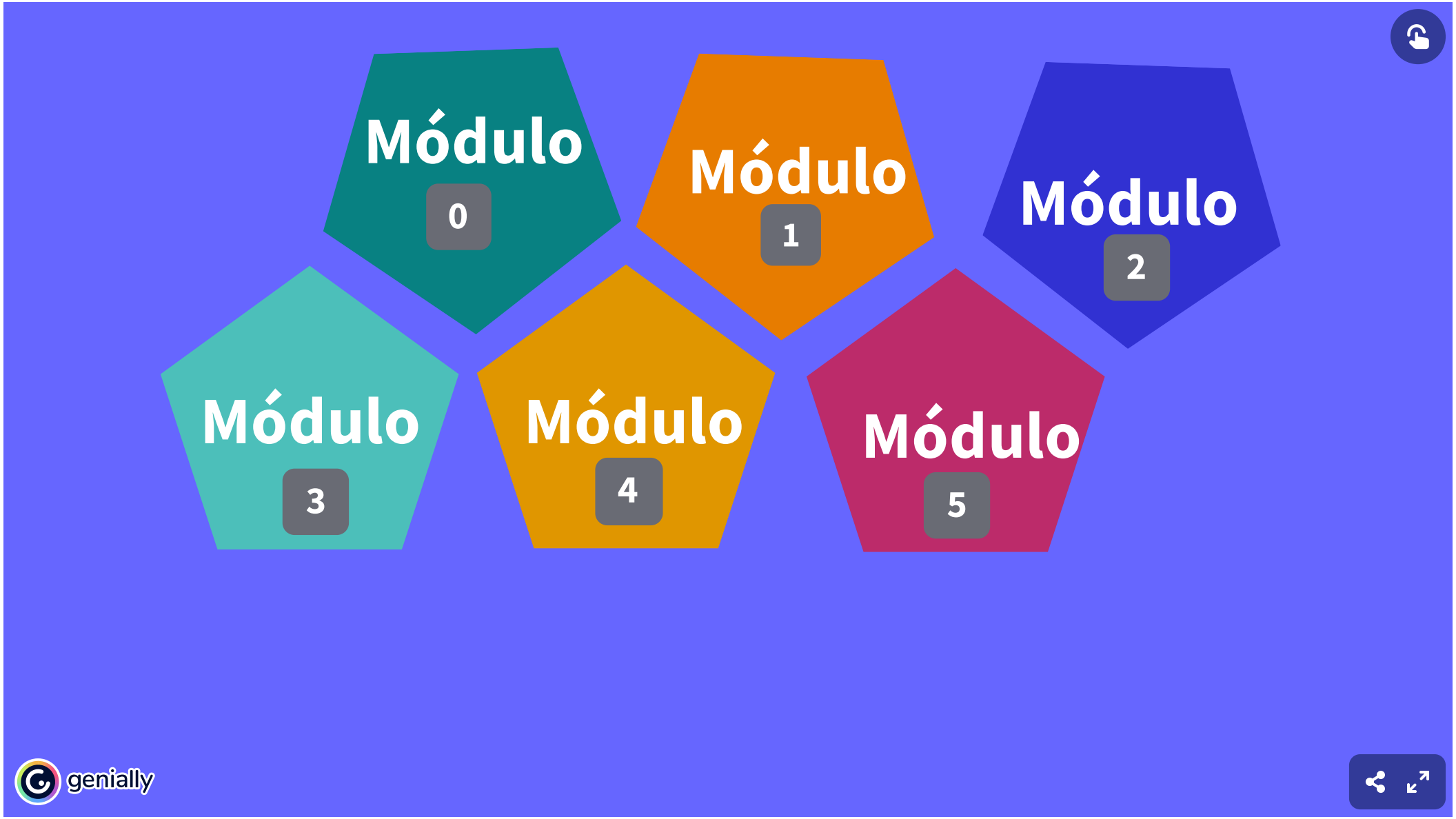
Introducción a R

Aplicaciones a la enseñanza de la Estadística

IV - Encuentro Colombiano de Educación Estocástica

Daniel Enrique González Gómez

2021-06-01


Módulo
0
Módulo
1
Módulo
2
Módulo
3
Módulo
4
Módulo
5
genially

Sumatoria

```
x=1:10
sum(x)
```

```
## [1] 55
```

```
x=c(1,2,3,4,5,6,7,8,9,NA)
sum(x)
```

```
## [1] NA
```

```
x=c(1,2,3,4,5,6,7,8,9,NA)
sum(x, na.rm = TRUE)
```

```
## [1] 45
```

