

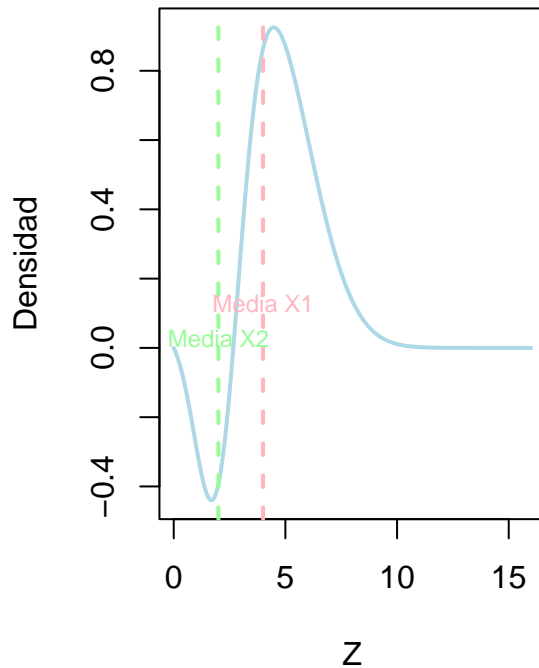
# Ejercicio 6

2024-02-09

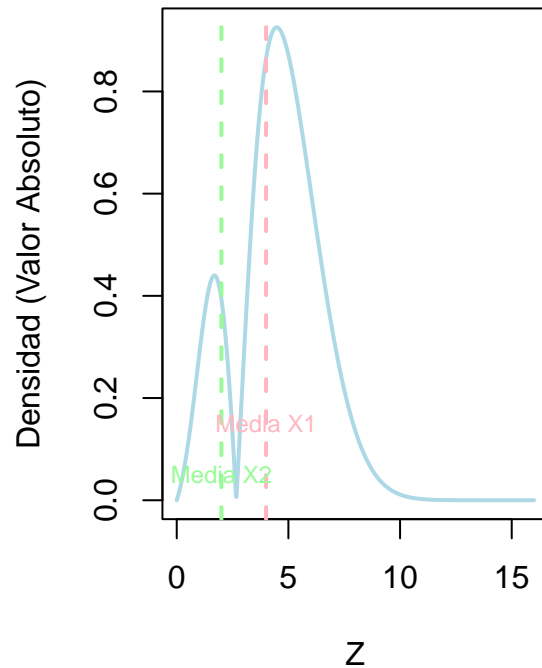
## Ejercicio 6

```
fnormal <- function(x,mu1,mu2,sigma1, sigma2) {  
  fx <- exp(-((x-mu1)^2/(2*(sigma1)))) - exp(-((x-mu2)^2/(2*(sigma2))))  
  return(fx)  
}  
  
mu1 <- 4  
mu2 <- 2  
sigma1 <- 4  
sigma2 <- 1  
  
fZ <- function(x){return(fnormal(x,mu1,mu2,sigma1,sigma2))}  
  
# Valores para el rango de la gráfica  
x_values <- seq(0, 16, length.out = 1000)  
  
par(mfrow = c(1, 2))  
  
# Gráfico de la distribución de Z y las medias de X1 y X2  
plot(x_values, fZ(x_values), type = "l", col = "#ADD8E6", lwd = 2,  
      xlab = "Z", ylab = "Densidad", main = "Distribución de Z = X1 - X2")  
  
# Líneas verticales para las medias de X1 y X2  
abline(v = c(mu1, mu2), col = c("#FFB6C1", "#98FB98"), lty = c(2, 2), lwd = 2)  
  
# Etiquetas para las medias  
text(mu1, 0.20, "Media X1", pos = 1, col = "#FFB6C1", cex = 0.75)  
text(mu2, 0.10, "Media X2", pos = 1, col = "#98FB98", cex = 0.75)  
  
# Gráfico de la distribución en valor absoluto de Z y las medias de X1 y X2  
plot(x_values, abs(fZ(x_values)), type = "l", col = "#ADD8E6", lwd = 2, xlab = "Z",  
      ylab = "Densidad (Valor Absoluto)", main = "Distribución de Z = X1 - X2")  
  
abline(v = c(mu1, mu2), col = c("#FFB6C1", "#98FB98"), lty = c(2, 2), lwd = 2)  
  
text(mu1, 0.20, "Media X1", pos = 1, col = "#FFB6C1", cex = 0.75)  
text(mu2, 0.10, "Media X2", pos = 1, col = "#98FB98", cex = 0.75)
```

