

Práctica Evaluación 21-22

Diego Fernández

31/12/2021

Tarea 1 : lectura de datos.

Enunciado:

Cada un@ de vosotr@s recibiréis un archivo de datos personalizado. Fijaros bien en el formato: tabla con cabecera, de la que debéis extraer los datos en forma de vector.

La definición del “working directory” debéis hacerla en la línea de comandos, o desde un archivo script(.R) sin que quede rastro en este documento.

```
datos <- read.table("dfernandez118")
datos <- datos$x
```

Tarea 2: estimación de media y varianza

Seleccionar aleatoriamente una muestra de tamaño $n = 3$ y calcula su media, \bar{x} , y su cuasivarianza, s_c^2 , como estimadores de la media poblacional μ y la varianza poblacional σ^2 .

```
n <- 3

muestra <- sample(datos, n)

#media
media <- mean(muestra)
media
```

```
## [1] 62.42109
```

```
#cuasivarianza
cuasivarianza <- var(muestra)
cuasivarianza
```

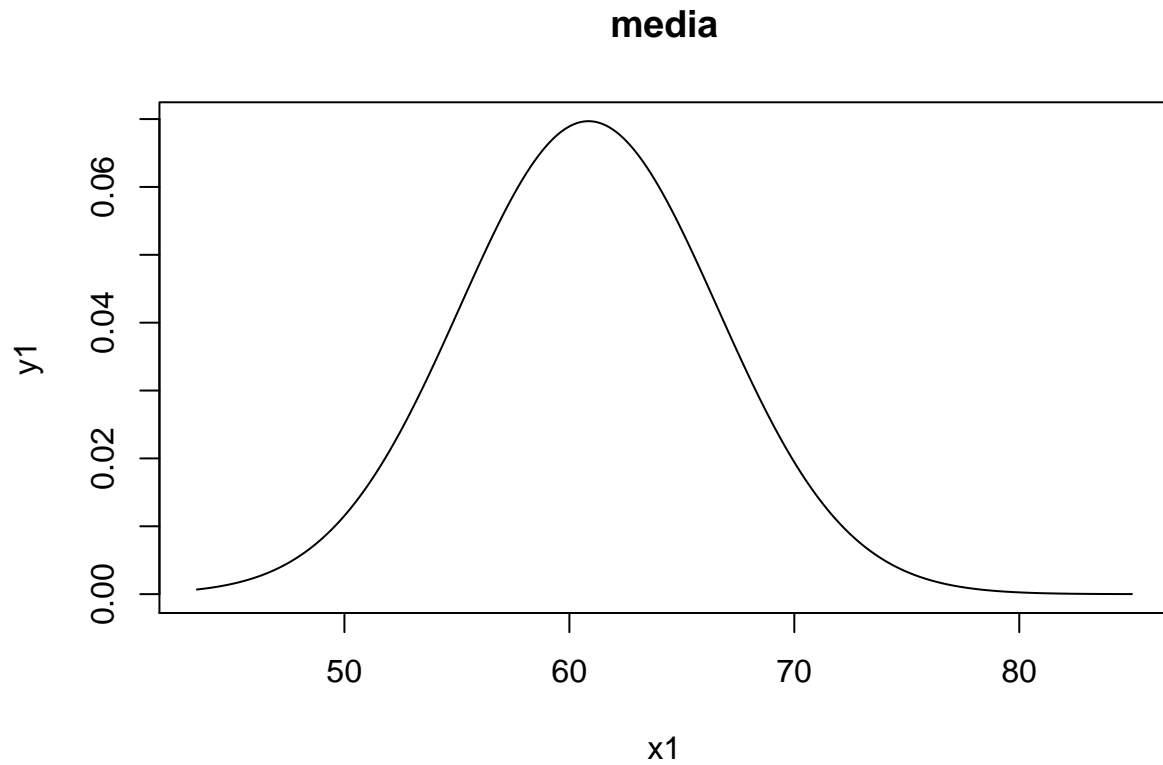
```
## [1] 41.97005
```

Representar las curvas correspondientes a las distribuciones de la media y la varianza.

```
#Distribución de la media
```

```
x1 <- seq(min(datos), max(datos), 0.01)
y1 <- dnorm(x1, mean(datos), sd(datos))

plot(x1,y1, type = "l", main = "media")
```

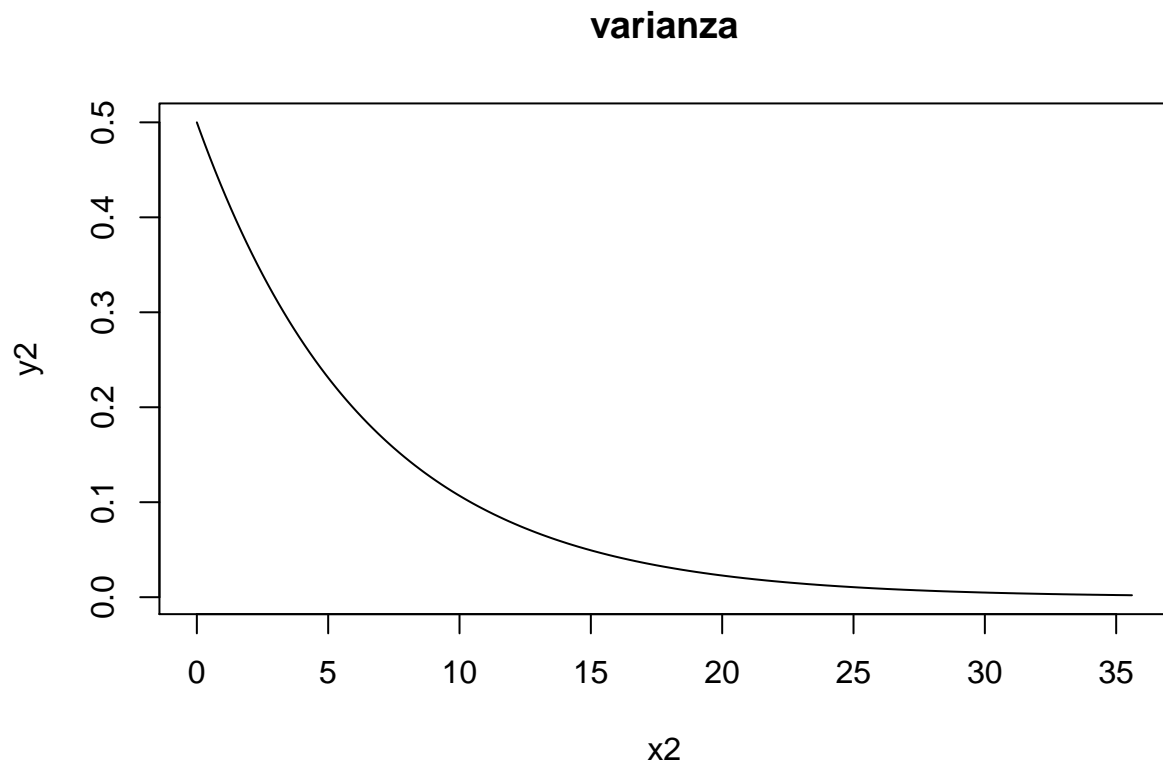


#Distribución de la varianza

```
libertad <- n-1
varianza <- sqrt(cuasivarianza)

x2 <- seq(0,5.5*varianza,0.1)
tipificar <- (n-1)*x2/varianza      #tipificar x
y2 <- dchisq(tipificar,libertad)

plot(x2,y2,type = "l", main = "varianza")
```



Tarea 3: intervalos de confianza.

A partir de las estimaciones anteriores, calcular intervalos de confianza para la media poblacional μ y la varianza poblacional σ^2 , usando distintos niveles de confianza $\alpha \in \{0.05, 0.025, 0.01\}$

```
lalpha <- list(0.05,0.025,0.01)

for (alpha in lalpha){
  calcular_intervalo <- function(n,media,varianza,alpha) {
    z1 <- qnorm((1-alpha/2), media, sqrt(varianza), lower.tail = FALSE)
    delta1 <- z1*sqrt(varianza)/sqrt(n)
    a1 <- media + delta1
    b1 <- media - delta1
    return(c(b1,a1))
  }
  calcular_intervalo(n,media,varianza,alpha)
}
```

Representar estos tres intervalos en las gráficas anteriores.

```
l <- list(3)
lalpha <- list(0.05,0.025,0.01)
```

```

for (n in 1){
  for (alpha in lalpha){
    z1 <- qnorm((1-alpha/2), media, sqrt(varianza), lower.tail = FALSE)

    dt <- sqrt(varianza)
    error1 <- dt/sqrt(n)
    errorest1 <- z1*error1

    liminf1 <- media-errorest1
    limsup1 <- media+errorest1

    x <- seq(min(datos),max(datos), 0.01)
    y <- dnorm(x,mean(datos),sd(datos))
    plot(x,y, type = "l", main = c("media, cuando alpha es igual a",alpha))

    segments(x0=liminf1,x1=liminf1,y0=0,y1=dnorm(liminf1,mean(datos),sd(datos)),col = 2)

    segments(x0=limsup1,x1=limsup1,y0=0,y1=dnorm(limsup1,mean(datos),sd(datos)),col = 2)

    lines(x,y,col = 5)

    ###varianza

    alpha <- 0.05
    z1 <- qnorm((1-alpha/2), media, sqrt(varianza), lower.tail = FALSE)

    dt <- sqrt(varianza)
    error1 <- dt/sqrt(n)
    errorest1 <- z1*error1

    liminf1 <- media-errorest1
    limsup1 <- media+errorest1

    libertad <- n-1
    varianza <- sqrt(cuasivarianza)

    x2 <- seq(0,5.5*varianza,0.1)
    tipificar <- (n-1)*x2/varianza #tipificar x
    y2 <- dchisq(tipificar,libertad)

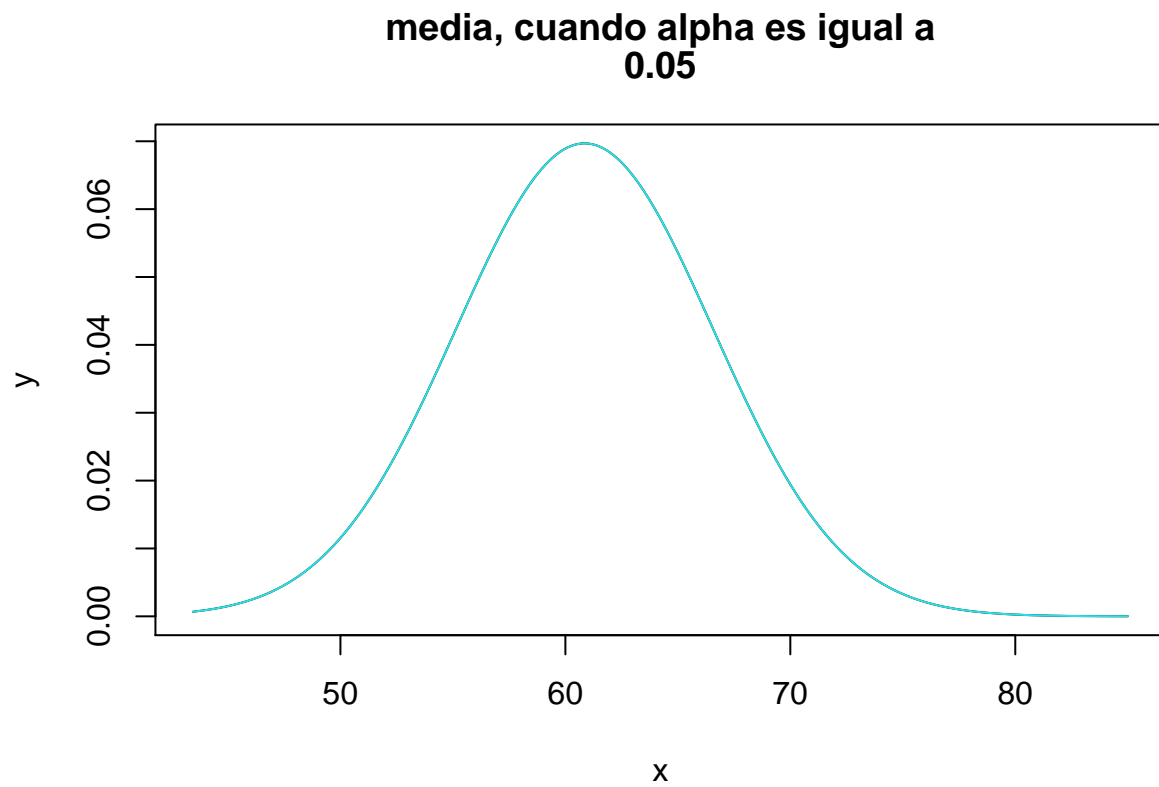
    plot(x2,y2,type = "l", main = c("varianza, cuando alpha es igual a",alpha))

    segments(x0=liminf1,x1=liminf1,y0=0,y1=dnorm(liminf1,mean(datos),sd(datos)),col = 2)

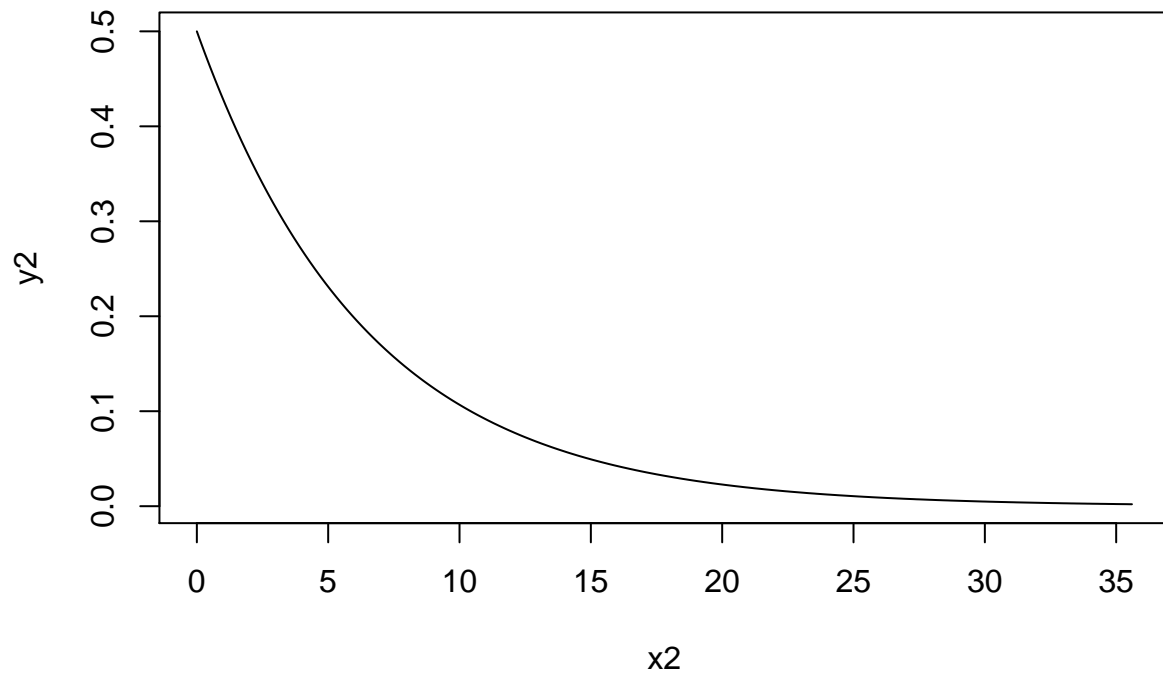
```

```
segments(x0=limsup1,x1=limsup1,y0=0,y1=dnorm(limsup1,mean(datos),sd(datos)),col = 2)

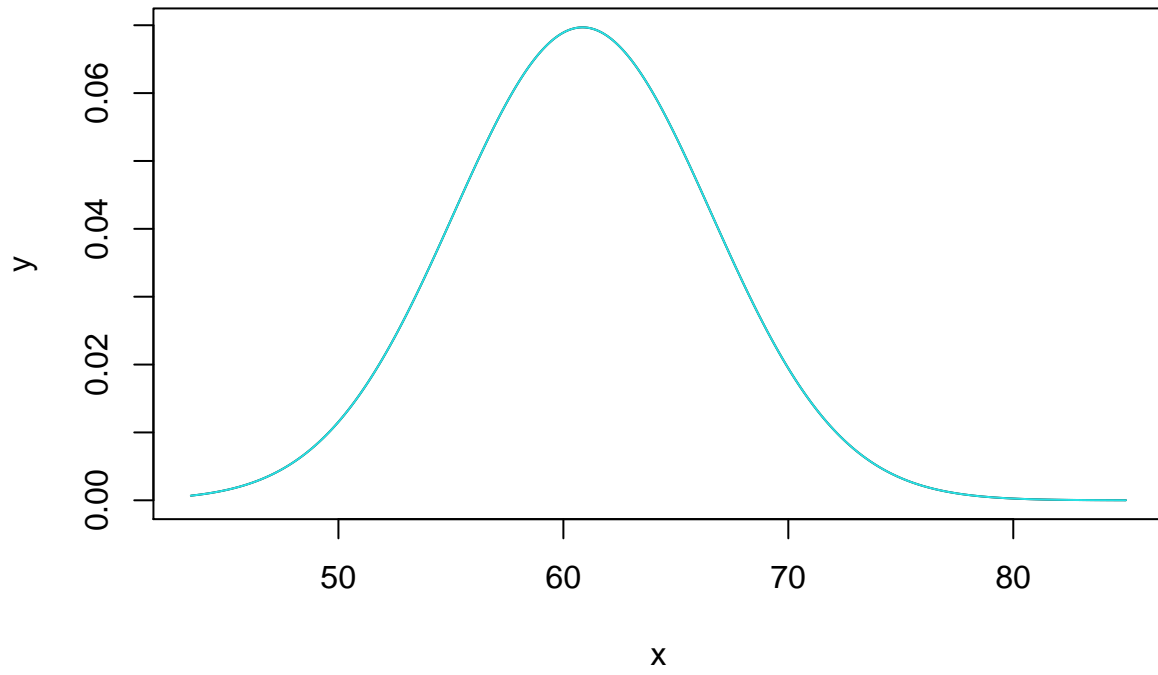
lines(x,y,col = 5)
}
```



