Ampliación de Inferencia Estadística

Tercero Grado de Estadística

Universidad de Sevilla

Tema 5: Inferencia Bayesiana en el modelo Normal

Práctica-R-5. Segunda parte.

La Normal-GammaI

```
library("plot3Drgl")
```

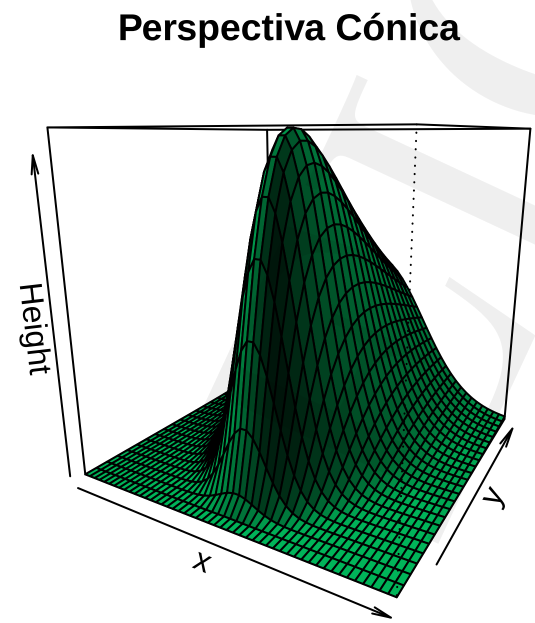
```
## Loading required package: rgl
```

```
## Loading required package: plot3D
```

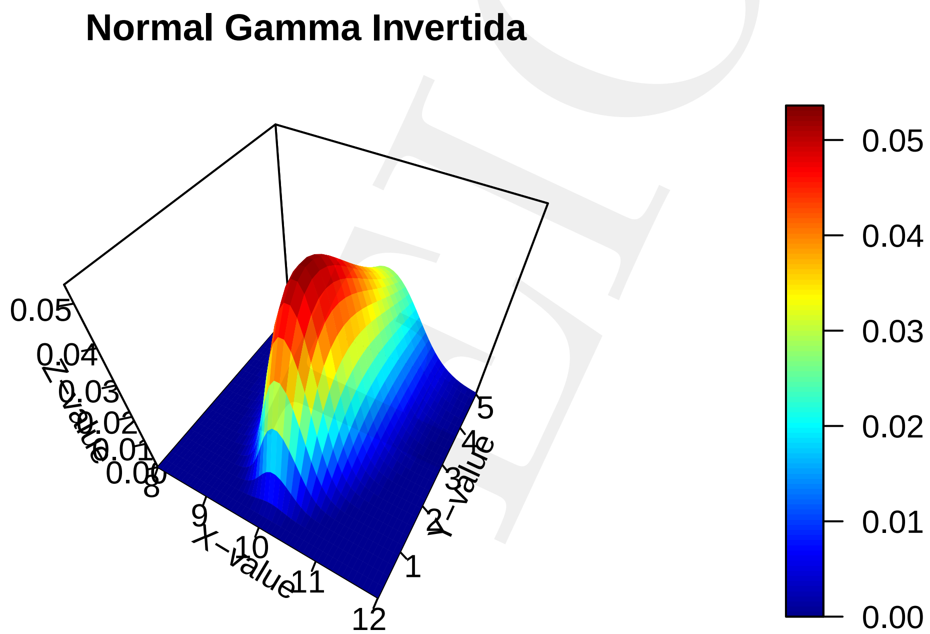
```
library(plot3D)
require(rgl)
```

```
m=10 # valor de la media
c= 1/10 # valor de la constante
a=1# primer parámetro de la gamma invertida
b=4 # segundo valor de la gamma invertida
x <- seq(m-2, m+2, length.out = 50)
y <- seq(0.11, 5, length.out = 25)
ngai <- function(x,y) {
 b^a/(gamma(a)) *(1/y^(a+1))*exp(-b/y)*(1/(sqrt(2*pi)))*exp(-(x-m)^2/(2*y*c))
  }

z <- outer(x, y, ngai)
persp(x, y, z,main="Perspectiva Cónica",
      zlab = "Height",
      theta = 30, phi = 15,
      col = "springgreen", shade = 0.5)
```