Distribución de  $Z = X1 - X2$



Distribución de  $Z = X1 - X2$



```
fpK <- function(x,y){
  pK <- dcauchy(y,location = x) #x es elcentro del pico de la distribución.
  return(pK)
}

N <- 10^5 #Número de Iteraciones
L <- 1000 #periodo quemado (burnin)
MCMC <- matrix(data=0,nrow=N,ncol=12)
colnames(MCMC) <- c("x","y","PIx","PIy","Kxy","Kyx","Rxy","Ryx","Mxy","Myx","Fxy",
  "Salto")

#1.Iniciar con un valor arbitrario de x del dominio de distribución
x <- runif(1,-50,50)
for(i in 1:N){
  #2.Generamos la propuesta con una distribucion arbitraria
  y <- rcauchy(1,location=x) #Valor aleatorio según X

  #3.Tasa de Aceptación
  PIx <- fZ(x)
  PIy <- fZ(y)
  Kxy <- fpK(x,y)
  Kyx <- fpK(y,x)
  Rxy <- (PIy*Kyx) / (PIx*Kxy)
  Ryx <- (PIx*Kxy) / (PIy*Kyx)

  #Matriz estocástica de los estados de la distribución estacionaria
  if(x!=y){
```

```

Mxy <- Kxy*min(1,Rxy)
Myx <- Kyx*min(1,Ryx)
} else {
  Mxy <- -1
  Myx <- -1
}

#4.Criterio de Aceptacion o Rechazo
#Probabilidad de aceptación, runif(1)
Fxy <- runif(1)
MCMC[i,] <- c(x,y,PIx,PIy,Kxy,Kyx,Rxy,Ryx,Mxy,Myx,Fxy,0)

if(Fxy < Rxy) {
  x <- y
  lsalto <- 1
} else {
  lsalto <- 0
}

MCMC[i,12] <- lsalto
}

mcmc <- MCMC[(L+1):N,"x"]

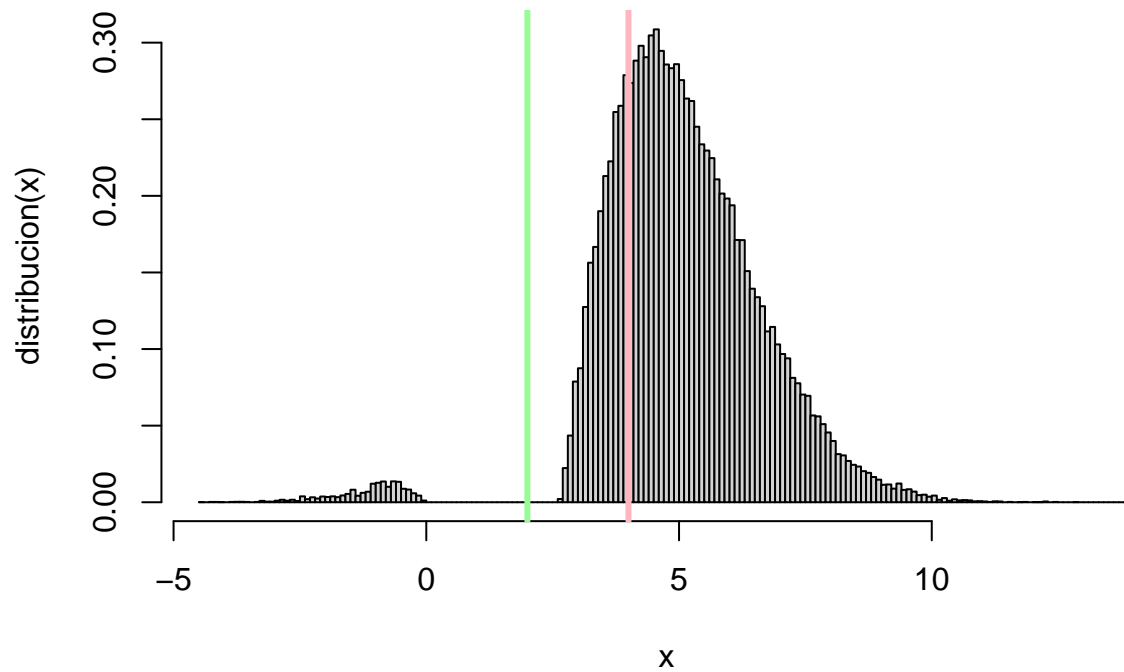
```

```

hist(mcmc, freq=FALSE, main="Distribución de muestra MCMC", xlab="x",
     ylab="distribucion(x)", breaks=200)
abline(v=mu1,col="#FFB6C1",lwd=3)
abline(v=mu2,col="#98FB98",lwd=3)

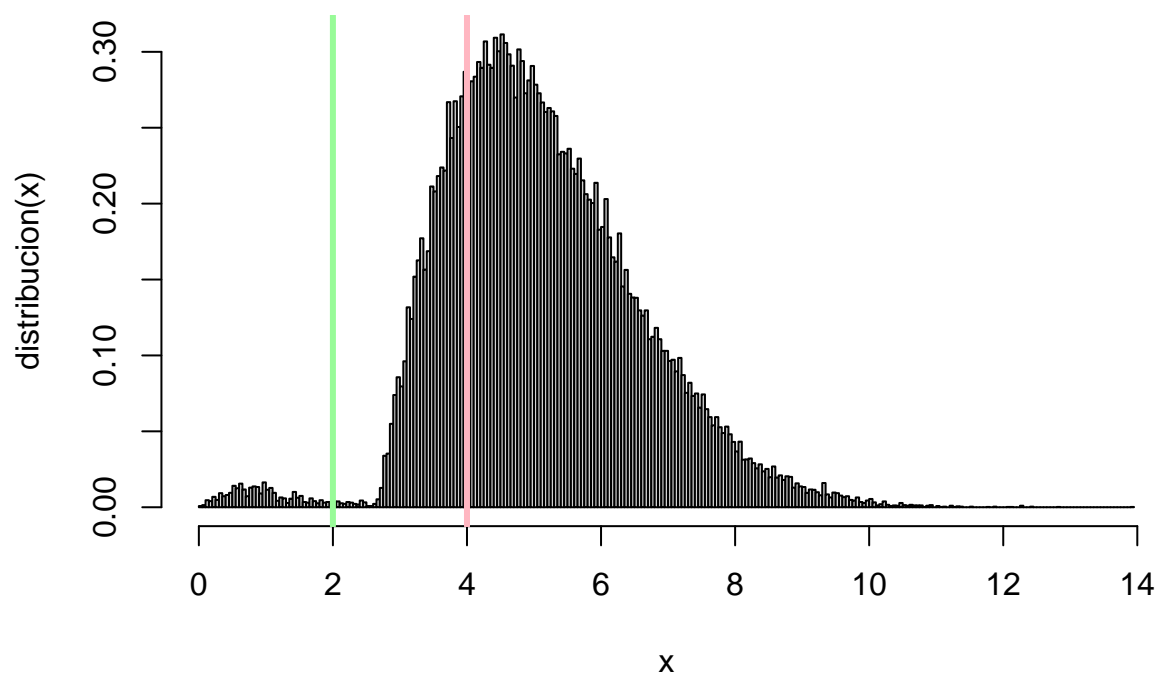
```

