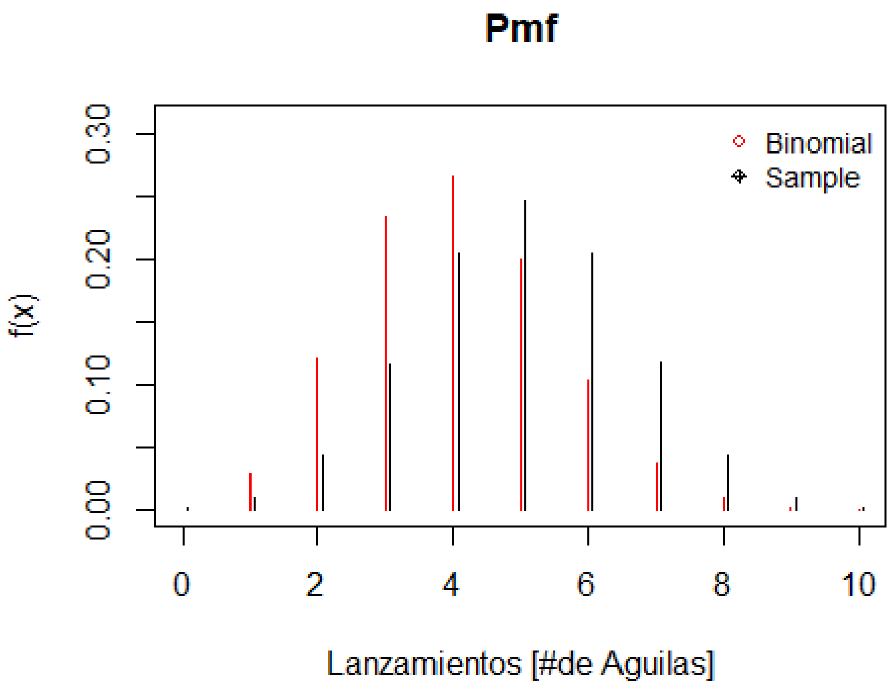

```r
b<-table(count2)
c<-rep(0,11)
for (i in seq(1,length(b),1)){
  c[i]<-b[[i]]/1000000
}

a<-rep(0,10)

a<-dbinom(0:10, 10,.3)


plot(a, type="h", xlab="Lanzamientos [#de Aguilas]", ylab="f(x)", main =
"Pmf  binomial", xlim = c(0,10), col="red")
```




```r
plot(c, type="h", xlab="Lanzamientos [#de Aguilas]", ylab="f(x)" , main=
"Pmf  sample", xlim = c(0,10))
```



```
plot(a, type="h", xlab="Lanzamientos [#de Aguilas]", ylab="f(x)" , main=
"Pmf", xlim = c(0,10), ylim=c(0,.31), col="red")
lines(0:10+.07, c, type="h", xlim = c(1,10), ylim=c(0,.31))
legend("topright", legend=c("Binomial", "Sample"), pch=c(1,10),
       col=c("red", "black"),
       horiz=FALSE, bty='n', cex=0.9)
```



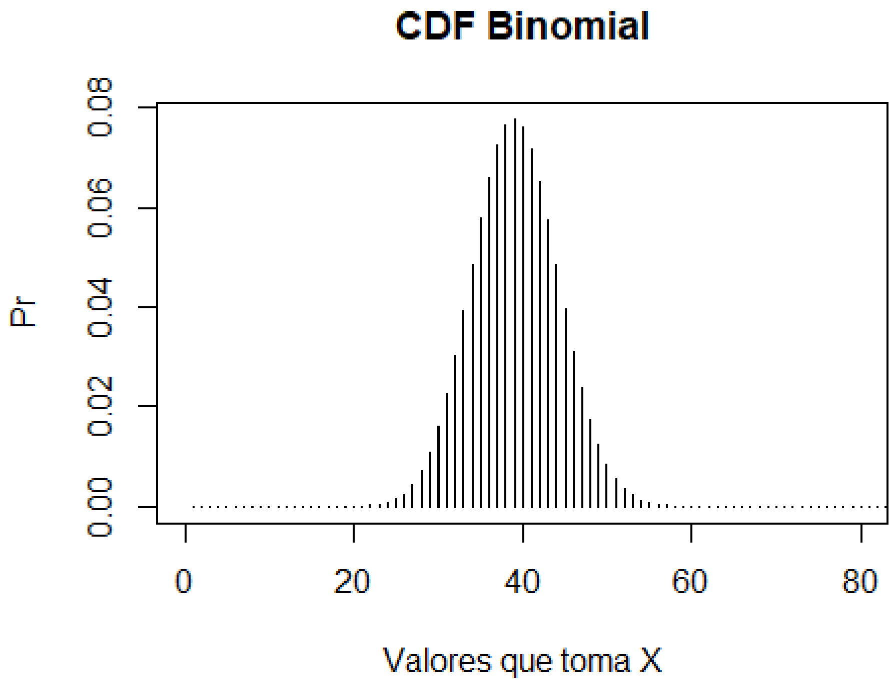
Con probabilidad de .3 se observa que con la binomial el valor medio se sesga a un valor medio cercano a cuatro. Caso contrario con la simulación sample, cuyo valor medio se aproxima al seis. De esta manera, con la función sample a una probbabilidad del .3, no se aproxima bien a la binomial.

