PruebaDistribución

Daniel Gonzalez

2025-02-21

library(ggplot2)  
library(fitdistrplus)

## Cargando paquete requerido: MASS

## Cargando paquete requerido: survival

knitr::opts\_chunk$set(echo = TRUE)

# Distribuciones de Probabilidad y Prueba de Bonda de Ajuste

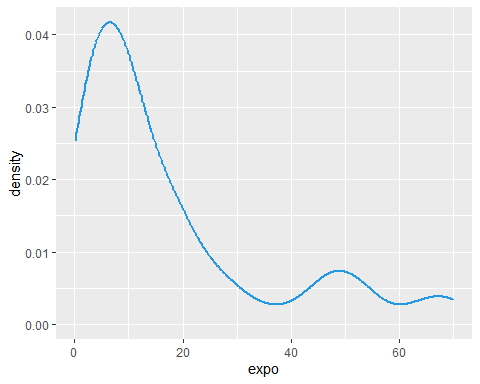
## Distribución Exponencial

set.seed(1073173761)  
  
expo=round(rexp(100,0.0625),2) #Distribución Exponencial  
print(expo)

## [1] 21.58 20.12 4.67 5.44 7.90 27.77 4.98 35.11 49.99 5.99 6.30 19.25  
## [13] 25.87 5.15 19.73 19.59 18.46 20.38 5.13 3.93 1.45 15.22 47.39 0.50  
## [25] 70.00 9.32 12.38 2.48 16.72 8.69 8.02 10.92 18.48 0.83 9.26 8.05  
## [37] 10.71 0.16 17.30 10.40 14.11 12.56 8.53 3.28 6.44 7.23 13.59 7.35  
## [49] 40.52 13.12 12.53 63.68 11.55 0.81 20.61 12.97 9.91 46.79 3.28 15.71  
## [61] 0.93 8.37 7.79 3.71 3.24 7.72 5.41 17.86 4.49 50.31 26.71 1.68  
## [73] 7.55 52.33 0.30 44.51 69.89 28.11 0.25 1.53 5.07 54.65 7.00 63.30  
## [85] 54.37 32.04 10.06 48.97 25.83 11.86 67.90 9.44 2.15 0.14 4.02 45.14  
## [97] 2.55 13.57 9.25 2.80

## **Gráfica de Densidad de la distribución Exponencial**

df = data.frame(expo)  
ggplot(df,aes(x=expo))+geom\_density(color=4,lwd=1,linetype=1)



## Prueba de Ajuste para la distribución Exponencial

fit1=fitdist(expo,"exp")  
ks.test(expo,"pexp",fit1$stimate)

## Warning in ks.test.default(expo, "pexp", fit1$stimate): ties should not be  
## present for the one-sample Kolmogorov-Smirnov test

## Warning in max(c(x, 1/n - x)): ningun argumento finito para max; retornando  
## -Inf

##   
## Asymptotic one-sample Kolmogorov-Smirnov test  
##   
## data: expo  
## D = -Inf, p-value = 1  
## alternative hypothesis: two-sided

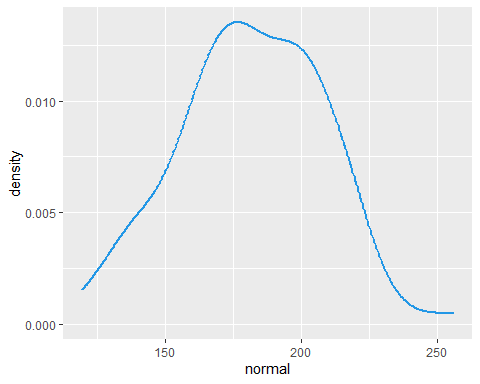
## Distribución Normal

normal= round(rnorm(100,179.1,28.2),2) #Distribución Normal  
print(normal)