## Distribución de muestra MCMC



```
hist(abs(mcmc), freq = FALSE,  
     main = "Distribución de muestra MCMC (Valor Absoluto)",  
     xlab = "x", ylab = "distribucion(x)", breaks = 200)  
abline(v=mu1,col="#FFB6C1",lwd=3)  
abline(v=mu2,col="#98FB98",lwd=3)
```

## Distribución de muestra MCMC (Valor Absoluto)



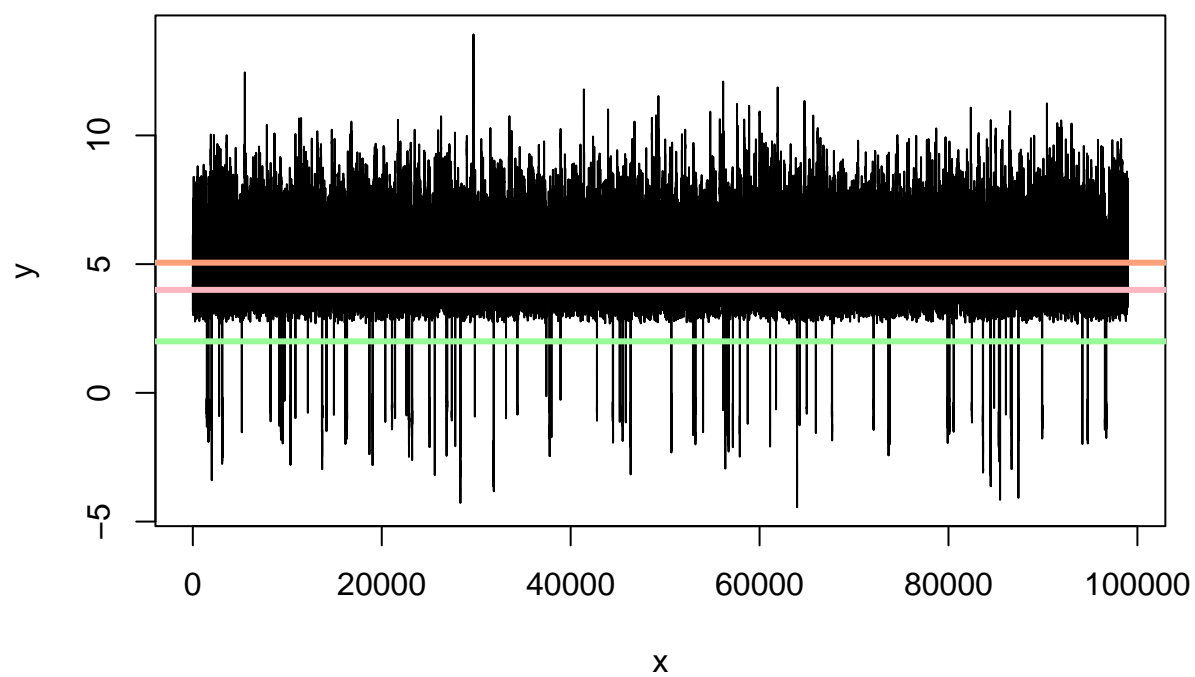
```
media <- mean(mcmc)
media
```

```
## [1] 5.055247
```

```
options(scipen = 999)

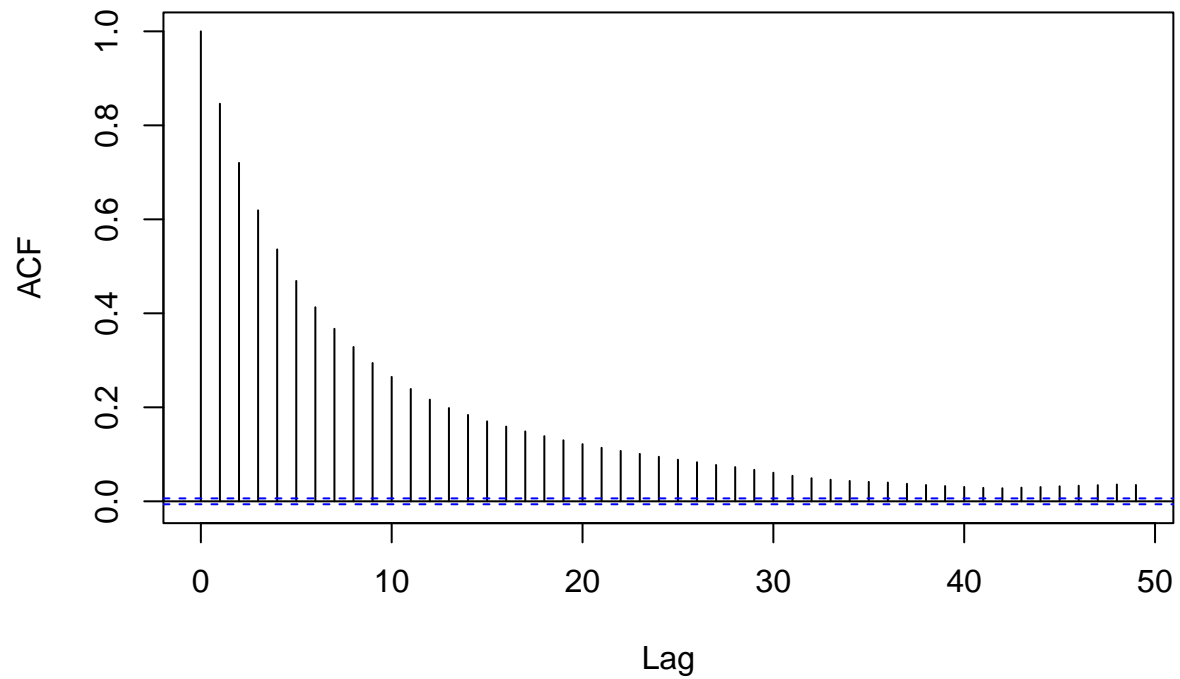
plot(mcmc,type="l",xlab="x",ylab="y",main="Traceplot de muestra MCMC")
abline(h=mu1,col="#FFB6C1",lwd=3)
abline(h=mu2,col="#98FB98",lwd=3)
abline(h=media,col="#FFA07A",lwd=3)
```