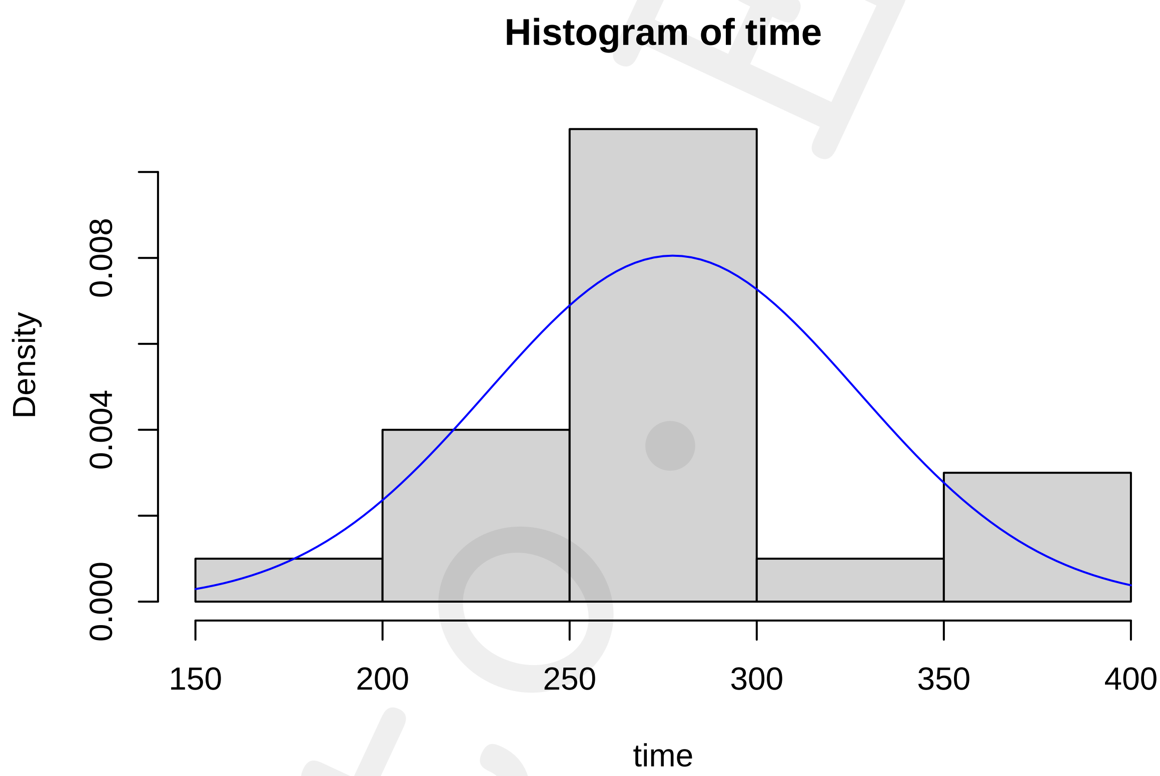

```
persp3D(x,y,z,theta=30, phi=50, axes=TRUE,scale=2, box=TRUE, nticks=5,
        ticktype="detailed", xlab="X-value", ylab="Y-value", zlab="Z-value",
        main="Normal Gamma Invertida")
```


Normal Gamma Invertida
Z-value
0.05
0.04
0.03
0.02
0.01
0.00
X-value
8
9
10
11
12
Y-value
1
2
3
4
5
0.05
0.04
0.03
0.02
0.01
0.00

Problema 1.

LA MARATHON DE NUEVA YORK

```
library(LearnBayes)
data("marathontimes")
View(marathontimes)
attach(marathontimes)
hist(time,freq=F)
curve(dnorm(x,mean=mean(time),sd=sd(time)),add=T,col="blue")
```




```
shapiro.test(time)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  time
## W = 0.97012, p-value = 0.7573
```

