

Distribución Gamma:

Teoría

La distribución Gamma de una variable aleatoria continua X presenta dos parámetros α y β , y su función de densidad está dada por:

$$f(x) = \begin{cases} \frac{1}{\beta^\alpha \Gamma(\alpha)} x^{\alpha-1} e^{-\frac{x}{\beta}}, & x > 0, \alpha > 0, \beta > 0 \\ 0, & \text{Otro caso} \end{cases}$$

Donde $\Gamma(\alpha)$ representa la función gamma de alfa, la cual se define como:

$$\Gamma(\alpha) = \int_0^{+\infty} x^{\alpha-1} e^{-x} dx, \quad \alpha > 0$$

Gráfico en R

Para crear un gráfico de esta distribución se necesita instalar el paquete ggplot2. Se instala y se ocupa con la sintaxis:

```
install.packages('ggplot2')  
library(ggplot2)
```

Luego, se introducen los valores de los parámetros *alfa_gamma* y *beta_gamma*, en este caso se han definido sus valores como 2 y 1, respectivamente. Estos pueden ser modificados por el usuario introduciendo el nuevo valor en lugar de los dígitos establecidos.

```
alfa_gamma <- 2  
beta_gamma <- 1
```

Una vez se tienen los parámetros, se establece un dominio para la variable aleatoria continua X de la distribución. Por medio de una secuencia definimos el dominio de X, esta función necesita un límite inferior, un límite superior y una cantidad de particiones.

```
seq('limite_inferior', 'limite_superior', length.out = 'num_particiones')
```

En este caso se considera el límite superior de -1 y el superior de 7.5. Sin embargo, esta variable puede tomar distintos valores dependiendo de la situación.

```
x_gamma <- seq(-1, 7.5, length.out = 1000)
```

Luego se obtienen los valores de la probabilidad para cada X dentro del dominio; se utiliza la función *dgamma()*, en donde primero se colocan los valores de la variable aleatoria X a evaluar; luego, se establece el valor de *shape* que es al valor de alfa que se le pide al usuario; por último, se coloca el valor de *rate* que hace referencia al valor de beta.

```
y_gamma = dgamma(x_gamma, shape = alfa_gamma, rate = beta_gamma)
```

Ahora, para colocar estos valores en un gráfico se crea un *data.frame()*, en donde primero se colocan los valores de la variable aleatoria X, y luego los valores de las probabilidades para cada valor de X, que serían los datos obtenidos de la función *dgamma()* titulados como *y_gamma*.

```
data_gamma <- data.frame(x_gamma, y_gamma)
```

Para la creación del gráfico se colocan los datos en la función *ggplot()* y se agrega un tipo de gráfico, en este caso *geom_line()*. Para agregar los datos en *ggplot()* primero se coloca el *data.frame()* titulado como *data_gamma()* y con la función *aes()* se especifican que datos irán en cada uno de los ejes, primero se coloca el eje x y luego, el eje y.

```
ggplot(data_gamma, aes(x_gamma, y_gamma)) + geom_line()
```

Por último, se pueden agregar elementos para dar estilo al gráfico. En la sección de `geom_line()`, con la función `colour` se puede especificar un color para la curva que se produce y con `size`, el tamaño del grosor.

```
geom_line(colour = 'blue', size = 1)
```

Para agregar título al gráfico se realiza agregándole la función `ggtitle()`, se puede colocar un subtítulo seguido del título y separados por coma de la siguiente manera `ggtitle('titulo','subtitulo')`. Para el subtítulo se utiliza la función `paste()` para concatenar textos y caracteres, cada valor de los parámetros alfa y beta se colocan en la función `toString()` para convertirlos en datos de tipo String.

```
ggtitle("Función de densidad Gamma",paste("Para Alfa = ",toString(alfa_gamma),  
"y Beta = ",toString(beta_gamma)))
```

Se colocan los títulos de cada eje con la función `labs()`, indicando el nombre para el eje x y el y como se indica: `labs(x = "x", y = "Distribución")`. Por último, para agregarle color y estilo al texto del título se realiza desde la función `theme()`, seleccionamos el título con la sintaxis `plot.title` e indicamos que es al texto con `element_text()` y se indica el color y el estilo del texto con las funciones `color` y `face`, respectivamente.

```
theme(plot.title = element_text(color = "blue", face = "bold"))
```