## Traceplot de muestra MCMC



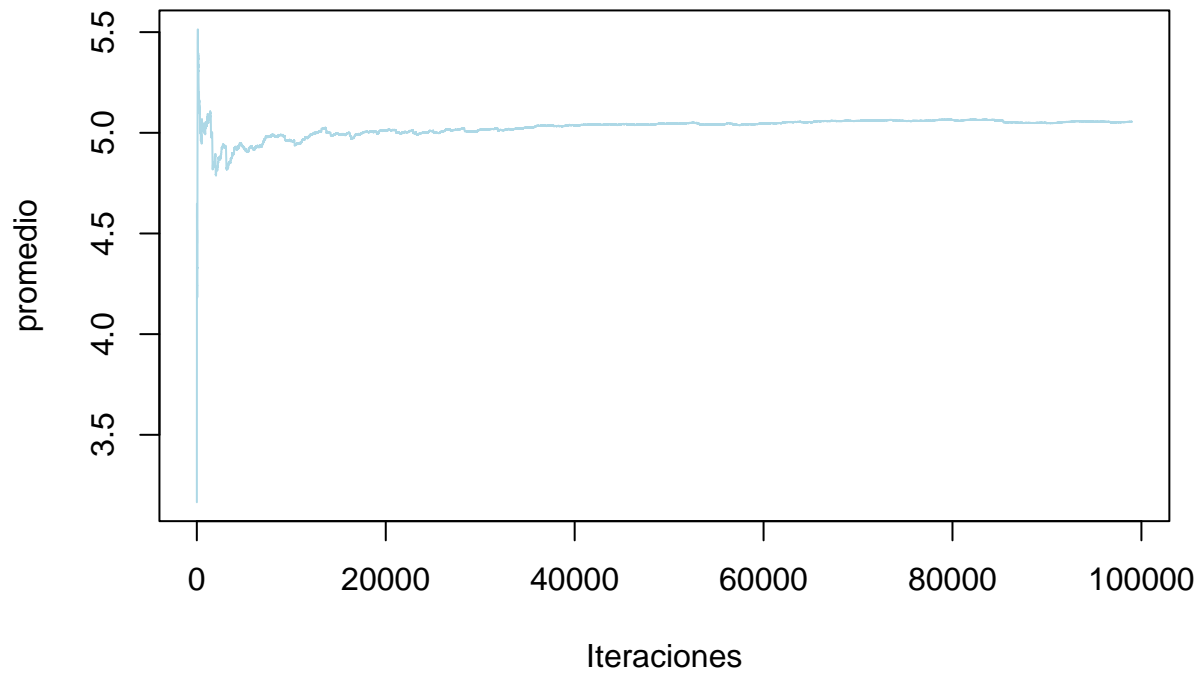
```
acf(mcmc,main="Autocorrelación de muestra MCMC")
```

## Autocorrelación de muestra MCMC



```
m <- N-L
acumulado <- cumsum(mcmc)/(1:m)
plot(1:m,acumulado,col="#ADD8E6",type="l",ylab="promedio",xlab="Iteraciones",
     main="Convergencia de la media de la muestra MCMC")
```

## Convergencia de la media de la muestra MCMC



```
cat("Tasa de aceptación:", mean(MCMC[, "Salto"]), "\n")
```

```
## Tasa de aceptación: 0.60021
```