Sumatoria

```
# sumatoria acumulada
x=1:10
cumsum(x)
```

```
##  [1]  1  3  6 10 15 21 28 36 45 55
```

$$\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}$$

```
x=1:10
sum((x-mean(x))^2)/(length(x)-1)
```

```
## [1] 9.166667
```

```
var(x)
```

```
## [1] 9.166667
```

Permutaciones - combinaciones

```r
# Permutacion
P=function(n,k){choose(n,k)*factorial(k)}
P(4,2)
```

```
## [1] 12
```

```r
# Combinacion
C=function(n,k){choose(n,k)}
C(4,2)
```

```
## [1] 6
```

Permutaciones

```
library(gtools)
N=4  # Número de elementos
n=2  # grupos de 2 en 2
id=c(1:N)
permutations(N, n, id)
```

```
##       [,1] [,2]
##  [1,]    1    2
##  [2,]    1    3
##  [3,]    1    4
##  [4,]    2    1
##  [5,]    2    3
##  [6,]    2    4
##  [7,]    3    1
##  [8,]    3    2
##  [9,]    3    4
## [10,]    4    1
## [11,]    4    2
## [12,]    4    3
```

```
cat("Total grupos : ", P(4,2))
```

Combinaciones

```
library(gtools)
N=5  # Número de elementos
n=2  # grupos de 2 en 2
id=c(1:N)
combinations(N, n, id)
```

```
##       [,1] [,2]
##  [1,]    1    2
##  [2,]    1    3
##  [3,]    1    4
##  [4,]    1    5
##  [5,]    2    3
##  [6,]    2    4
##  [7,]    2    5
##  [8,]    3    4
##  [9,]    3    5
## [10,]    4    5
```

```
cat("Total grupos : ", C(5,2))
```

```
## Total grupos :  10
```

Permutaciones en urna

```r
library(gtools)
# urna con 3 bolas
x <- c('Rojo', 'Azul', 'Verde')
permutations(n=3,r=2,v=x,repeats.allowed=TRUE)
```

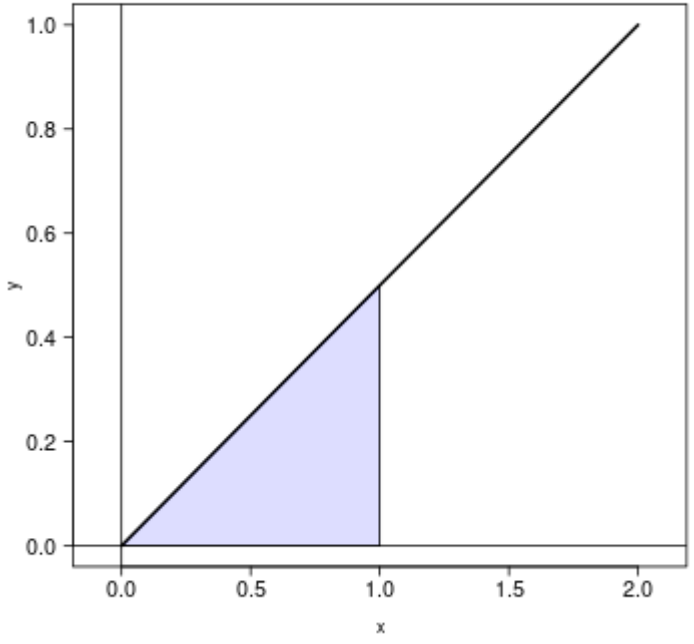
```
##       [,1]    [,2]
##  [1,] "Azul"  "Azul"
##  [2,] "Azul"  "Rojo"
##  [3,] "Azul"  "Verde"
##  [4,] "Rojo"  "Azul"
##  [5,] "Rojo"  "Rojo"
##  [6,] "Rojo"  "Verde"
##  [7,] "Verde" "Azul"
##  [8,] "Verde" "Rojo"
##  [9,] "Verde" "Verde"
```

Gráficos de funciones 2D

$$f_X(x) = \begin{cases} \dfrac{x}{2} & , \quad 0 \leq x \leq 2 \\ 0 & , \quad \text{en cualquier otro caso} \end{cases}$$