## [1] 179.25 197.35 201.32 214.84 218.50 163.22 203.27 148.93 213.71 188.98  
## [11] 218.58 172.60 153.28 202.79 180.23 235.85 199.75 172.14 183.33 164.02  
## [21] 183.29 179.14 256.10 184.80 200.73 172.12 202.80 202.17 175.31 186.19  
## [31] 158.76 179.93 154.44 186.34 134.31 131.44 152.01 158.81 139.47 212.94  
## [41] 188.11 157.89 208.05 119.53 222.24 167.04 170.15 192.23 196.04 168.29  
## [51] 213.05 138.84 216.45 197.20 224.25 201.52 207.17 217.81 174.60 154.94  
## [61] 203.79 186.70 133.33 172.49 197.81 204.15 169.28 179.28 193.30 145.30  
## [71] 219.49 154.75 181.45 170.77 201.64 213.65 179.65 157.65 203.77 197.85  
## [81] 172.14 139.55 169.34 196.07 184.77 176.44 180.73 192.52 159.69 187.22  
## [91] 121.90 213.44 165.46 194.10 140.18 139.70 166.21 167.46 166.73 167.55

## **Gráfica de la Distribución Normal**

df = data.frame(normal)  
ggplot(df,aes(x=normal))+geom\_density(color=4,lwd=1,linetype=1)



## Prueba de Shapiro para la distribución Normal

shapiro.test(normal)

##   
## Shapiro-Wilk normality test  
##   
## data: normal  
## W = 0.9906, p-value = 0.7132

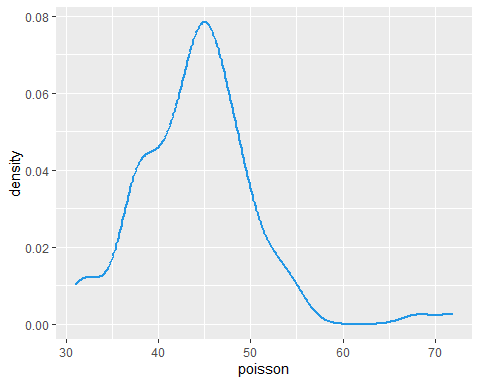
## **Distribución de Poisson**

poisson =round(rpois(100,43.2) ,2) # Distribución Poisson  
print(poisson)

## [1] 45 43 45 43 37 51 48 39 45 46 42 49 47 38 55 44 41 45 41 46 44 46 49 46 38  
## [26] 42 49 37 44 32 47 39 49 43 46 52 46 45 48 44 37 54 44 47 55 46 72 36 38 48  
## [51] 31 38 39 41 44 47 50 46 48 41 43 38 40 40 49 46 44 38 41 41 45 68 43 34 45  
## [76] 42 48 37 32 37 42 44 53 31 45 41 38 51 38 33 49 53 50 47 45 45 48 44 52 43

## Gráfica de la distribución de Poisson

df = data.frame(poisson)  
ggplot(df,aes(x=poisson))+geom\_density(color=4,lwd=1,linetype=1)



## Prueba de Bondad de ajuste de la distribución Poisson

fit1=fitdist(poisson,"pois",method="mle")  
summary(fit1)

## Fitting of the distribution ' pois ' by maximum likelihood   
## Parameters :   
## estimate Std. Error  
## lambda 44.21 0.664906  
## Loglikelihood: -326.3517 AIC: 654.7034 BIC: 657.3086

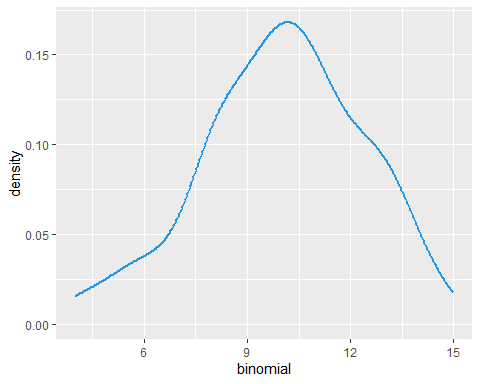
## **Distribución Binomial**

binomial=round(rbinom(100,20,0.5) ,2) #Distibución binomial.  
print(binomial)