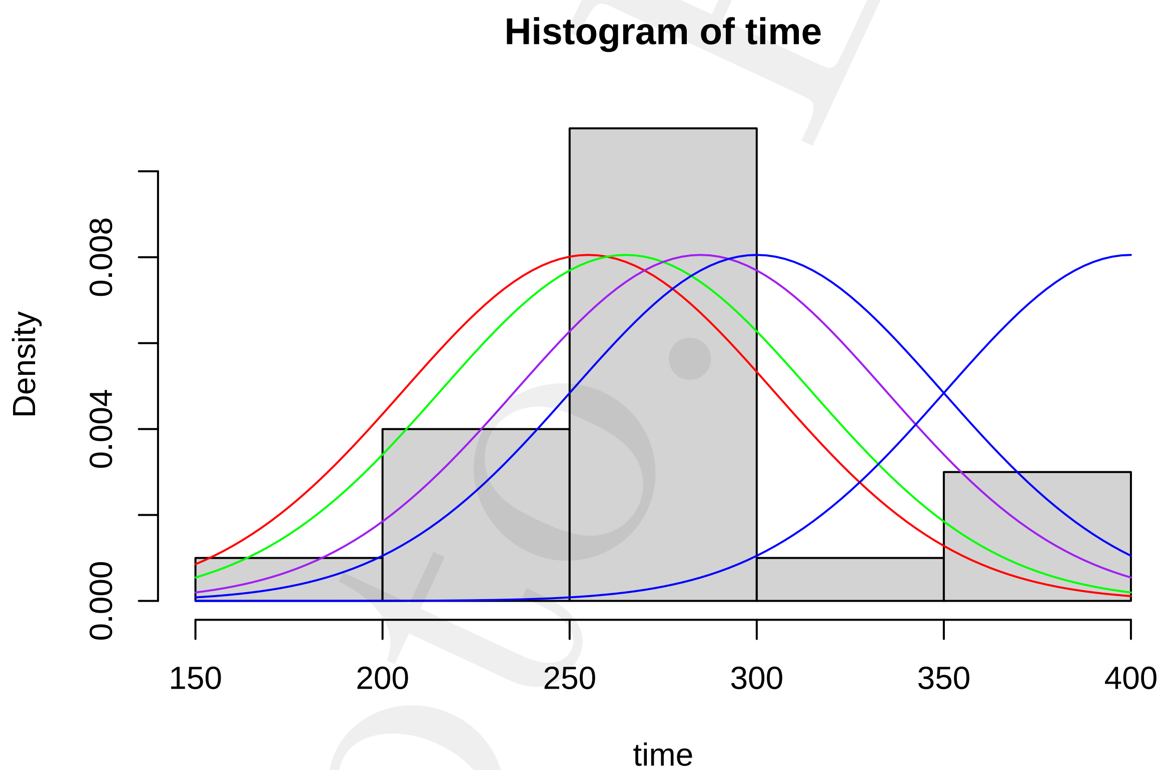
```
t.test(time)
```

```
##
##  One Sample t-test
##
## data:  time
## t = 25.061, df = 19, p-value = 5.094e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
```

```
##  254.4154 300.7846
## sample estimates:
## mean of x
##     277.6
```

Efecto gráfico sobre distintas curvas según posibles valores de las medias del IC.

```
hist(time,freq=F)
curve(dnorm(x,mean=255,sd=sd(time)),add=T,col="red")
curve(dnorm(x,mean=265,sd=sd(time)),add=T,col="green")
curve(dnorm(x,mean=285,sd=sd(time)),add=T,col="purple")
curve(dnorm(x,mean=300,sd=sd(time)),add=T,col="blue")
curve(dnorm(x,mean=400,sd=sd(time)),add=T,col="blue")
```



Hagamos lo mismo para la varianza

```
n=length(time)
uppert=(n-1)*sd(time)^2/qchisq(0.025,df=n-1);sqrt(uppert)
```

```
## [1] 72.35422
```