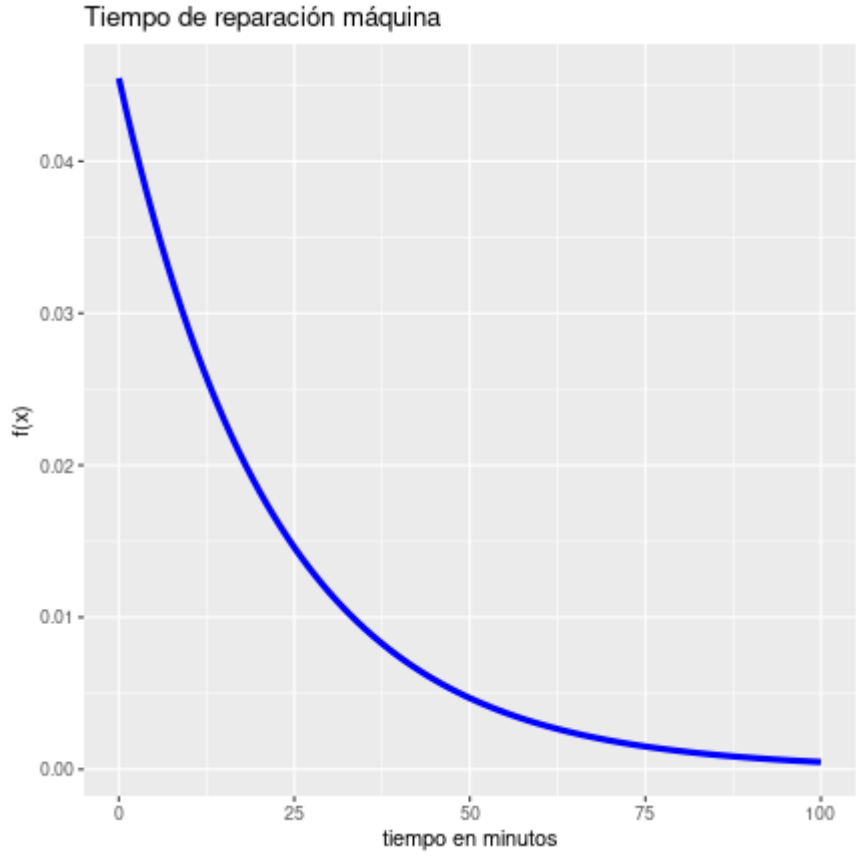
```
par(cex=0.6, cex.axis=1.2, cex.lab=1.2, cex.
f=function(x){x/2}
x1=c(-0.1,2.1)
x2=c(0,1)
plot(x2~x1,type="p", xlab="x", ylab="y", col
grid()

curve(f,0,2, add=TRUE, lwd=2)
t=seq(0,1,by=0.01)
x=c(0,t,1)
y=c(0,f(t),0)
polygon(x,y,
        col="#0000ff22")
abline(h=0,v=0)
```



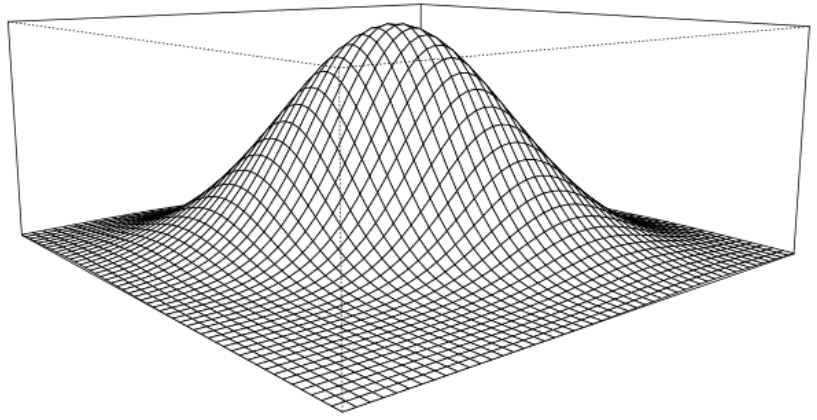
Gráficos 2D

```
library(ggplot2)
f=function(x){1/22*exp(-x/22)}
p9=ggplot(data.frame(x=c(0,100)),aes(x=x)) +
stat_function(fun=f,color="blue",size=1.5) +
ggtitle("Tiempo de reparación máquina")+
scale_x_continuous(name="tiempo en minutos"
scale_y_continuous(name="f(x)")
p9
```