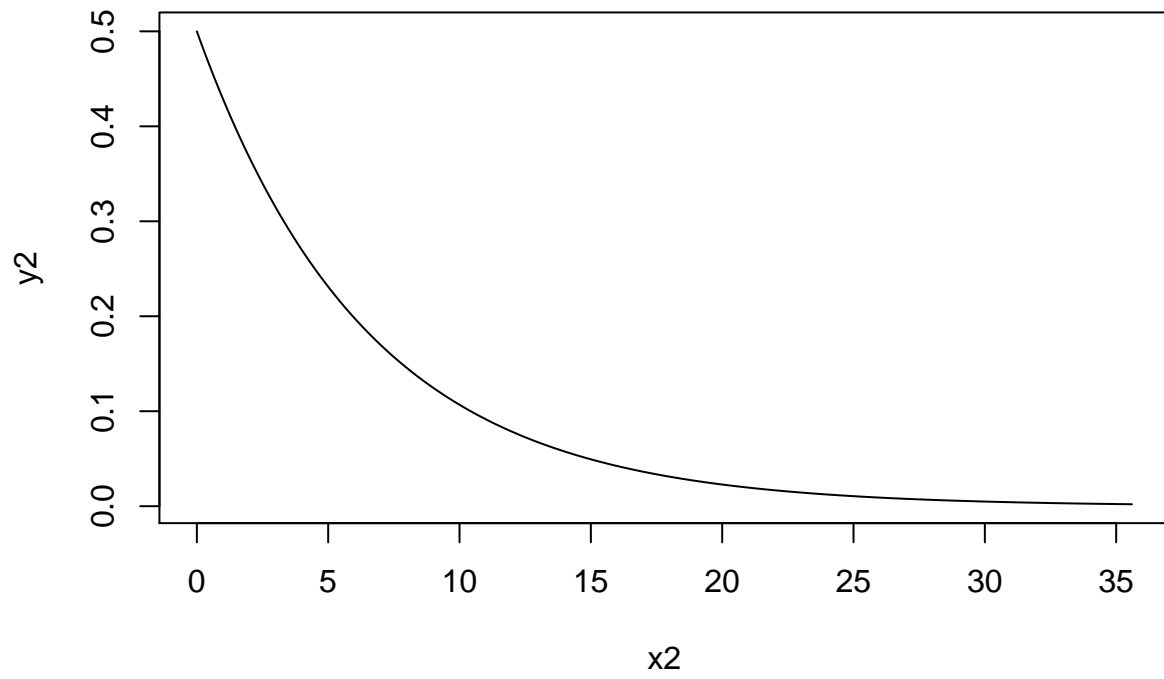
**varianza, cuando alpha es igual a
0.05**



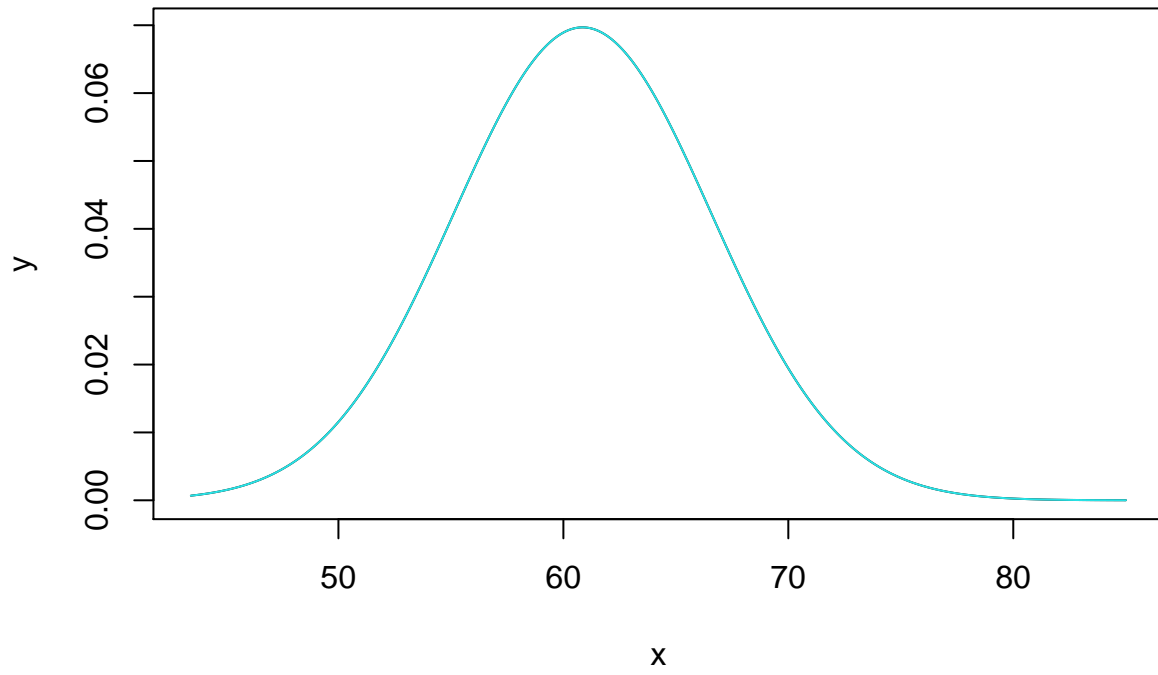
**media, cuando alpha es igual a
0.025**