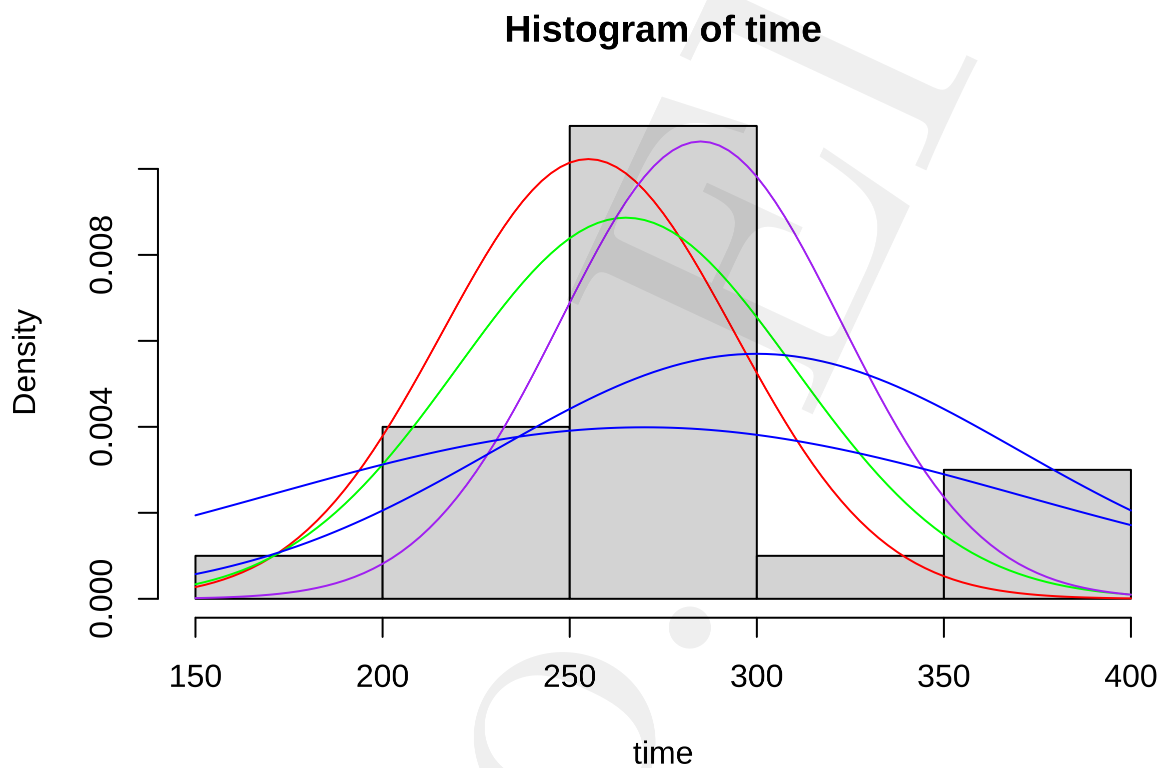
```
lowert=(n-1)*sd(time)^2/qchisq(0.975,df=n-1);sqrt(lowert)
```

```
## [1] 37.67339
```

Representemos curvas normales para diferentes desviaciones típicas.

```
hist(time,freq=F)
curve(dnorm(x,mean=255,sd=39),add=T,col="red")
curve(dnorm(x,mean=265,sd=45),add=T,col="green")
```

```
curve(dnorm(x,mean=285,sd=37.5),add=T,col="purple")
curve(dnorm(x,mean=300,sd=70),add=T,col="blue")
curve(dnorm(x,mean=270,sd=100),add=T,col="blue")
```



Un problema típico de inferencia consiste en estimar los parámetros desconocidos de una normal. Para ilustrar los métodos bayesianos, vamos a suponer que estamos interesados en estimar dichos parámetros en la prueba masculina de la maratón de Nueva York para edades comprendidas entre 20 y 29 años. Se observan los tiempos $\mathbf{y} = (y_1, \ldots, y_{20})$ para 20 corredores en minutos, y vamos a suponer que proceden de una población normal. Si también suponemos que no disponemos de información a priori lo sensato es proponer una distribución a priori de Jeffrey:

$$g(\mu, \sigma_2) := 1/\sigma_2^{3/2}$$

Entonces, la densidad a posteriori de la media y la varianza viene dada por

$$g(\mu, \sigma_2|\mathbf{y}) := (1/\sigma_2)^{\frac{n+3}{2}} \exp[-\frac{1}{2\sigma_2}(nS^2 + n(\mu - \bar{y})^2)]$$