Gráficos de funciones 3D

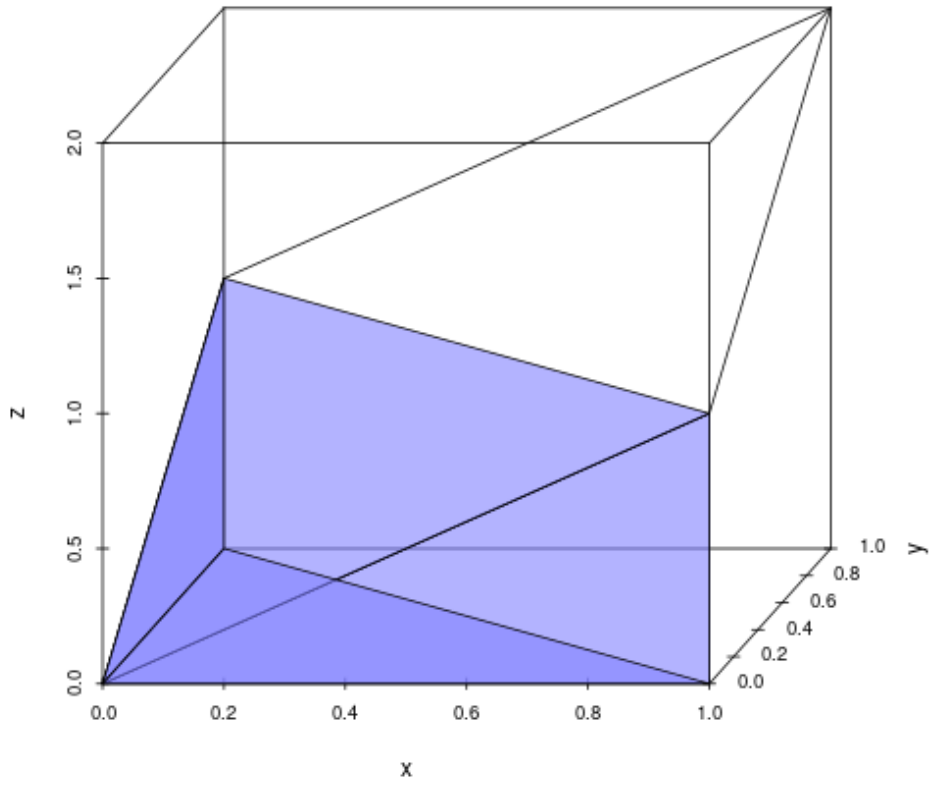
```
library("mvtnorm")
N=50
x <- seq(-3,3, length=N)
y <- seq(-3,3,length=N)
z <- matrix(0, N, N)
for (i in 1:N) for (j in 1:N) {
  z[i,j]=dmvnorm(c(x[i],y[j]), c(0,0),
        matrix(c(1,0.5,0.5,1),2,2))}
persp(x,y,z,theta=50, phi=10,
      xlab=" ",
      ylab=" ",
      zlab=" ",
      scale=TRUE,
      expand=.4,
      axes=FALSE)
```



[*] Cual es la función?

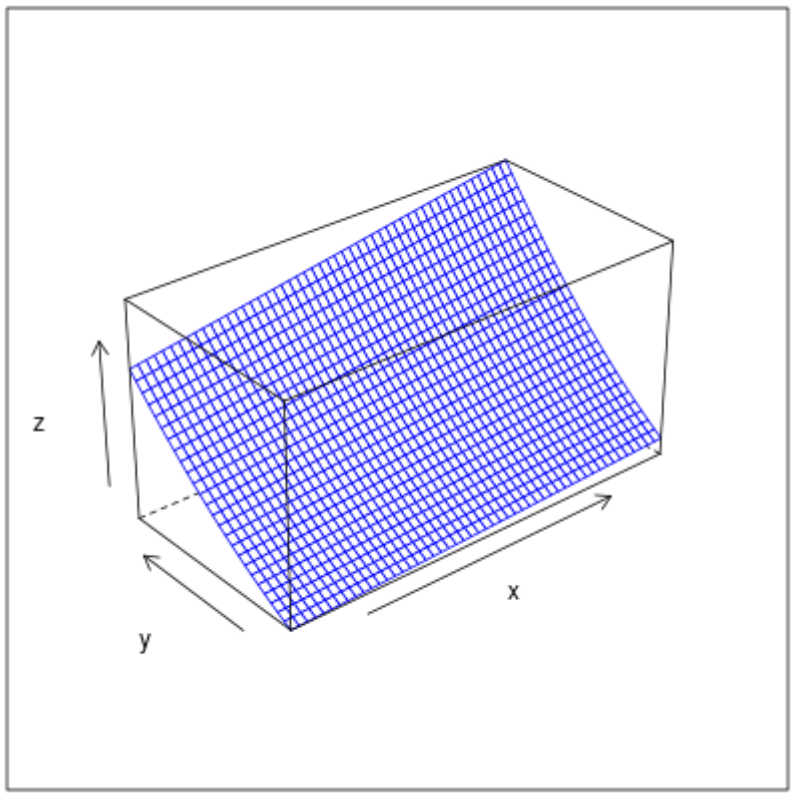
Gráficos de funciones 3D

```
library(scatterplot3d)
x=c(0,1,1,0,0)
y=c(0,0,1,1,0)
z=c(0,1,2,1,0)
s=scatterplot3d(x,y,z, type='l',xlim=c(0,1),
x0=c(0,1,1)
y0=c(0,0,0)
z0=c(0,0,1)
polygon(s$xyz.convert(x0,y0,z0),col="#8080FF
x1=c(0,1,0)
y1=c(0,0,1)
z1=c(0,1,1)
polygon(s$xyz.convert(x1,y1,z1),col="#8080FF
x2=c(0,0,0)
y2=c(0,1,1)
z2=c(0,0,1)
polygon(s$xyz.convert(x2,y2,z2),col="#8080FF
x3=c(0,1,0)
y3=c(0,0,1)
z3=c(0,0,0)
polygon(s$xyz.convert(x3,y3,z3),col="#8080FF
```