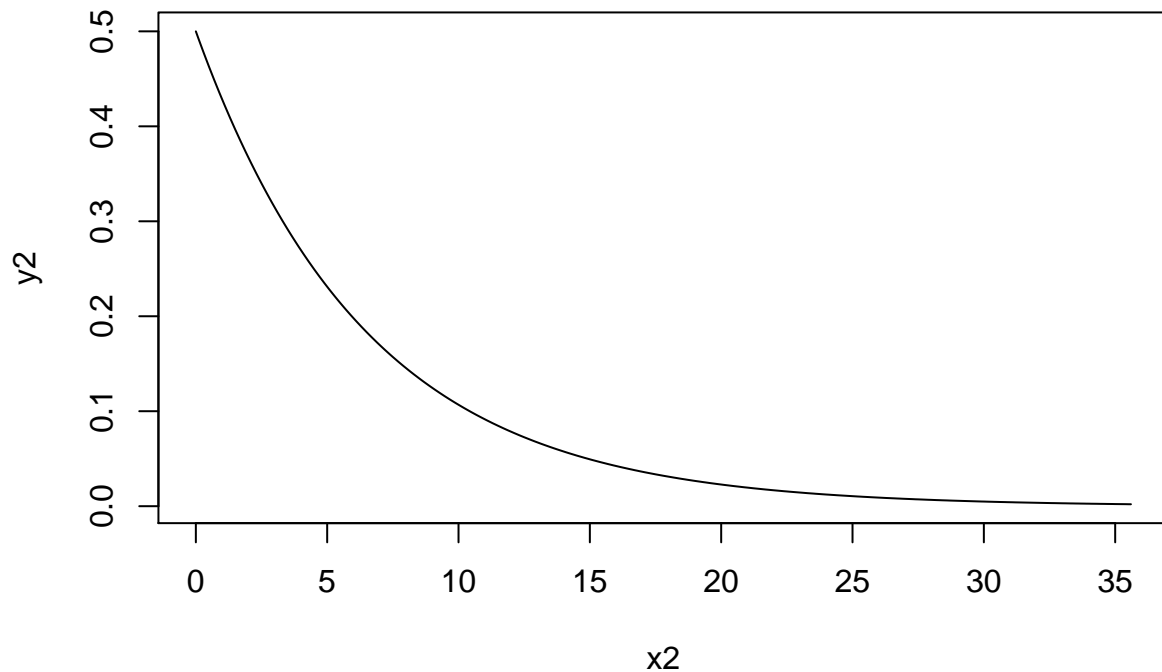
varianza, cuando alpha es igual a
0.05



media, cuando alpha es igual a
0.01



varianza, cuando alpha es igual a 0.05



Calcular y representar también los intervalos de confianza para otros tamaños de muestra: $n \in \{10, 50, 100\}$

```
l <- list(3,10,50,100)
lalpha <- list(0.05,0.025,0.01)

for (n in l){
  for (alpha in lalpha){

    if (n <= 30){
      q1 <- qt(alpha/2, n-1)*varianza*sqrt(n)+media
      q2 <- qt(1-(alpha/2), n-1)*varianza*sqrt(n)+media

      x <- seq(media - 5, media + 5, 0.1)
      y <- dt((x-media)/(varianza/sqrt(n)),n-1)
      plot(x,y,type = "l", col = 3, lwd = 3, main = c("n igual a",n))

      cord.x <- c(q1, seq(q1,q2,0.1),q2)
      cord.y <- c(0,(dt((seq(q1,q2,0.1)-media)/(varianza/sqrt(n)),n-1)),0)
      polygon(cord.x,cord.y, col = 3)

      #varianza

      q1v <- (n-1)*cuasivarianza/qchisq(1-(alpha/2),n-1)
      q2v <- (n-1)*cuasivarianza/qchisq(alpha/2,n-1)
```

```

xvar <- seq(0,3*cuasivarianza,0.1)
yvar <- dchisq((n-1)*(xvar/cuasivarianza),n-1)
plot(xvar,yvar, type = "l", col = 5, lwd = 3, main = c("n es igual a",n))

cord.xvar <- c(q1,seq(q1,q2,0.1),q2)
cord.yvar <- c(0,(dchisq(((seq(q1,q2,0.1)/cuasivarianza)*(n-1)),n-1)),0)

polygon(cord.xvar,cord.yvar, col = "red")
}
else{

q1 <- qnorm(alpha/2,media,(varianza/sqrt(n)))
q2 <- qnorm(1-(alpha/2),media,(varianza/sqrt(n)))

x <- seq(media - 5, media + 5, 0.1)
y <- dnorm(x, media, varianza/sqrt(n))
plot(x,y,type = "l", lwd=3, main = c("n es igual a",n))

cord.x <- c(q1,seq(q1,q2,0.1),q2)
cord.y <- c(0,dnorm(seq(q1,q2,0.1),media,varianza/sqrt(n)),0)
polygon(cord.x,cord.y,col = "green")

#varianza
q1v <- (n-1)*cuasivarianza/qchisq(1-(alpha/2),n-1)
q2v <- (n-1)*cuasivarianza/qchisq(alpha/2,n-1)

xvar <- seq(3*cuasivarianza,4*cuasivarianza,0.1)
yvar <- dchisq((n-1)*(xvar/cuasivarianza),n-1)
plot(xvar,yvar,type = "l",lwd = 3, col = 4, main = c("alpha igual a",alpha))

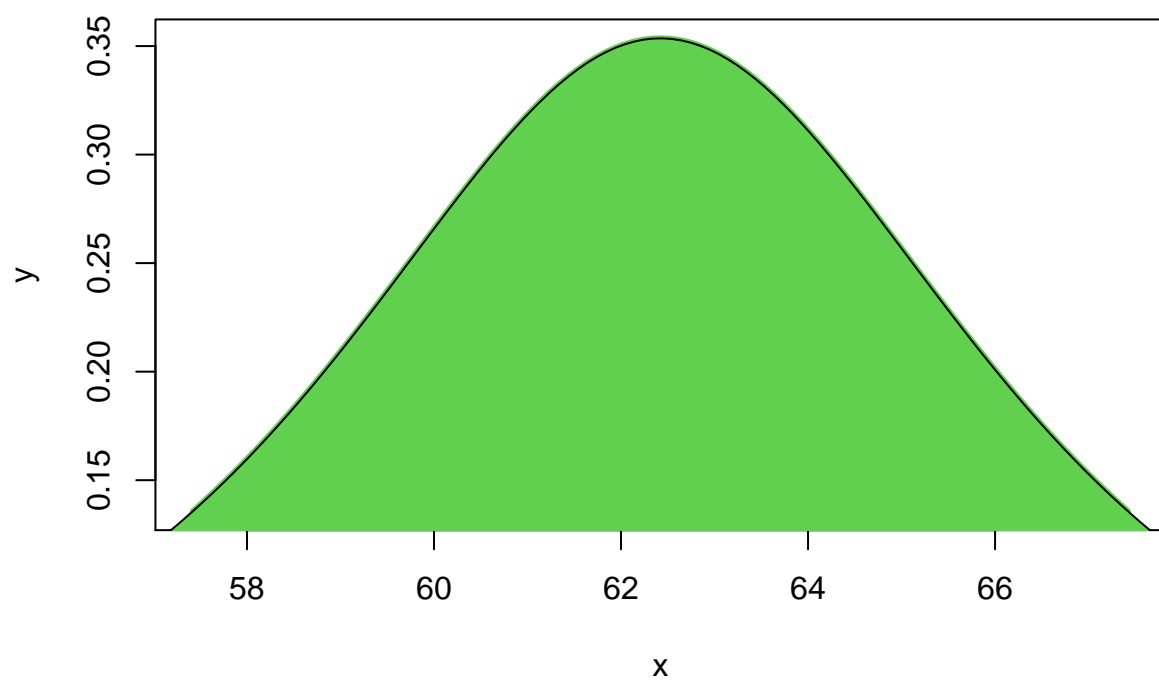
cord.x <- c(q1,seq(q1,q2,0.1),q2)
cord.y <- c(0,(dchisq(((seq(q1,q2,0.1)/cuasivarianza)*(n-1)), n-1)),0)

polygon(cord.x,cord.y, col = "blue")

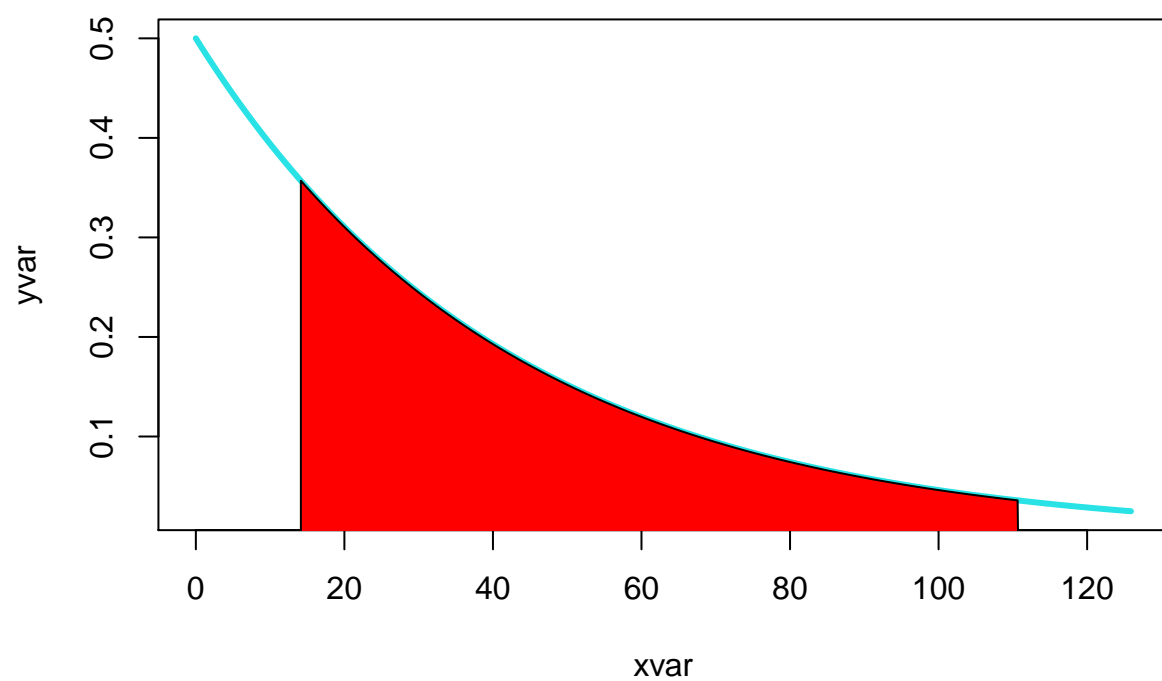
}
}
}