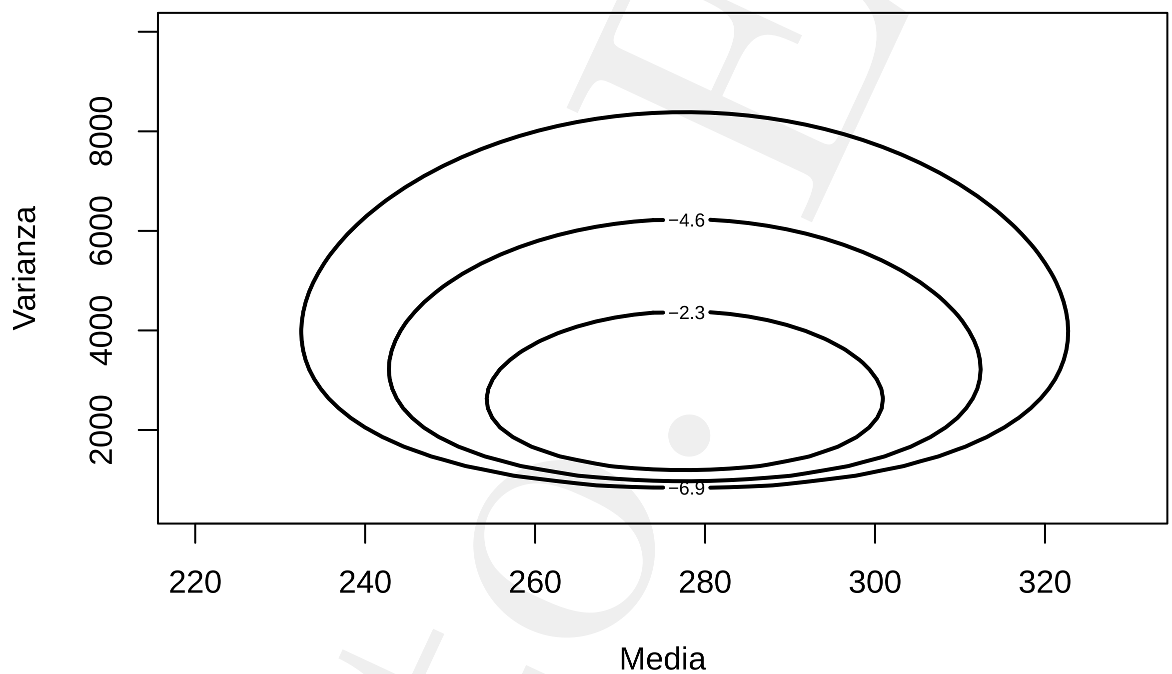
donde n es el tamaño muestral, $\bar{y}$ es la media muestral y S^2 la varianza muestral.

Es decir la a posteriori conjunta es una normal-gamma-invertida ya que

1. La a posteriori de μ condicionada a σ_2 es $N(\bar{y}, \frac{\sigma_2}{n})$.
2. La a posteriori de σ_2 es una gamma invertida cuyos parámetros son: $\frac{n}{2}$ y $ns^2/2$.

Primero vamos a usar `R` para construir un gráfico de contorno de la densidad a posteriori. La función en R `normchi2post.R` que está en `LearnBayes` calcula un algoritmo para el logaritmo de la densidad a posteriori de (μ, σ_2). También usaremos la función `mycontour.R` que facilita el uso del comando `contour`. El comando `mycontour` tiene tres parámetros: el nombre de la función que define la densidad log, un vector con valores que definen el rectángulo donde la densidad es representada y los datos que se usan en la función para la densidad log. La función produce un contorno donde las lineas del contorno son dibujadas al 10%,1% y 0.1% del valor máximo de la densidad a posteriori sobre la malla.

```
d=mycontour(normchi2post,c(220,330,500,10000),time)
title(xlab="Media",ylab="Varianza")
```



Es conveniente describir esta distribución a posteriori mediante técnicas de simulación. Para simular un valor de (μ, σ_2) de la a posteriori conjunta, primero simulamos un valor de σ_2 de la correspondiente gamma-invertida y luego simulamos un valor de μ de una normal con parámetros $(\bar{y}, \frac{\sigma_2}{n})$. Hay que tener en cuenta que la distribución a posteriori de σ_2 viene dada por