7. Suponga que X B(123; 0.31). Resuelva lo siguiente:

a) Escriba un programa en R que calcule las siguientes probabilidades directamente de la funcion de masa: i) P(X = 0), P(X = 123) y P(X = 62); ii) P(0 <= X <= 10), P(0 < X < =10) y P(0 < =X < 10); iii) P(X > 11) y P(X <= 10).

Generando la Pdf una binomial

```
rm(list=ls())
PmfBinomial<-dbinom(0:123, 123, 0.31)
plot(PmfBinomial, type = "h", ylab="Pr", xlab="Valores que toma X", main
= "CDF Binomial", xlim=c(0,80))
```



Probabilidades a calcular

```
dbinom(0, 123, 0.31)#P(X=0)

## [1] 1.508128e-20

PmfBinomial[123]#P(X=123)

## [1] 7.496885e-61

PmfBinomial[62+1]#P(X=62)

## [1] 3.275387e-06

sum(PmfBinomial[0:10+1])#P(0<-X<-10)

## [1] 9.364108e-10

sum(PmfBinomial[1:10+1])#P(0<X<-10)

## [1] 9.364108e-10

sum(PmfBinomial[0:9+1])#P(0<-X<10)

## [1] 1.783427e-10

1-(sum (PmfBinomial[0:10+1]))#P(X>11)

## [1] 1

sum(PmfBinomial[0:10+1])#P(X<-10)
```

```
## [1] 9.364108e-10
```

b) Calcule las probabilidades del inciso anterior usando la funciones pbinom y dbinom.

```
dbinom(0, 123, 0.31)

## [1] 1.508128e-20

dbinom(123, 123, 0.31)

## [1] 2.738346e-63

dbinom(62, 123, 0.31)

## [1] 3.275387e-06

pbinom(10, 123, 0.31)

## [1] 9.364108e-10

pbinom(10, 123, 0.31)-pbinom(0, 123, 0.31)

## [1] 9.364108e-10

pbinom(9, 123, 0.31)

## [1] 1.783427e-10

1-pbinom(10, 123, 0.31)

## [1] 1

pbinom(10, 123, 0.31)

## [1] 9.364108e-10
```

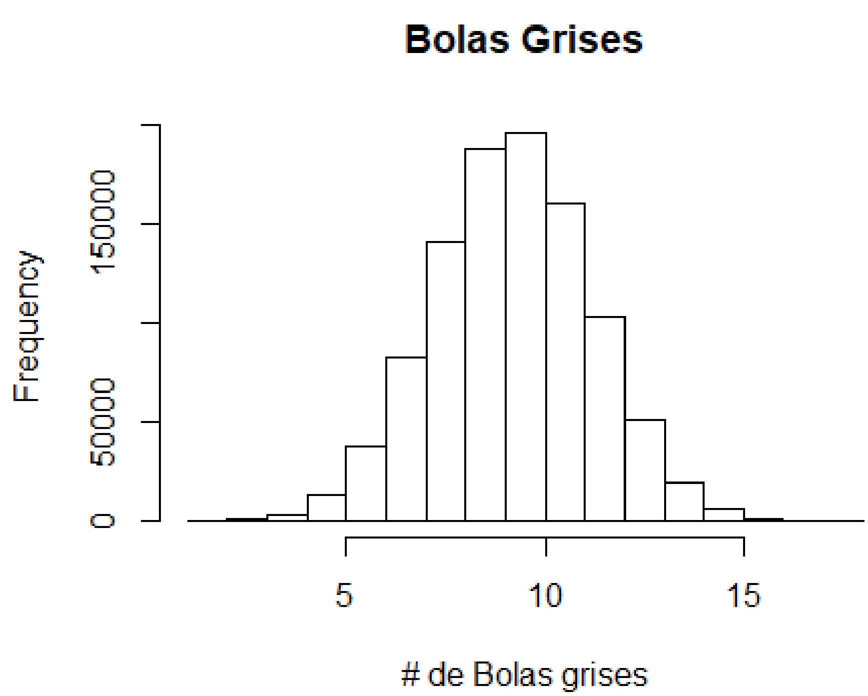
c) Escriba un programa en R que calcule los cuantiles de 0.25, 0.5 y 0.75. >Existe alguna funcion en R que calcule cuantiles?

```
qbinom(0.25, 123, 0.31)

## [1] 35

qbinom(0.5, 123, 0.31)

## [1] 38

qbinom(0.75, 123, 0.31)

## [1] 42
```