```

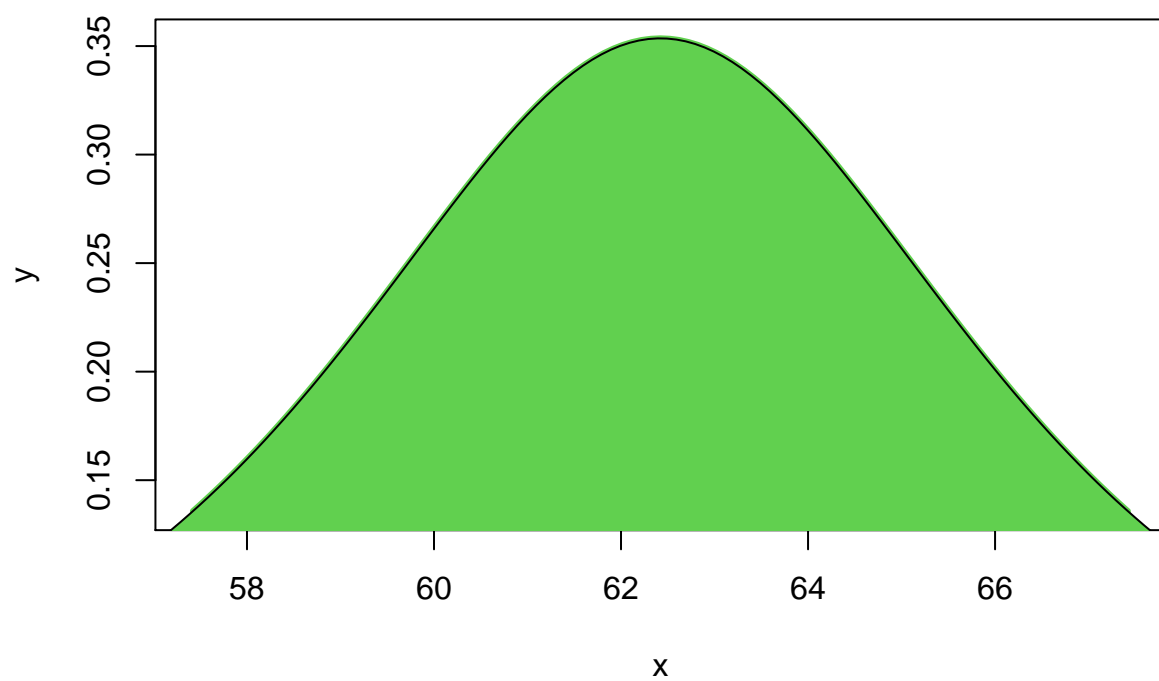
n igual a
3



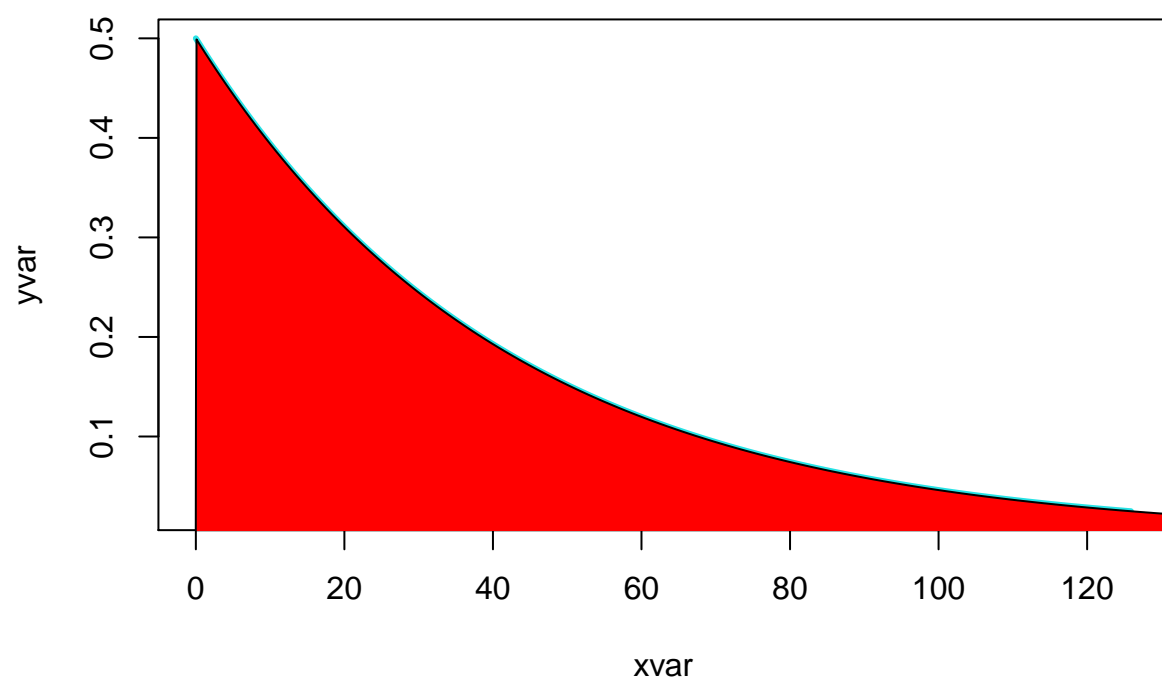
n es igual a
3



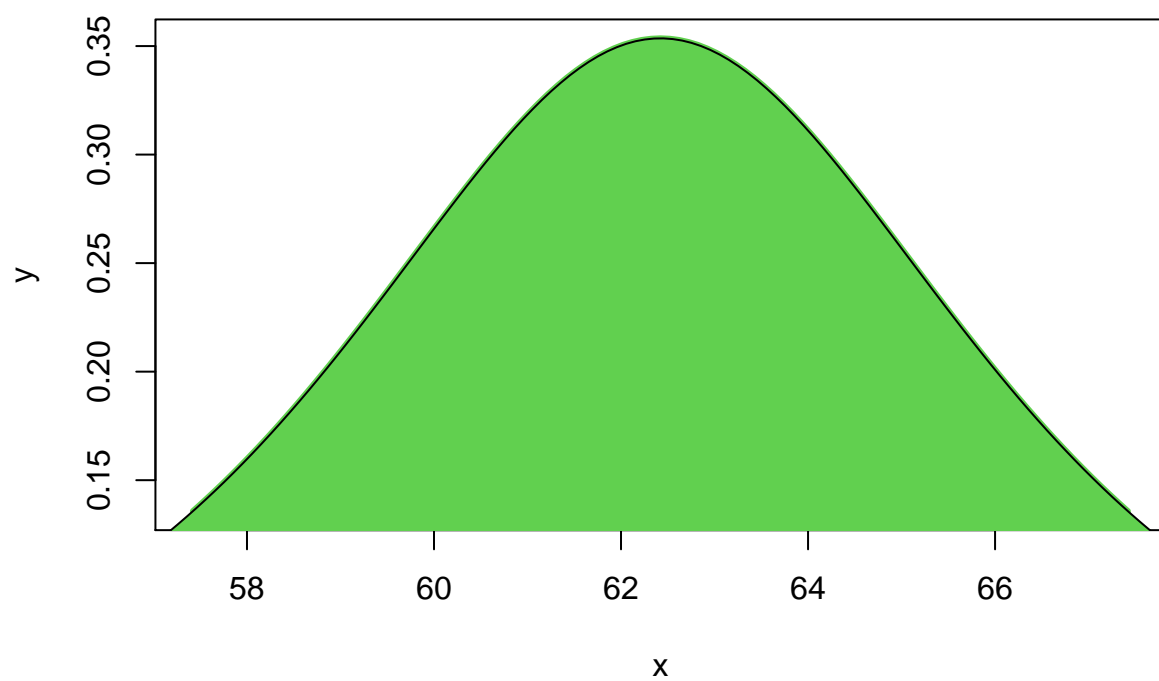
n igual a
3



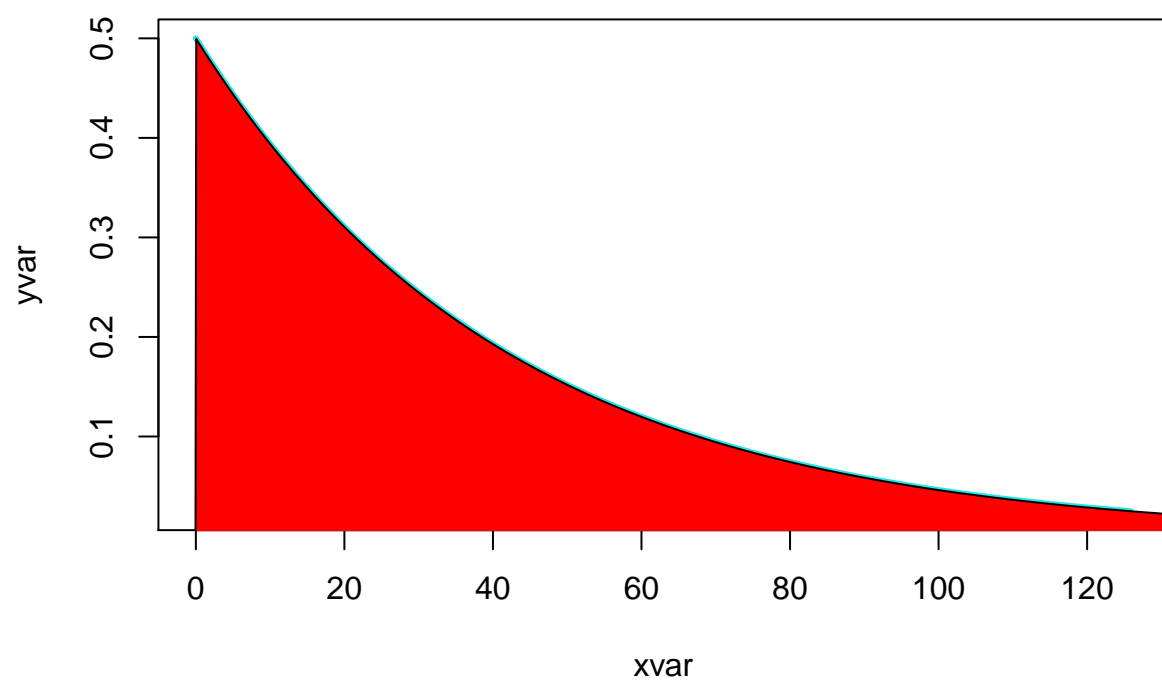
n es igual a
3

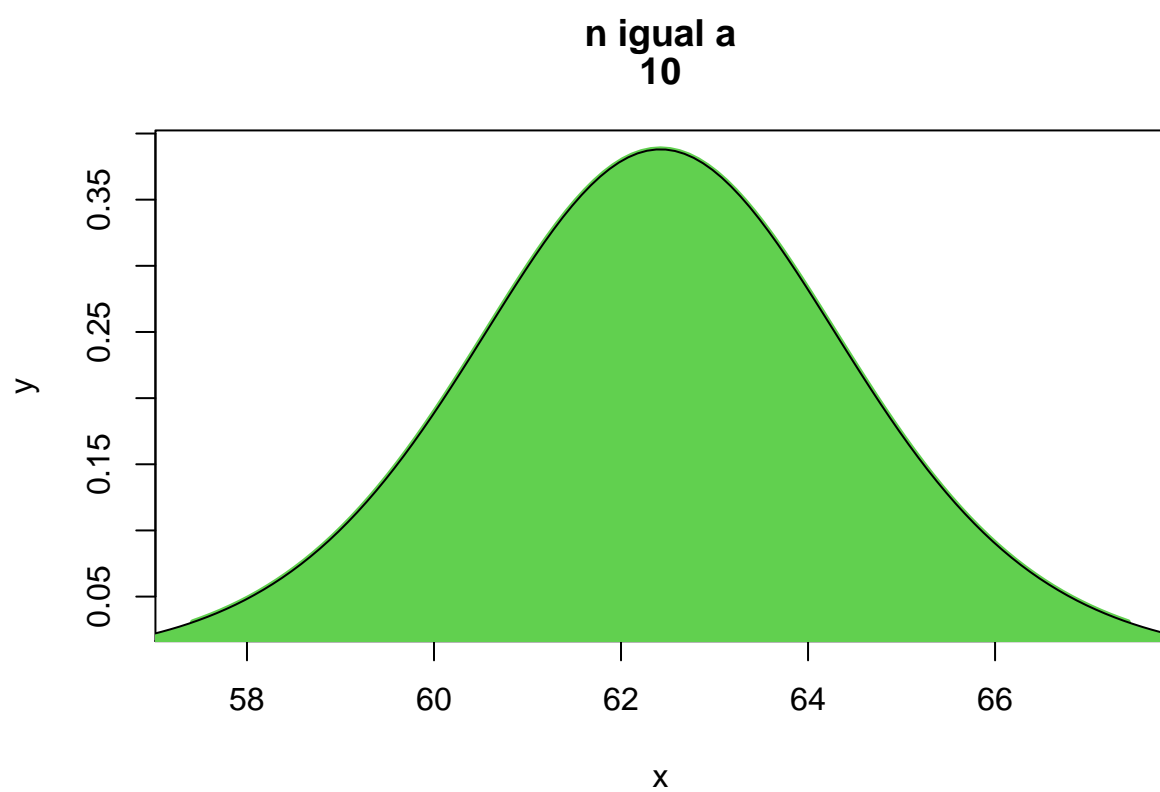


n igual a
3

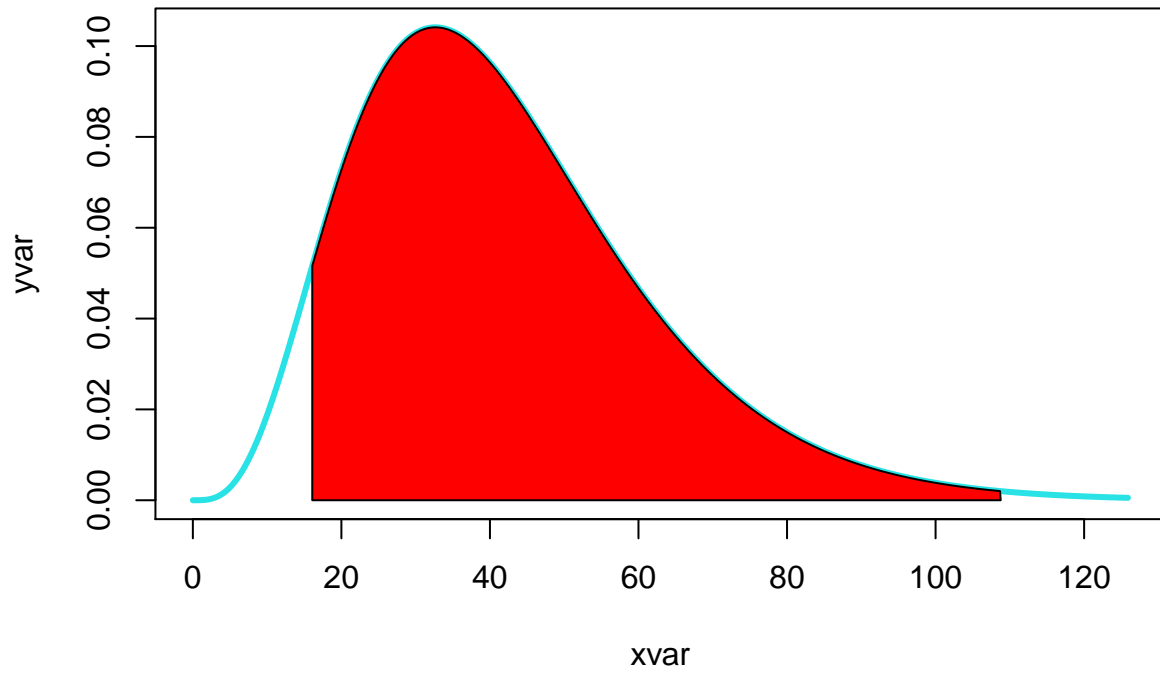


n es igual a
3

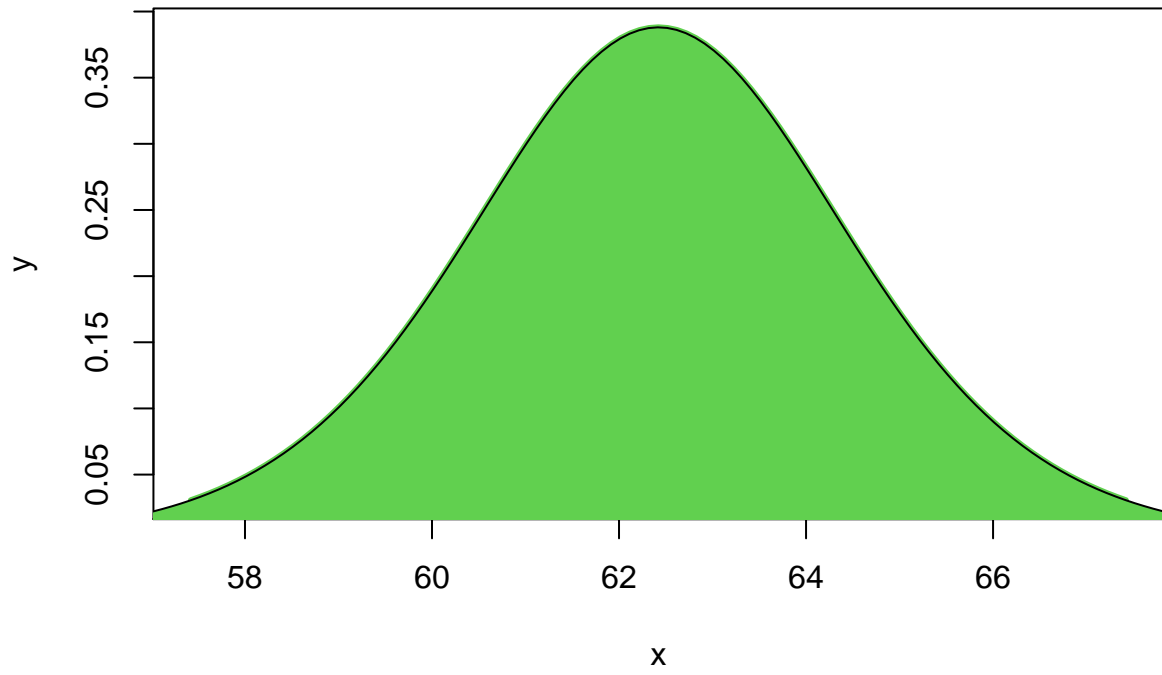




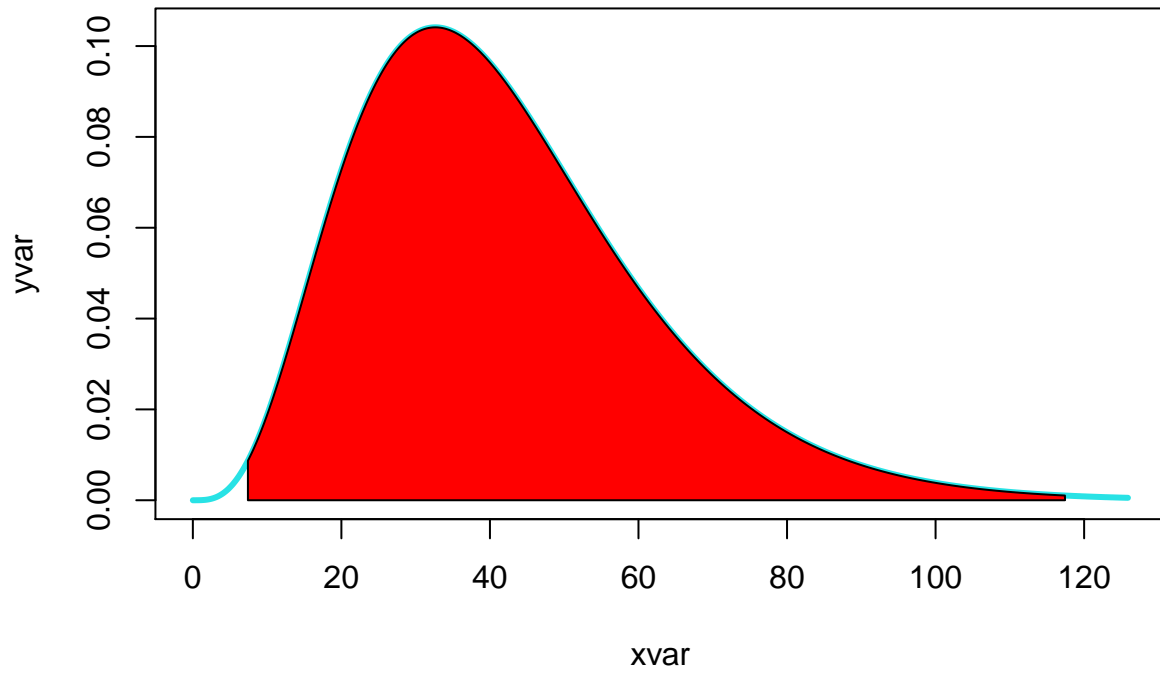
n es igual a
10



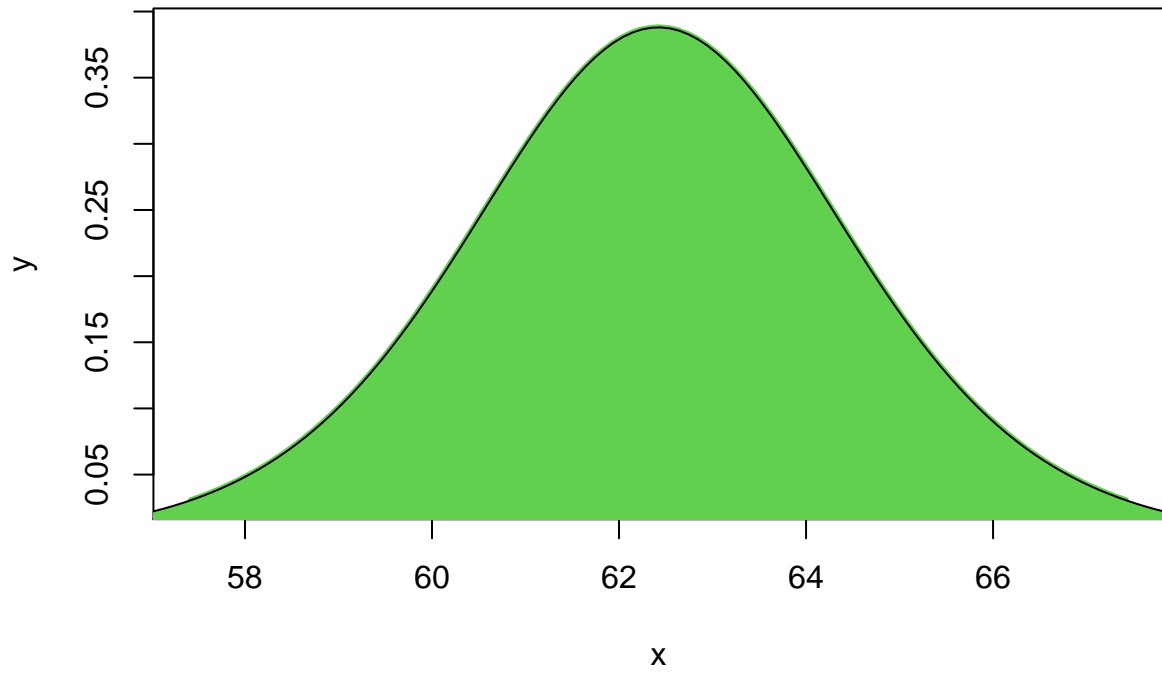
n igual a
10



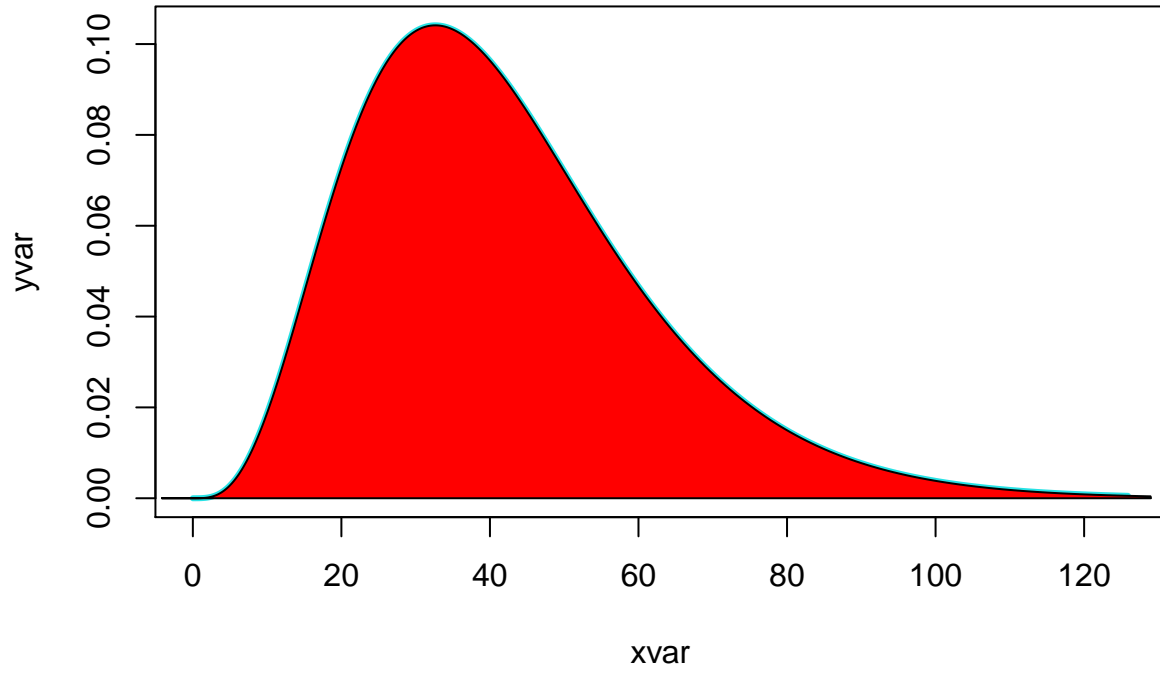
n es igual a
10



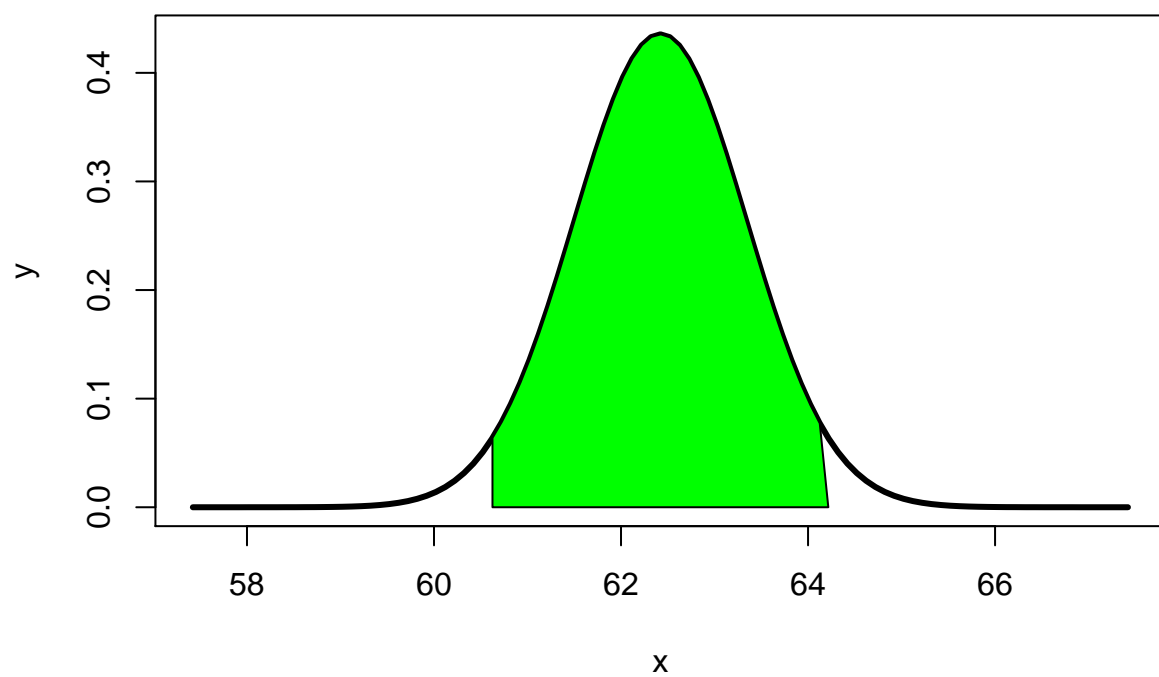
n igual a
10



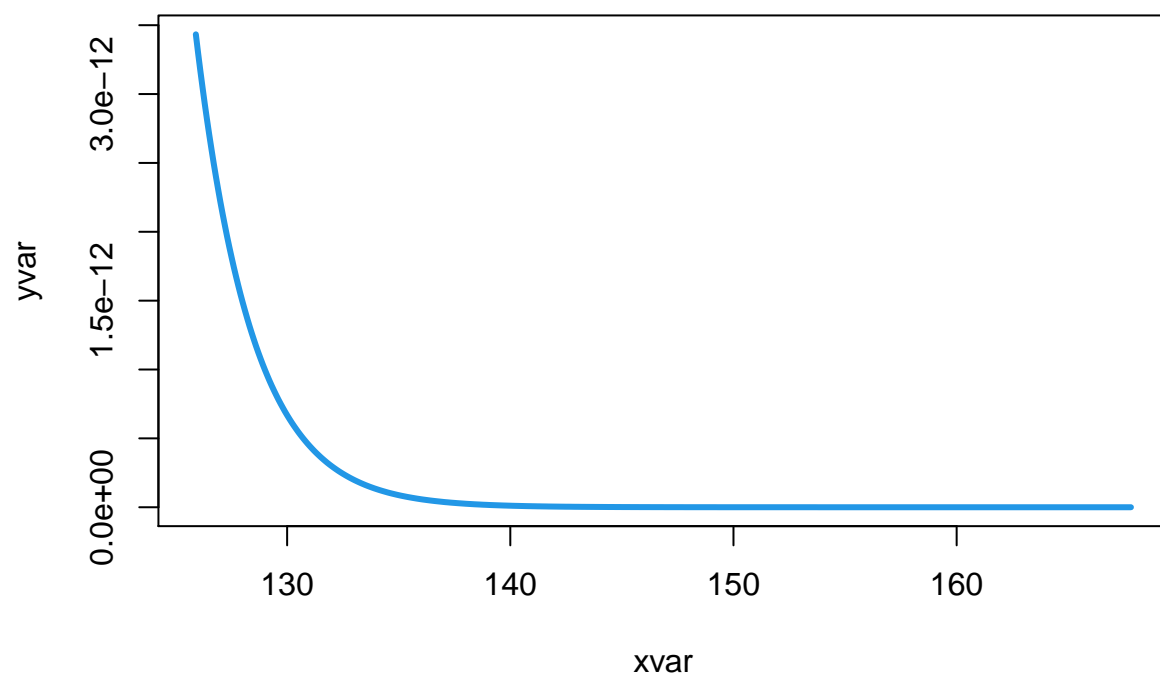
n es igual a
10



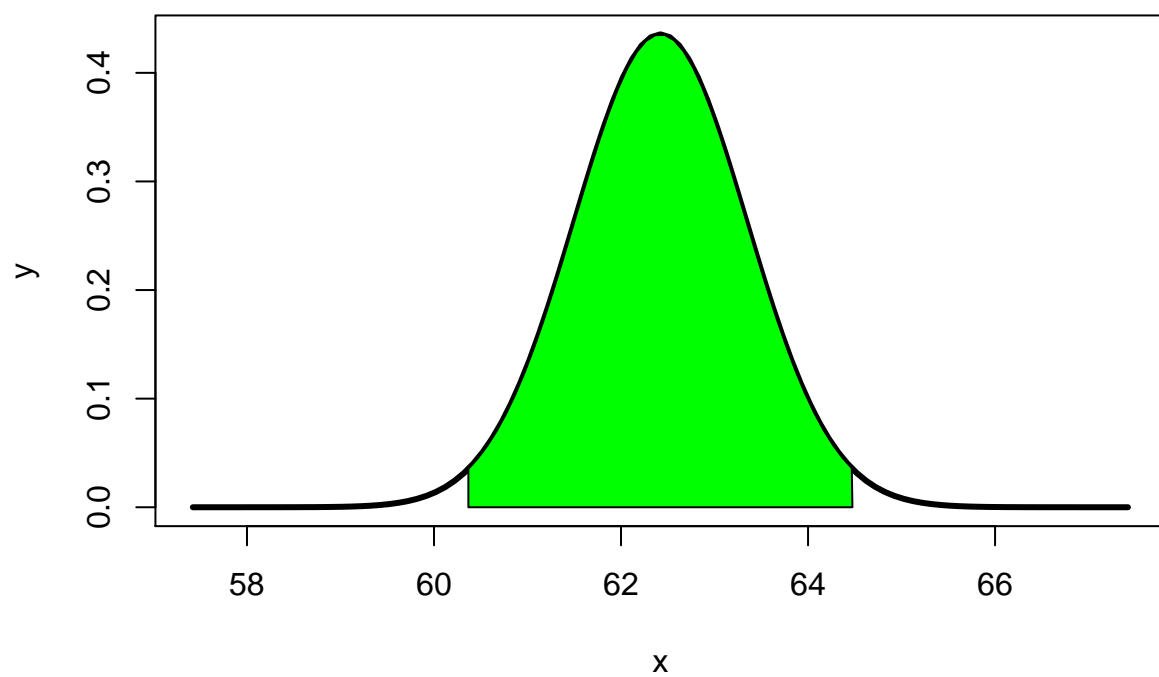
n es igual a
50



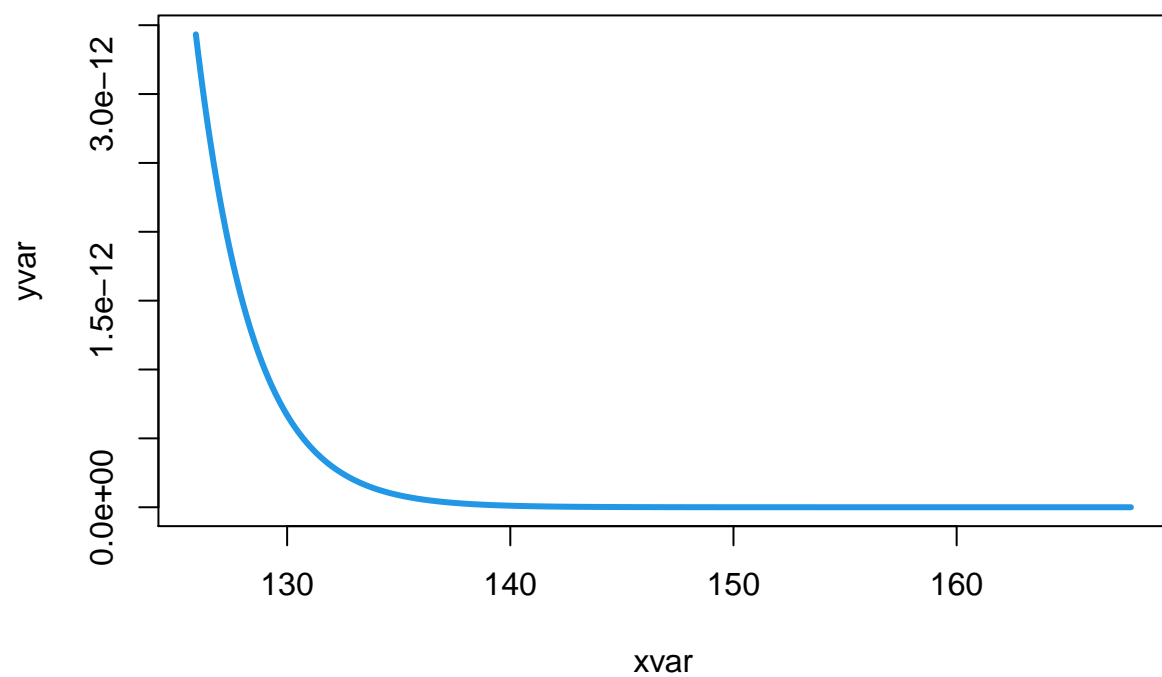
alpha igual a
0.05



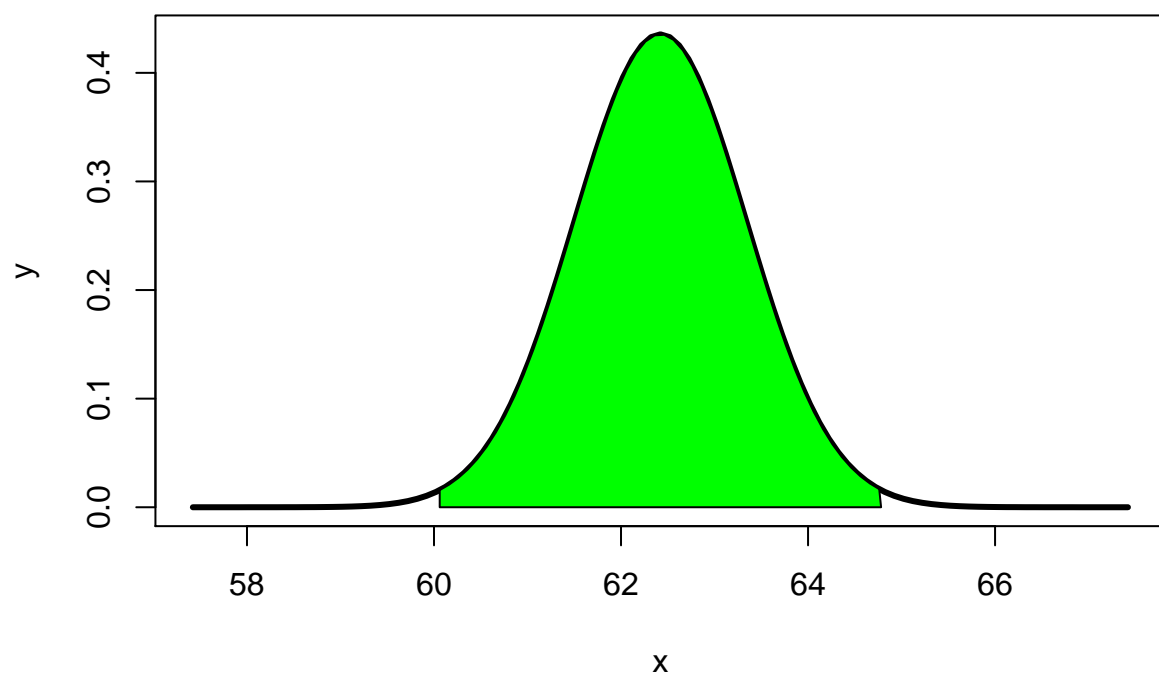
n es igual a
50



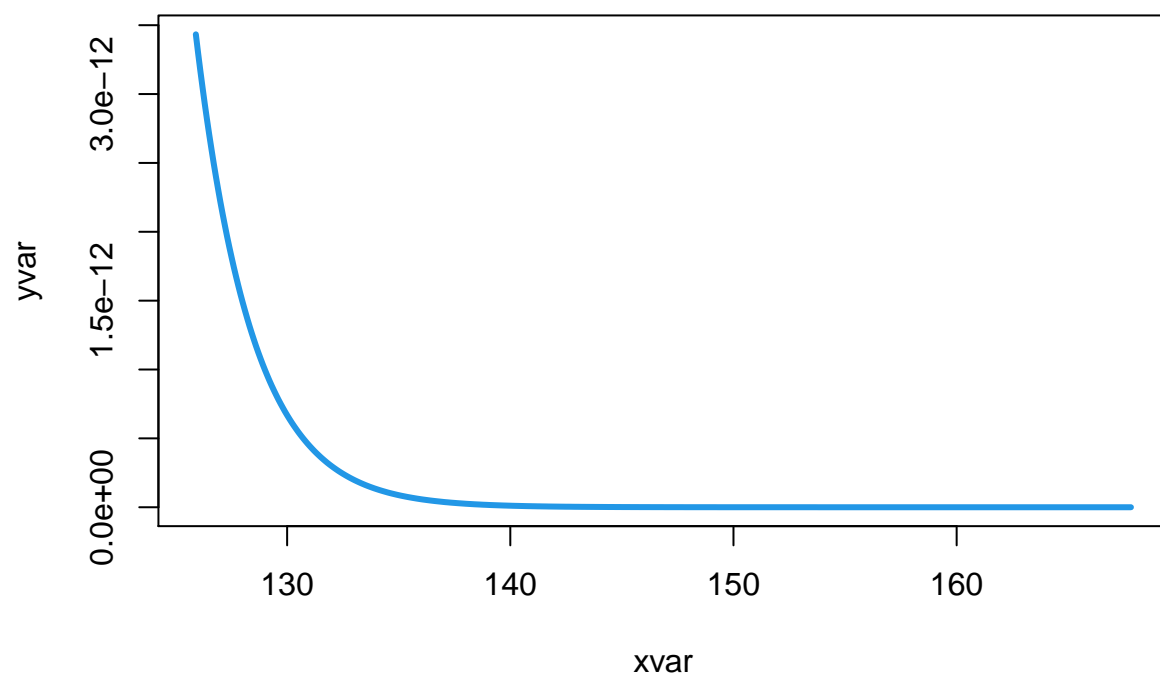
**alpha igual a
0.025**



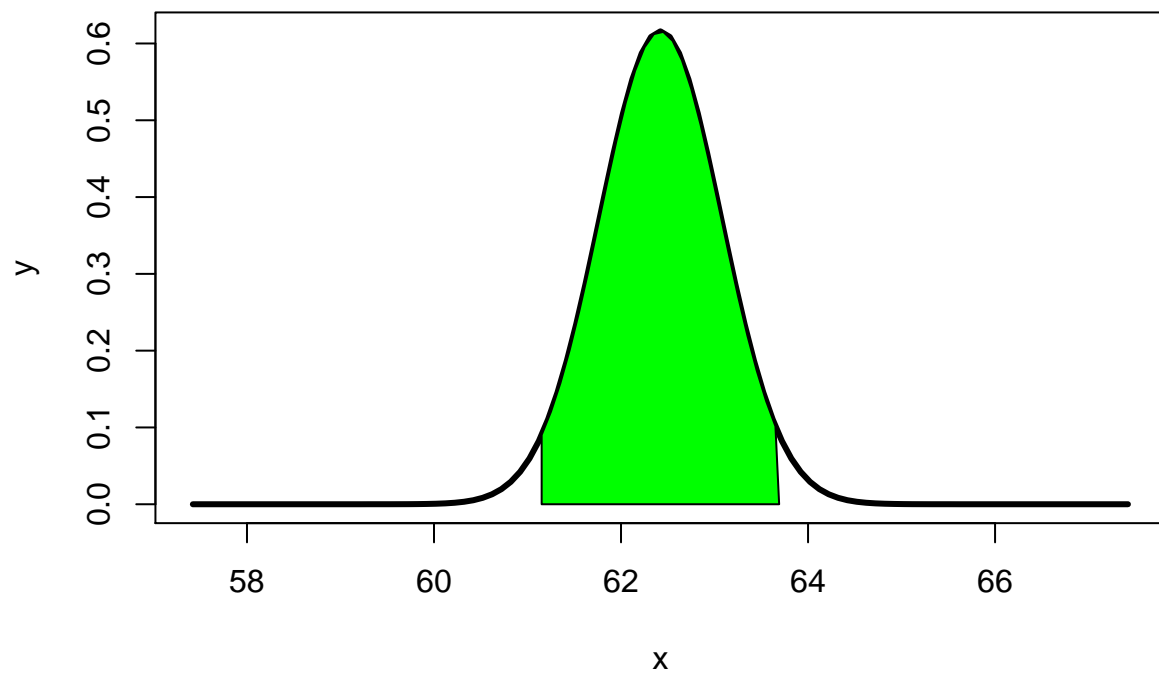
n es igual a
50

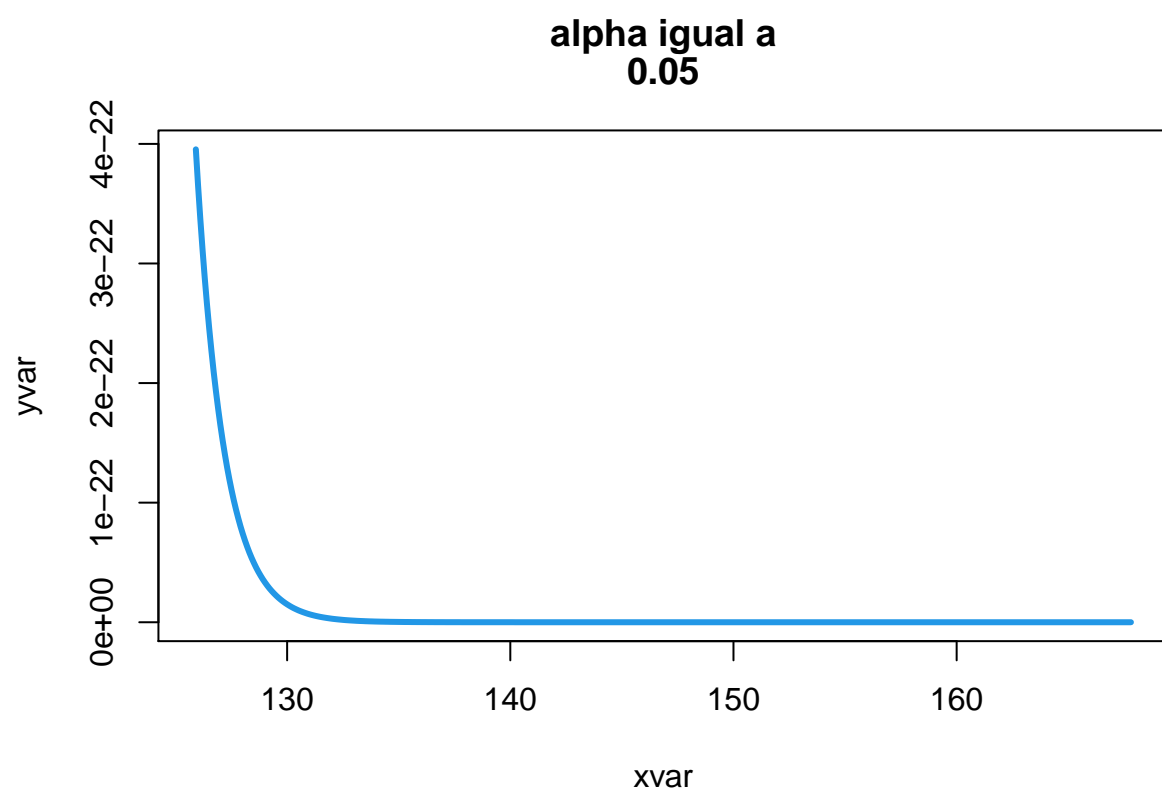


alpha igual a
0.01

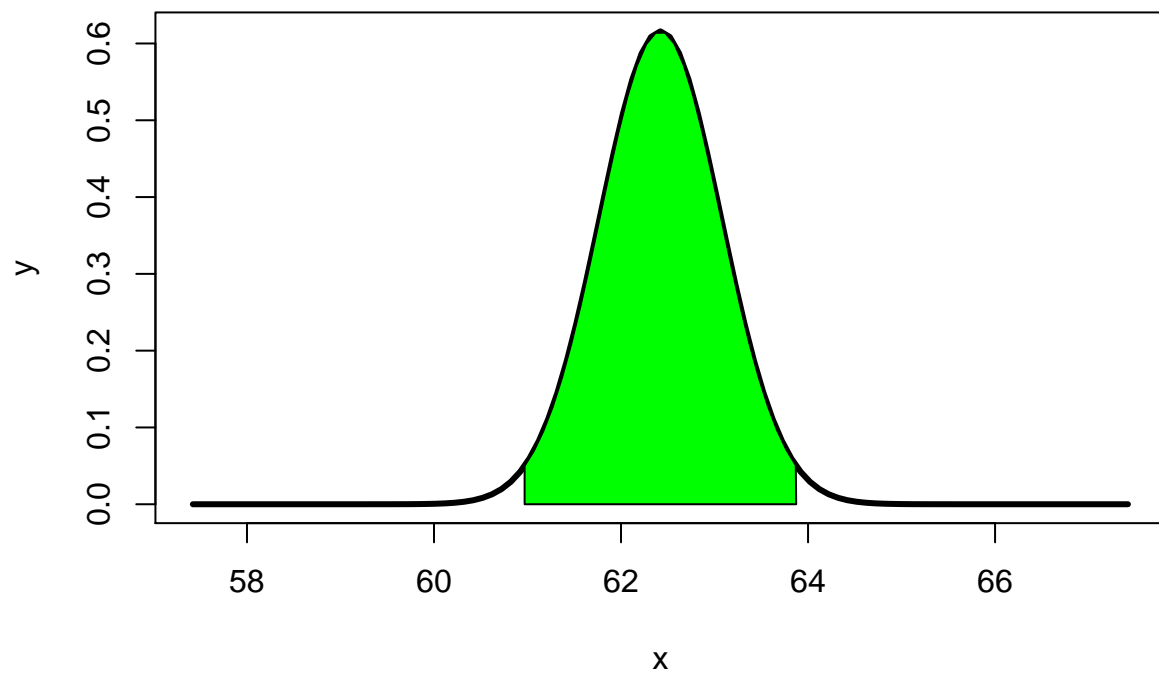


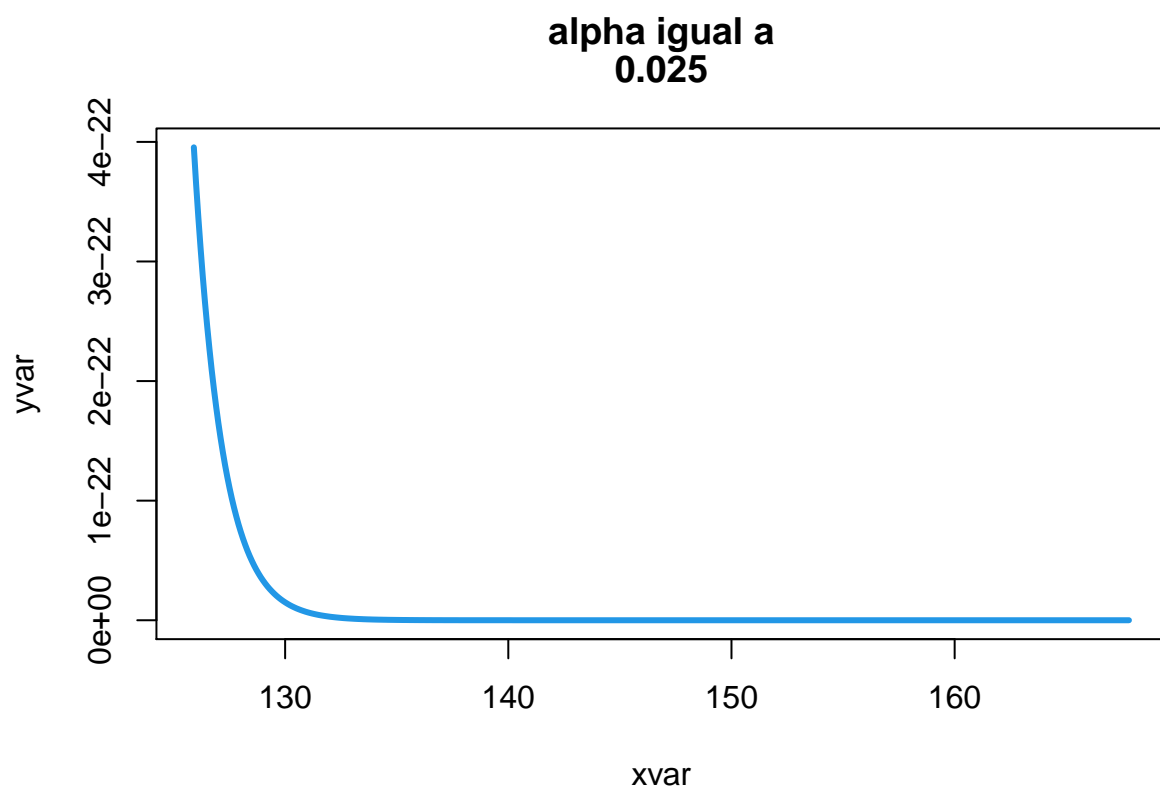
n es igual a
100



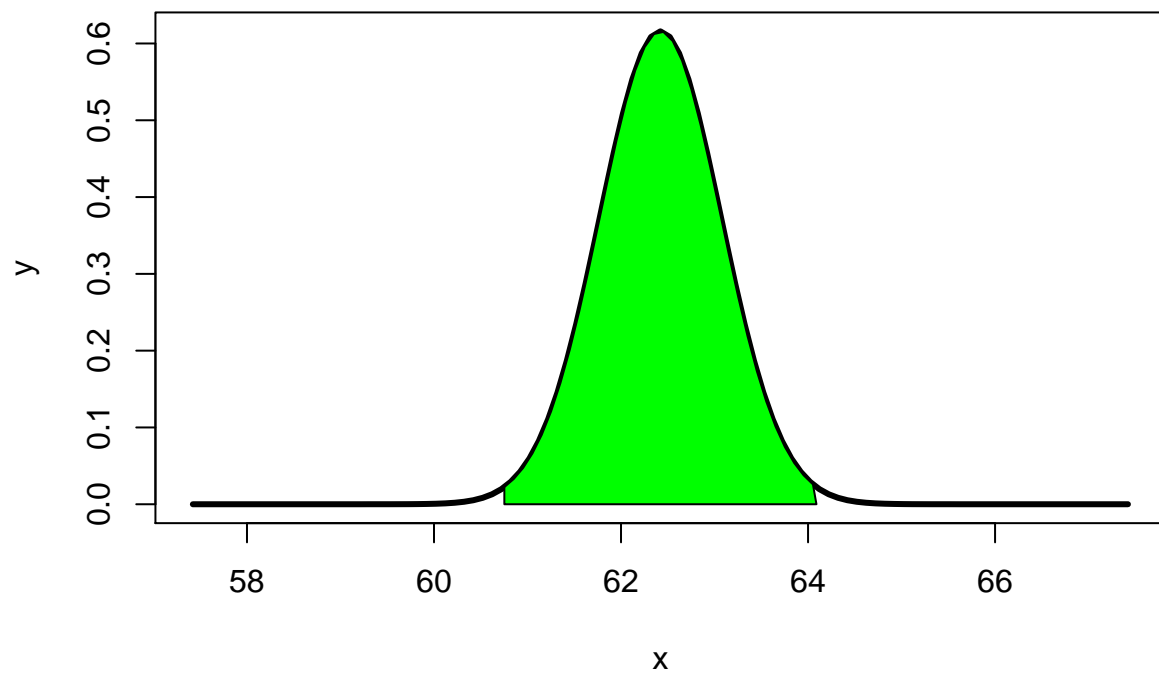


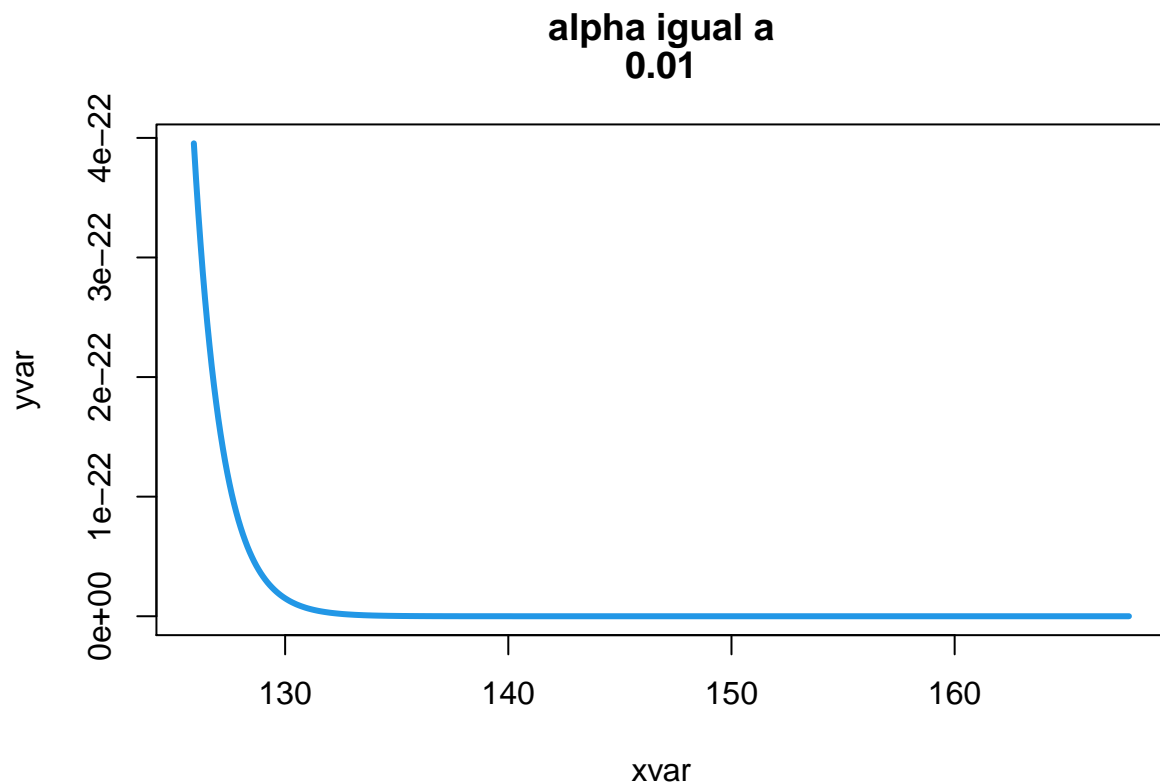
n es igual a
100





n es igual a
100





Tarea 4 : test de hipótesis.

Considerar las estimaciones de μ y σ^2 calculadas en el apartado 2, con $n = 3$