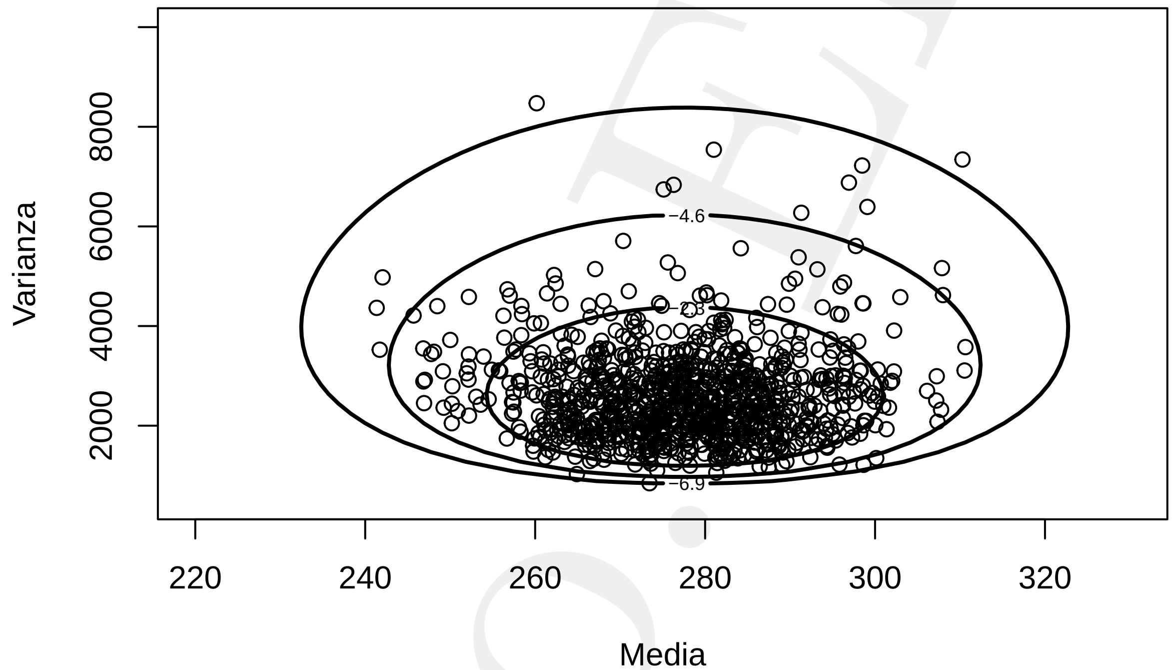
$$\frac{nS^2}{\sigma_2^2} \sim \chi^2_{n-1}.$$

En la siguiente salida de `R`, primero simulamos una muestra de tamaño 1000 de la chi-cuadrado usando la función `rchisq`. A partir de aquí se obtienen muestras de la distribución chi-cuadrada inversa escalada de la varianza σ_2 mediante la transformación de las muestras chi-cuadrado. Por último, se obtienen muestras de la media μ usando la función `rnorm`.

```
Su=sum((time-mean(time))^2)
n=length(time)
S=Su/(n) # Se usará más adelante
sigma2=(Su)/rchisq(1000,df=n)
mu=rnorm(1000,mean=mean(time),sd=sqrt(sigma2)/sqrt(n))
```

A continuación se muestran todos estos datos simulados en un contorno.

```
d=mycontour(normchi2post,c(220,330,500,10000),time)
title(xlab="Media",ylab="Varianza")
points(mu,sigma2)
```



Si quisiéramos obtener estimaciones de la desviación típica $\sqrt{\sigma_2}$ que describe la dispersión de los tiempos de la Maratón debemos obtener una muestra de la a posteriori de σ_2, le calculamos las raíces cuadradas y obtenemos la región creíble HPDI al 90%. En este caso tendríamos

```
library(HDInterval)
dens2 <- density(sigma2)
sqrt(hdi(dens2, credMass=0.90))
```

```
##    lower    upper
## 35.02789 62.04783
## attr(,"credMass")
## [1] 0.9
## attr(,"height")
## [1] 9.775413e-05
```

En el caso de la a priori que hemos supuesto si queremos obtener una región de credibilidad de la media tenemos que usar el resultado correspondiente a la densidad marginal a posteriori:

$$Z|\mathbf{x} \sim t_n, \text{ donde } Z = \sqrt{n}\frac{\theta - \bar{x}}{S}.$$

```
tsup=qt(0.975,n)
ss=mean(time)+tsup*sqrt(S/n)
```

```
ii=mean(time)-tsup*sqrt(S/n)
print(c(ii,ss))
```

```
## [1] 255.0786 300.1214
```