## [1] 13 13 11 4 14 8 10 10 11 10 11 8 9 11 8 10 9 14 11 12 12 11 6 9 14  
## [26] 8 8 13 13 9 8 10 11 9 10 15 12 10 11 6 8 10 9 8 9 10 8 9 8 8  
## [51] 13 13 13 11 9 13 10 11 6 11 12 13 14 11 10 5 12 8 5 13 10 8 11 10 10  
## [76] 13 10 11 7 10 6 11 9 4 9 8 13 12 10 12 7 10 9 8 12 11 12 10 6 10

## Gráfica de la distribución Binomial

df = data.frame(binomial)  
ggplot(df,aes(x=binomial))+geom\_density(color=4,lwd=1,linetype=1)



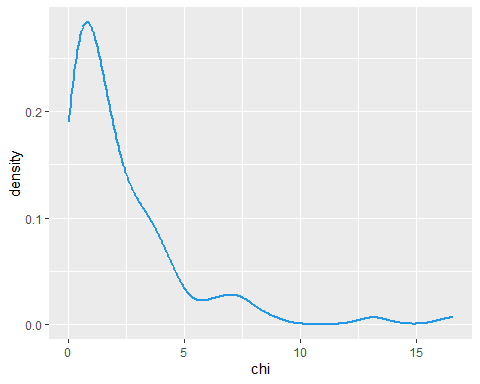
## Distribución Chi Cuadrado

chi=round(rchisq(100,2) ,2) #Distribución Chi Cuadrado.  
print(chi)

## [1] 1.79 1.99 7.54 0.03 2.02 1.29 0.23 2.32 0.43 1.31 1.27 0.29  
## [13] 0.74 0.48 0.67 3.64 0.51 3.01 2.94 6.81 2.03 0.28 6.27 0.98  
## [25] 1.68 1.75 0.12 2.55 0.32 0.68 3.68 0.07 3.09 7.47 0.19 4.40  
## [37] 0.95 2.71 0.62 0.78 3.58 0.96 3.36 0.89 4.61 1.67 1.38 0.34  
## [49] 0.32 0.36 4.22 1.78 0.19 1.70 0.97 7.27 8.58 1.66 3.45 1.68  
## [61] 0.11 2.36 3.93 4.49 1.17 3.06 0.45 0.23 1.31 2.82 0.43 16.61  
## [73] 0.90 0.53 2.33 0.43 0.00 4.59 3.76 1.47 1.48 0.40 0.70 3.58  
## [85] 0.84 1.05 0.06 2.16 0.75 2.55 0.86 0.87 13.16 1.98 3.06 5.54  
## [97] 1.08 6.19 1.67 0.87

## Gráfica de la distribución Chi Cuadrado

df = data.frame(chi)  
ggplot(df,aes(x=chi))+geom\_density(color=4,lwd=1,linetype=1)



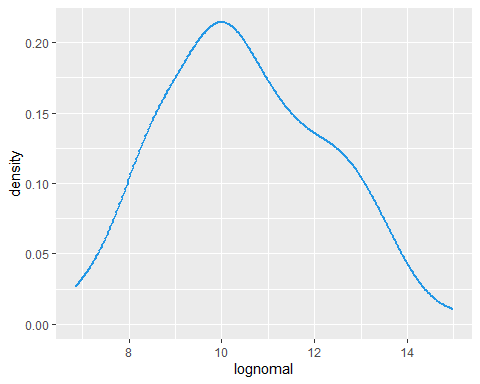
## Distribución Log Normal

lognomal=round(rlnorm(100,2.32,0.2),2)  
print(lognomal)

## [1] 10.06 9.33 15.00 13.40 13.16 12.49 9.58 11.02 8.43 7.19 11.94 7.74  
## [13] 9.88 11.32 12.24 12.73 10.22 13.63 12.33 11.02 8.26 10.27 12.04 11.77  
## [25] 12.68 11.12 9.54 9.30 12.24 7.16 9.45 10.07 11.26 8.49 10.75 9.63  
## [37] 9.16 9.00 10.25 7.84 9.97 11.26 12.53 13.95 9.90 9.86 8.56 8.57  
## [49] 13.15 9.81 11.40 11.37 9.48 8.43 8.14 10.98 8.64 11.22 9.30 10.49  
## [61] 12.79 10.82 8.69 12.33 8.24 9.26 12.00 10.40 13.71 9.43 13.00 10.60  
## [73] 8.44 12.54 11.91 9.43 8.93 11.30 10.18 11.68 12.74 13.56 10.40 10.88  
## [85] 8.64 11.55 8.16 13.20 10.19 10.44 9.85 8.04 10.24 10.37 10.15 8.75  
## [97] 10.37 9.81 9.28 6.84