Plantear el test de hipótesis adecuado para contrastar estos valores. Realizar el test calculando el p-valor con tamaños de muestra $n \in \{10, 50, 100\}$. Concluir si se acepta o se rechaza la hipótesis. Representar gráficamente la región correspondiente al p-valor.

```
#volvemos a poner los datos del apartado 2

muestra <- sample(datos, n)
media <- mean(muestra)
cuasivarianza <- var(muestra)

#un bucle para diferentes tamaños de la muestra

for (i in 1:3){
  if (i ==1) {
    n <- 10
    m <- 1
    v <- 0.5
    mediav <- media
    varianzav <- varianza
  }

  if (i ==2) {
```

```

n <- 50
m <- 2
v <- 1
mediax <- media
varianzax <- varianza
}

if (i == 3) {
  n <- 100
  m <- 3
  v <- 1.5
  mediax <- media
  varianzax <- varianza
}

#emplearemos las medias y varianzas de H0 y la desviación típica de H1 para realizar las gráficas

#Media

if (n >= 30) {
  xm <- seq(media - (4/m) * cuasivarianza, media + (4/m) * cuasivarianza, 0.1)
  ym <- dnorm(xm, media, sqrt(varianzax/n))
} else {
  xm <- seq(media - (4/m) * cuasivarianza, media + (4/m) * cuasivarianza, 0.1)
  ym <- dt((media - xm) / sqrt(varianzax/n), n - 1)
}

if (i == 1) {
  plot(xm, ym, type = "l", col = 6, main = "Prueba de hipotesis_Student (n = 10)")
}

if (i == 2) {
  plot(xm, ym, type = "l", col = 6, main = "Prueba de hipotesis_norm(n = 50)")
}

if (i == 3) {
  plot(xm, ym, type = "l", col = 6, main = "Prueba de hipotesis_norm(n = 100)")
}

abline(v = mediax)