8. Una urna contiene 46 bolas grises y 49 bolas blancas. Usando la funcion sample en R, simule la extraccion sin reemplazamiento de 20 de estas bolas y cuente el numero de bolas grises que obtuvo. Repita este proceso 10^6 veces, muestre sus primeros 3 resultados y grafique las frecuencias de bolas grises obtenidas en

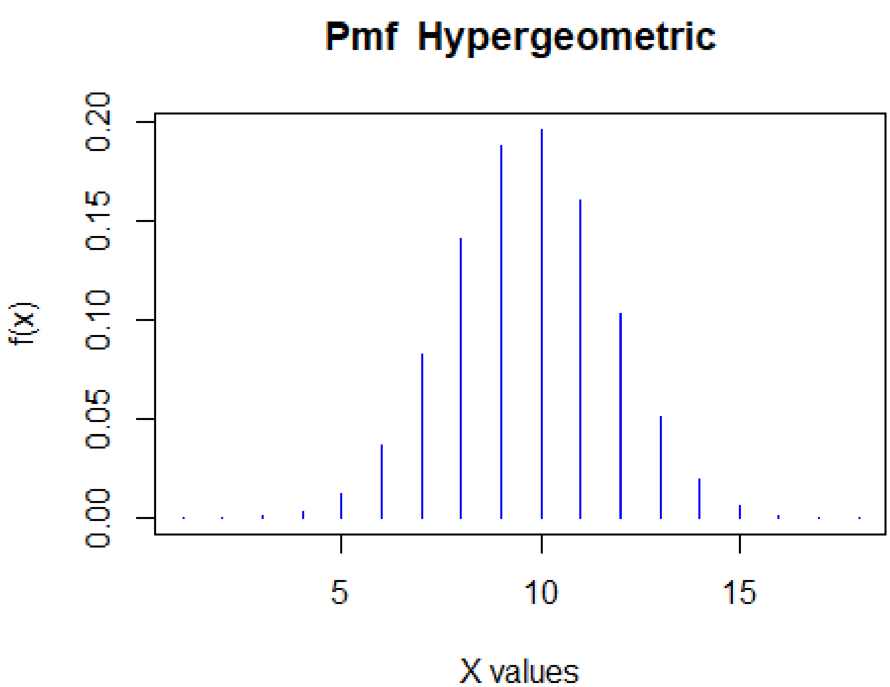
cada experimento. Cual es la probabilidad de que al extraer 20 bolas de la urna 5 de ellas sean grises? Tambien grafique la proporcion de bolas grises obtenidas en los experiementos anteriores y sobre esta gura anada la correspondiente funcion de masa de la distristibucion Hipergeometrica asociada al experimento total.

```
rm(list=ls())
count<-rep(0,1000000)
pb<-txtProgressBar(1, 1000000,1) #medidor de tiempo para simulación
for (i in seq(1,1000000,1)){
  urna<-c(rep("gris",46), rep("blanca",49)) #generando urna
  muestra<-sample(urna,20)
  count[i]<-(length(which(muestra=="gris")))
  setTxtProgressBar(pb,i)
}

##
==========================================================================
==

count[1:3] #mostrando primeros tres resultados

## [1] 11  8 11

hist(count,xlab="# de Bolas grises", main= "Bolas Grises") #frecuencia
del numero de bolas grises
```