## Gráfica de la Distribución Log Normal

df = data.frame(lognomal)  
ggplot(df,aes(x=lognomal))+geom\_density(color=4,lwd=1,linetype=1)



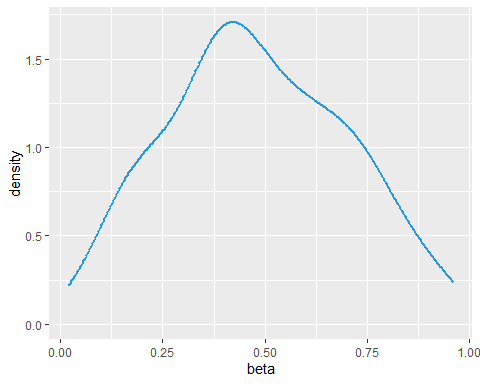
Distribución Beta

beta =round(rbeta(100,2,2),2)  
print(beta)

## [1] 0.37 0.37 0.96 0.40 0.77 0.66 0.33 0.37 0.58 0.75 0.21 0.35 0.41 0.70 0.54  
## [16] 0.60 0.48 0.49 0.42 0.19 0.61 0.39 0.59 0.28 0.48 0.51 0.18 0.74 0.71 0.83  
## [31] 0.25 0.19 0.64 0.57 0.51 0.60 0.18 0.11 0.11 0.45 0.37 0.82 0.71 0.02 0.76  
## [46] 0.67 0.34 0.36 0.24 0.82 0.24 0.47 0.90 0.40 0.62 0.52 0.13 0.74 0.45 0.41  
## [61] 0.65 0.10 0.17 0.74 0.40 0.92 0.44 0.48 0.32 0.67 0.45 0.60 0.24 0.51 0.53  
## [76] 0.39 0.77 0.28 0.63 0.12 0.53 0.33 0.40 0.46 0.49 0.19 0.66 0.56 0.34 0.60  
## [91] 0.42 0.20 0.23 0.90 0.72 0.41 0.33 0.75 0.85 0.38

## Gráfica de la distribución Beta

df = data.frame(beta)  
ggplot(df,aes(x=beta))+geom\_density(color=4,lwd=1,linetype=1)



## Distribución Gamma

gamma = round(rgamma(100,6,2),2)  
print(gamma)

## [1] 1.20 1.87 3.15 2.66 2.26 2.17 2.83 3.74 3.47 2.47 2.22 3.23 1.75 3.57 3.57  
## [16] 2.94 2.87 1.37 4.24 4.11 5.03 4.05 1.91 3.46 3.83 3.60 1.74 3.86 4.18 2.34  
## [31] 1.31 2.61 4.27 4.26 2.74 4.79 1.60 2.47 2.94 2.57 2.71 2.45 1.10 2.83 2.64  
## [46] 2.60 2.59 1.82 1.92 4.26 3.71 2.33 3.76 1.60 3.42 2.11 4.76 3.45 1.67 1.87  
## [61] 3.09 3.03 3.90 4.01 5.31 2.91 2.10 3.09 3.36 3.06 4.63 3.70 4.26 1.91 3.54  
## [76] 4.54 2.90 2.28 3.08 1.79 2.52 1.94 2.90 1.73 2.93 2.20 5.34 1.59 2.23 2.56  
## [91] 4.72 1.53 4.30 2.90 2.72 2.75 2.73 2.24 3.08 3.79

## Gráfica de la distribución Gamma

df = data.frame(gamma)  
ggplot(df,aes(x=gamma))+geom\_density(color=4,lwd=1,linetype=1)