abline(h = 0)
abline(v = media, col = 2)
text(media, 0, "media", col = 2)

#Dibujamos el p-valor

```

```

if (n>=30){
  if (mediax < media){
    xm <- seq(mediax-4*sqrt(varianzax),mediax,0.001)
    ym <- dnorm(xm,media,sqrt(varianzax/n))
  }
  else{
    xm <- seq(mediax,mediax+4*sqrt(varianzax),0.001)
    ym <- dnorm(xm, media,sqrt(varianzax/n))
  }
}

else{
  if (mediax < media ){
    xm <- seq(mediax-4*sqrt(varianzax),mediax,0.001)
    ym <- dt((media-xm)/sqrt(varianzax/n),n-1)
  }
  else{
    xm <- seq(mediax,mediax+4*sqrt(varianzax),0.001)
    ym <- dt((media-xm)/sqrt(varianzax/n),n-1)
  }
}

lines(xm,ym,type = "h")

  if (i==1){
text(media,0,"media (n = 10)", col = "skyblue")
}

if (i==2){
text(media,0,"media (50)", col = "skyblue")
}

if (i==3){
text(media,0,"media (100)", col = "skyblue")
}

#Calculamos el p-valor

if (n>=30){
  if (mediax>media) {
    p <- 1-pnorm((mediax - media)/sqrt(varianzax/n))
  }
  else{
    p <- pnorm((mediax-media)/sqrt(varianzax/n))
  }
}

else{
  if (mediax > media) {
    p <- 1-pt((mediax-media)/sqrt(varianzax/n),n-1)
  }
  else{

```

```

    p <- pt((mediax-media)/sqrt(varianzax/n),n-1)
  }
}

print(p)

if (p >= 0.05){
  print("como el p-valor(media) es mayor que alpha = 0.05, no se rechaza la hipotesis nula")
}
else{
  print("como el p-valor(media) es menor que alpha = 0.05, es incierta la hipotesis nula, por tanto")
}

#Varianza

xv <- seq(0, (4/v)*varianza, 0.1)