Distribución Exponencial:

Teoría:

La distribución exponencial es un caso especial de la distribución Gamma en donde $\alpha = 1$ y la densidad de probabilidad está dada por:

$$f(x) = \begin{cases} \frac{1}{\beta} e^{-\frac{x}{\beta}}, & x > 0, \beta > 0 \\ 0, & \text{Otro caso} \end{cases}$$

Gráfico en R

Para crear un gráfico de esta distribución se necesita igualmente instalar el paquete `ggplot2`. Se instala y se ocupa con la sintaxis:

```
install.packages('ggplot2')  
library(ggplot2)
```

Luego, se introduce el valor del parámetro `beta_exponencial`, en este caso se ha definido con un valor de 2, pero este puede cambiarse sustituyendo el dígito en el código:

```
beta_exponencial <- 2
```

Se debe definir el recíproco de beta exponencial, pues la función que se utilizará más adelante para sacar los valores de la distribución exponencial define `rate` como el recíproco.

```
beta_exponencial_reciproco <- 1/beta_exponencial
```

Una vez se tiene el parámetro, se establece un dominio para la variable aleatoria continua X de la distribución. Por medio de una secuencia definimos el dominio de X, esta función necesita un límite inferior, un límite superior y una cantidad de particiones.

```
seq('limite_inferior','limite_superior',length.out = 'num_particiones')
```

En este caso se considera el límite superior de -1 y el superior de 7.5. Sin embargo, esta variable puede tomar distintos valores dependiendo de la situación.

```
x_exponencial <- seq(-1,7.5,length.out = 1000)
```

Luego se obtienen los valores de la probabilidad para cada X dentro del dominio; se utiliza la función *dexp()*, en donde primero se colocan los valores de la variable aleatoria X a evaluar y luego, el valor de *rate* que hace referencia al valor del recíproco de beta.

```
y_exponencial = dexp(x_gamma,rate = beta_exponencial_reciproco)
```

Ahora, para colocar estos valores en un gráfico se crea un *data.frame()*, en donde primero se colocan los valores de la variable aleatoria X, y luego los valores de las probabilidades para cada valor de X, que serían los datos obtenidos de la función *dexp()* titulados como *y_exponencial*.

```
data_exponencial <- data.frame(x_exponencial,y_exponencial)
```

Para la creación del gráfico se colocan los datos en la función *ggplot()* y se agrega un tipo de gráfico, en este caso *geom_line()*. Para agregar los datos en *ggplot()* primero se coloca el *data.frame()* titulado como *data_exponencial()* y con la función *aes()* se especifican que datos irán en cada uno de los ejes, primero se coloca el eje x y luego, el eje y.

```
ggplot(data_exponencial,aes(x_exponencial,y_exponencial)) + geom_line()
```

Por último, se pueden agregar elementos para dar estilo al gráfico. En la sección de *geom_line()*, con la función *colour* se puede especificar un color para la curva que se produce y con *size*, el tamaño del grosor.

```
geom_line(colour = 'blue',size = 1)
```

Para agregar título al gráfico se realiza agregándole la función *ggtitle()*, se puede colocar un subtítulo seguido del título y separados por coma de la siguiente manera *ggtitle('titulo','subtitulo')*. Para el subtítulo se utiliza la función *paste()* para concatenar textos y caracteres, el valor del parámetro beta se coloca en la función *toString()* para convertirlo en un dato de tipo String.

```
ggtitle("Función de densidad Exponencial",paste("Para Beta = ",toString(beta_exponencial)))
```

Se colocan los títulos de cada eje con la función *labs()*, indicando el nombre para el eje x y el y como se indica: *labs(x = "x",y = "Distribución")*. Por último, para agregarle color y estilo al texto del título se realiza desde la función *theme()*, seleccionamos el título con la sintaxis *plot.title* e indicamos que es al texto con *element_text()* y se indica el color y el estilo del texto con las funciones *color* y *face*, respectivamente.

```
theme(plot.title = element_text(color = "blue",face = "bold"))
```