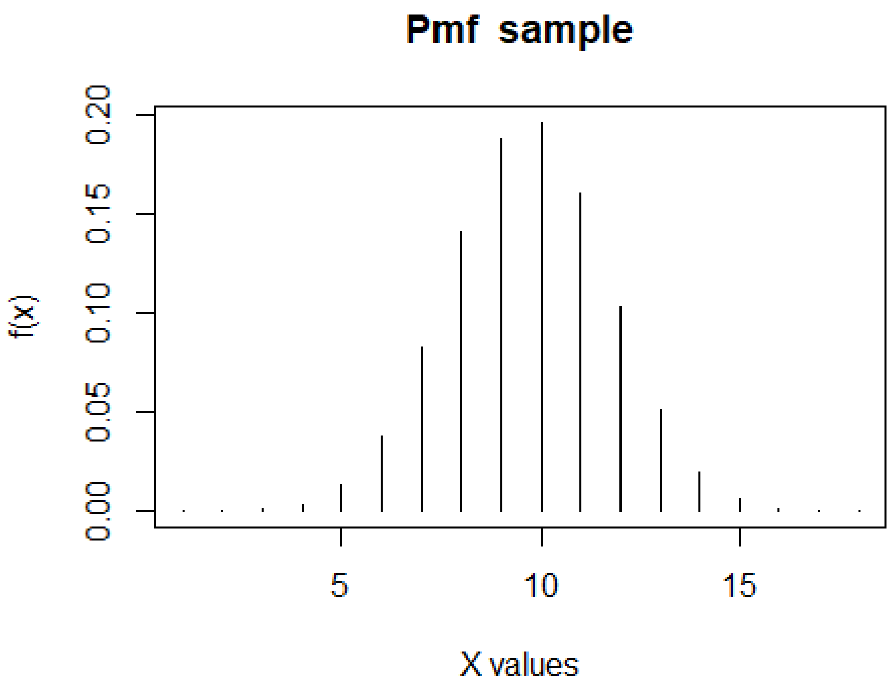



```
b<-table(count)
c<-rep(0,10)
```

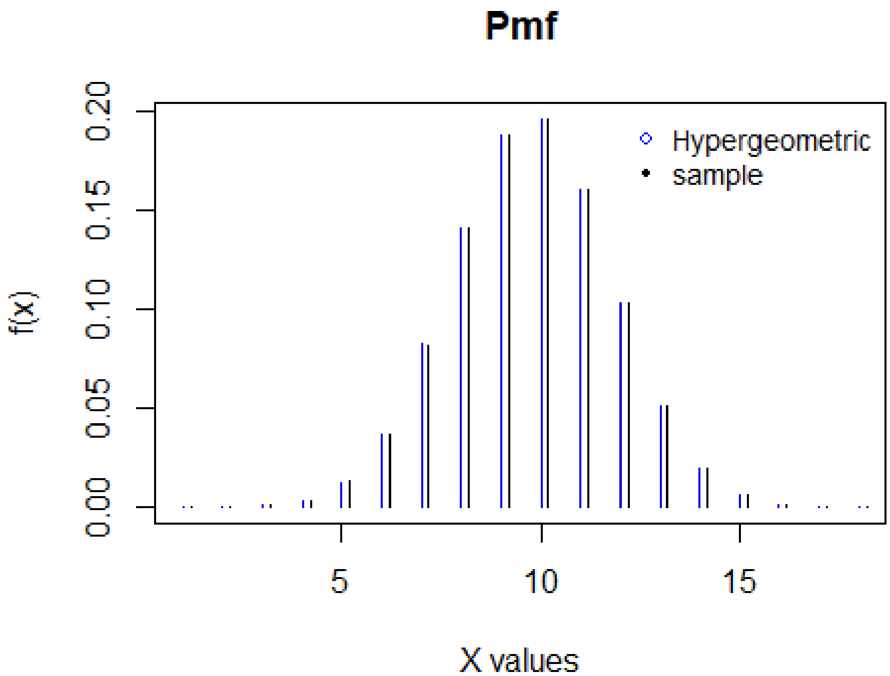
```r
#Probabilidades
for (i in seq(1,length(b),1)){
  c[i]<-b[[i]]/1000000
}
c[5]#probabilidad de sacar 5 bolas grises. Utilizando sample

## [1] 0.012659

pr<-(choose(46, 5)*choose(49, 20-5))/choose(46+49, 20) #probabilidad de
sacar 5 bolas grises de una muestra de 20
pr

## [1] 0.01261935

dhyper(5, 46, 49, 20) #probabilidad de sacar 5 bolas grises de una
muestra de 20

## [1] 0.01261935

a<-rep(0,length(b))
#Pdf utilizando hipergeometrica
for (i in seq(0,length(b),1)){
  a[i]<-dhyper(i, 46, 49, 20)
}
#Comparando pdf de las simulaciones con saple y función hipergeometrica
plot(a, type="h", xlab="X values", ylab="f(x)", main = "Pmf
Hypergeometric", col="blue")
```



```
plot(c, type="h", xlab="X values", ylab="f(x)" , main= "Pmf  sample")
```




```
plot(a, type="h", xlab="X values", ylab="f(x)" , main= "Pmf", col="blue")
lines(1:length(b)+.2,c, type="h")
  legend("topright", legend=c("Hypergeometric", "sample"), pch=c(1,20),
       col=c("blue", "black"),
       horiz=FALSE, bty='n', cex=0.9)
```



Se observa que al simular la pmf con la función sample se aproxima a la pmf con generada con una la hipergeometrica.