yv <- dchisq((n-1)*xv/varianza,n-1)

if (i == 1){
  plot(xv, yv, type = "l", col = 6, main = "Prueba de hipotesis_chi (n = 10)")
}

if (i ==2){
  plot(xv, yv, type = "l", col = 6, main = "Prueba de hipotesis_chi (n = 50)")
}

if (i ==3){
  plot(xv, yv, type = "l", col = 6, main = "Prueba de hipotesis_chi (n = 100)")
}

abline(v = varianzax)
abline(h = 0)
abline(v = varianza,col = 2)
text(varianza,0,"varianza",col = 2)

#Dibujamos el p-valor

if (varianzax<varianza){
  xv <- seq(varianzax-10*cuasivarianza,varianzax,0.001)
  yv <- dchisq((n-1)*xv/varianza,n-1)
}
else{
  xv <- seq(varianzax,varianzax+10*cuasivarianza,0.001)
  yv <- dchisq((n-1)*xv/varianza,n-1)
}
lines(xv,yv,type = "h")

```

```

if (i==1){
  text(varianza,0, "varianza (n = 10)", col = "skyblue")
}
if (i==2){
  text(varianza,0, "varianza (n = 50)", col = "skyblue")
}
if (i==3){
  text(varianza,0, "varianza (n = 100)", col = "skyblue")
}

#calculamos el p-valor

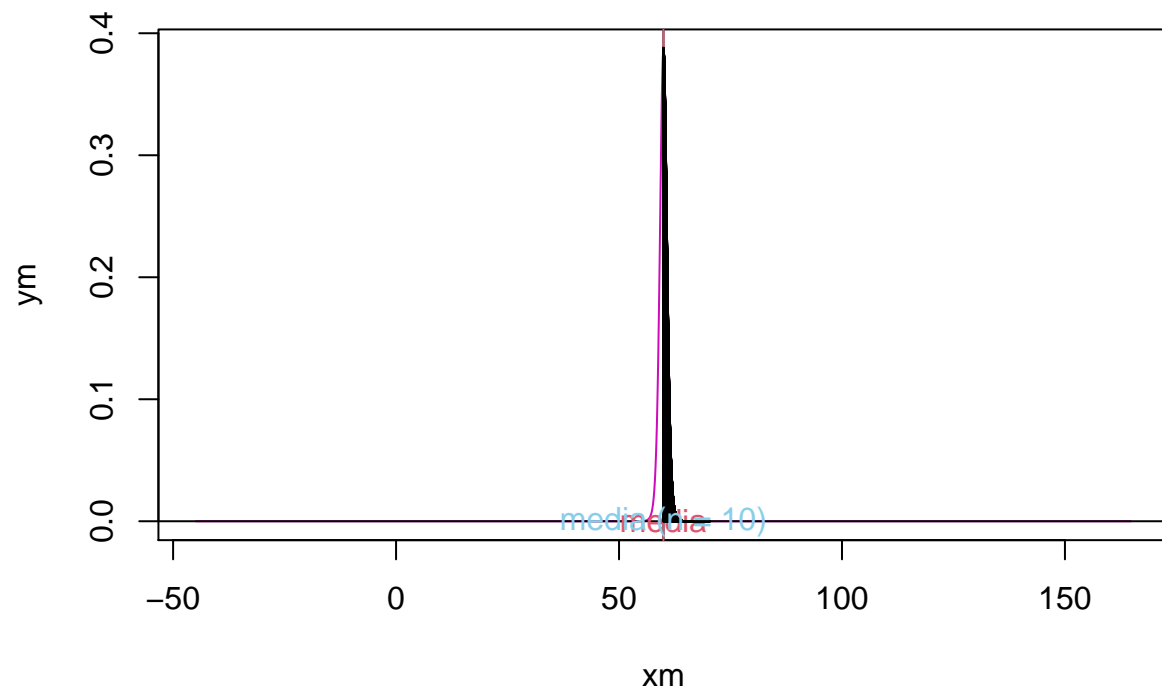
if (varianzax>varianza){
  pvarianza <- 1-pchisq(n-1,(n-1)*varianzax/varianza)
}
else{
  pvarianza <- pchisq(n-1,(n-1)*varianzax/varianza)
}
print(pvarianza)

if (pvarianza>= 0.05){
  print("como el p-valor (chi) es mayor que alpha = 0.05, no se rechaza la hipotesis nula")
}
else{
  print("como el p-valor (chi) es menor que alpha = 0.05, es incierta la hipotesis nula, por tanto ")
}

}

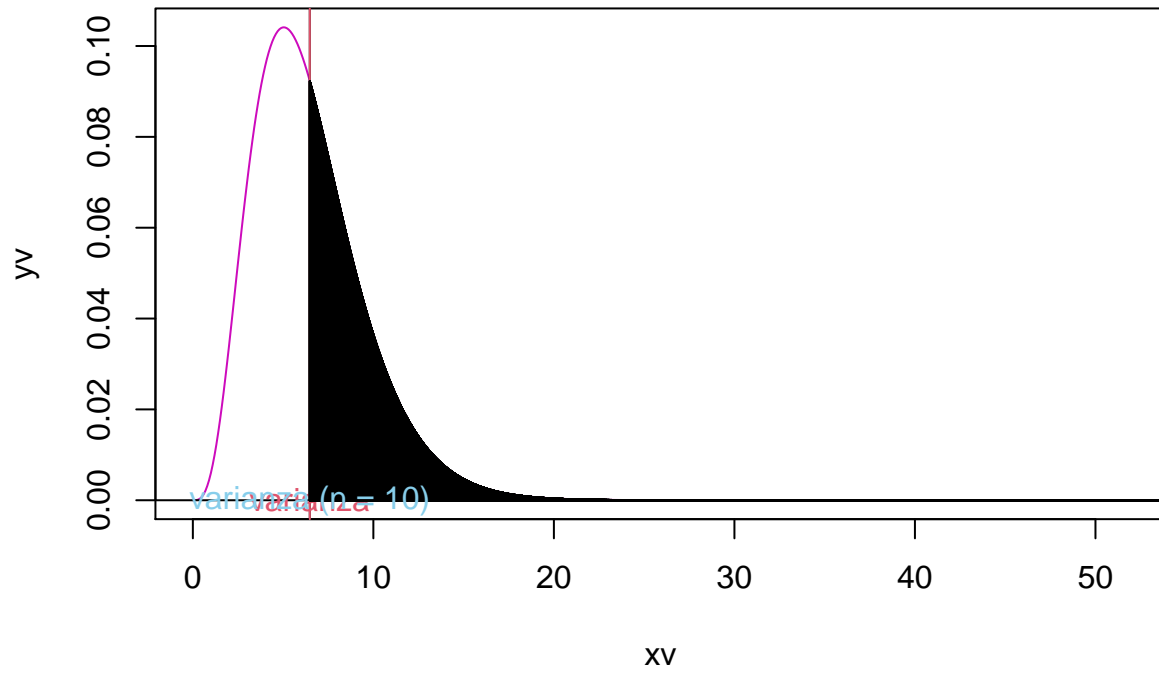
```

Prueba de hipotesis_Student (n = 10)



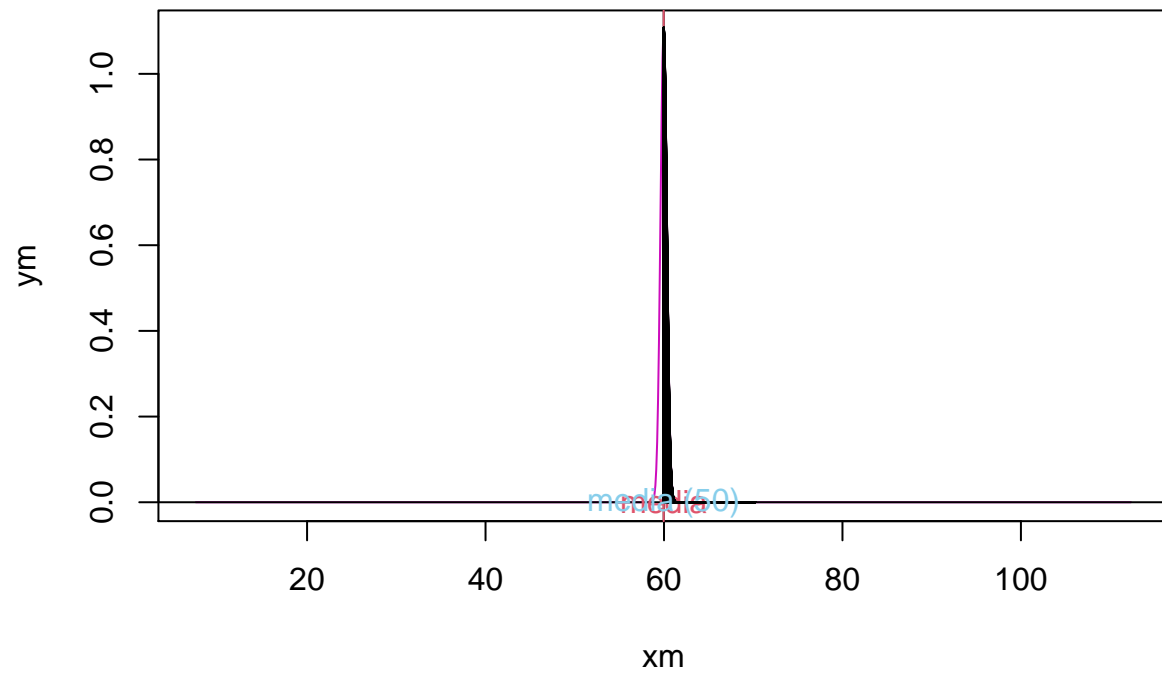
```
## [1] 0.5
## [1] "como el p-valor(media) es mayor que alpha = 0.05, no se rechaza la hipotesis nula"
```


Prueba de hipotesis_chi (n = 10)



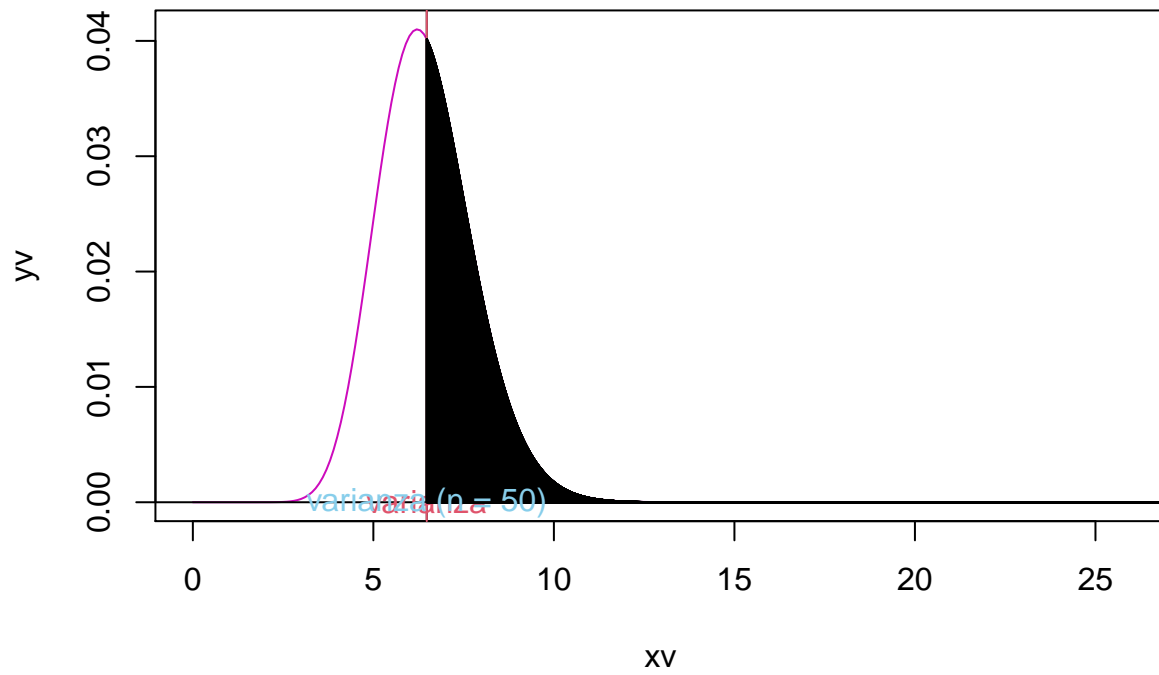
```
## [1] 0.5627258
## [1] "como el p-valor (chi) es mayor que alpha = 0.05, no se rechaza la hipotesis nula"
```

Prueba de hipotesis_norm(n = 50)



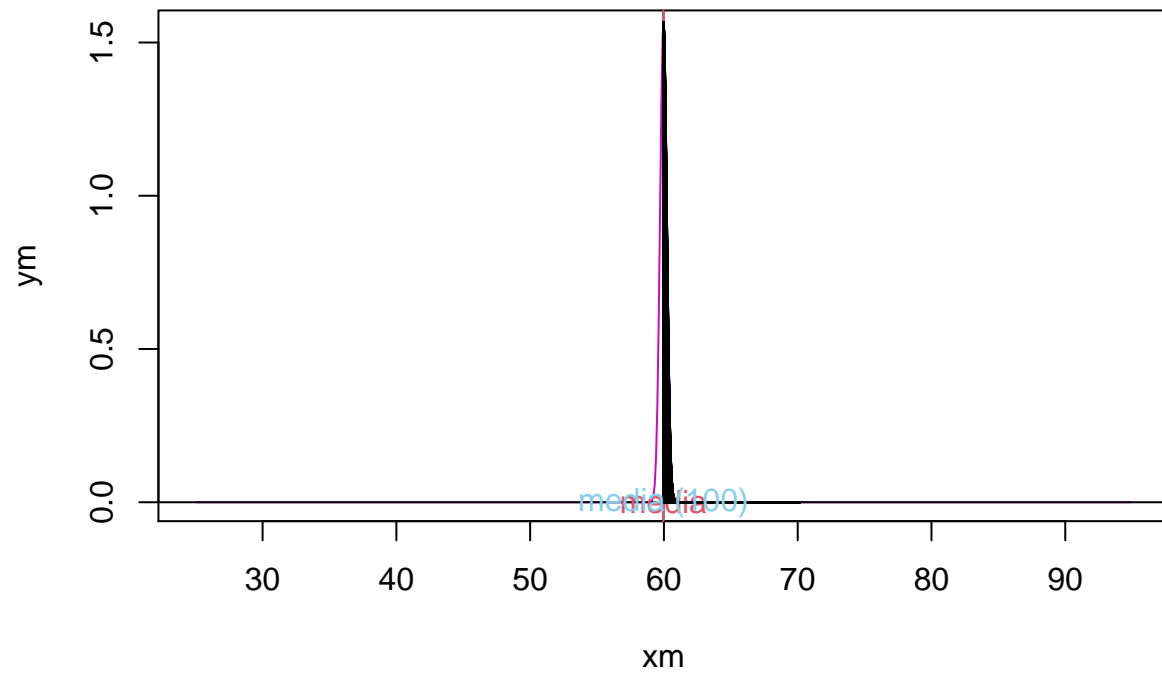
```
## [1] 0.5
## [1] "como el p-valor(media) es mayor que alpha = 0.05, no se rechaza la hipotesis nula"
```

Prueba de hipotesis_chi (n = 50)



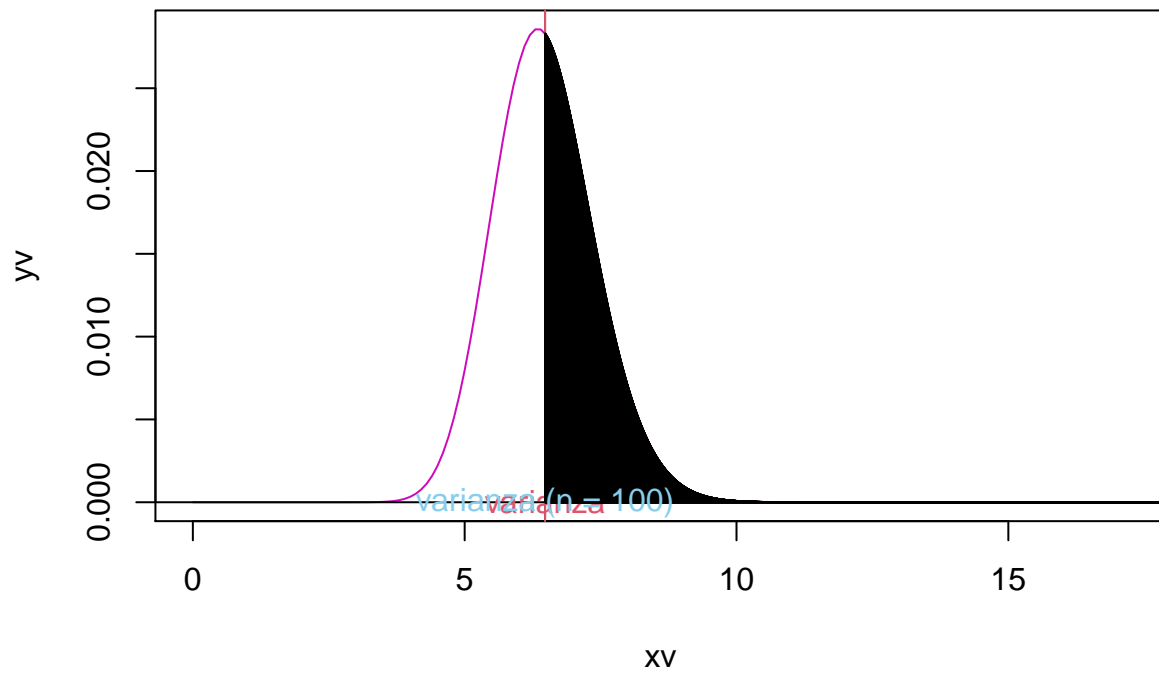
```
## [1] 0.5268717
## [1] "como el p-valor (chi) es mayor que alpha = 0.05, no se rechaza la hipotesis nula"
```

Prueba de hipotesis_norm(n = 100)



```
## [1] 0.5
## [1] "como el p-valor(media) es mayor que alpha = 0.05, no se rechaza la hipotesis nula"
```

Prueba de hipotesis_chi (n = 100)



```
## [1] 0.5189031
## [1] "como el p-valor (chi) es mayor que alpha = 0.05, no se rechaza la hipotesis nula"
```