Gráficos de funciones 3D

```
library(lattice)
x=seq(3,4,by=0.02)
y=seq(0.5,1,by=0.02)
fun=function(x,y){48*x*y^2/49}
z=outer(x,y,fun)
wireframe(z,xlab="x",ylab="y",col="blue")
```

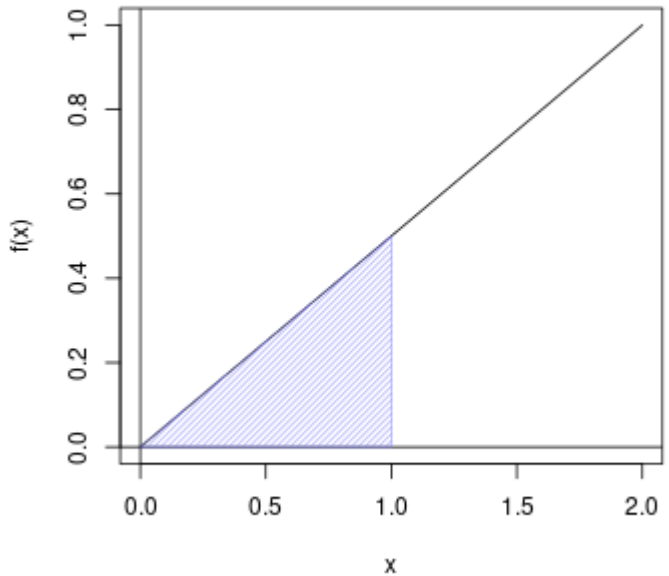


[*] Cual es la función?

Integración

$$\int_0^1 \frac{x}{2}\,dx = \frac{x^2}{4} = \frac{1}{4}.$$

```
f=function(x){x/2}
curve(f,0,2) # dibuja linea de la función
abline(h=0,v=0) # traza eje x y eje y
t=seq(0,1,by=0.01)
x=c(0,t,1)
y=c(0,f(t),0)
polygon(x,y,density=30, col="#8080FF99") # p
p=integrate(f,0,1)
p$value # resultado
```




```
## [1] 0.25
```

Integración- triples

$$\int_0^{1/2}\int_0^{1/2}\int_0^{1/2}\frac{2}{3}(x_1+x_2+x_3)dx_1\,dx_2\,dx_3$$

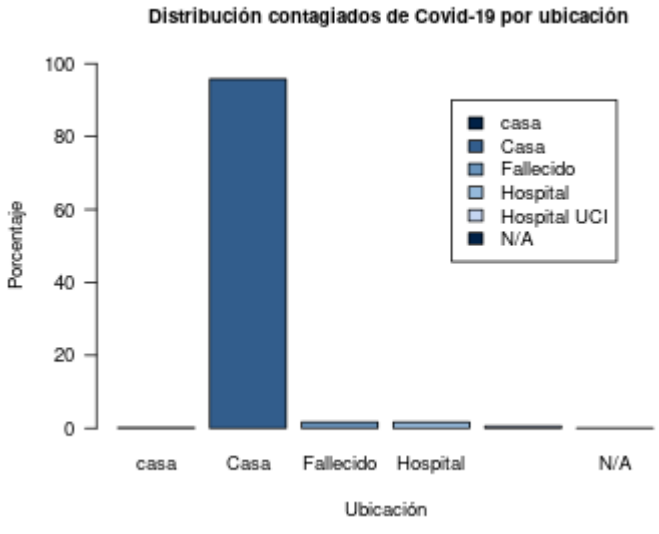
```
library(cubature)
f=function(x){2/3*(x[1]+x[2]+x[3])}
adaptIntegrate(f,lowerLimit=c(0,0,0),
               upperLimit=c(0.5,0.5,0.5))
```

```
## $integral
## [1] 0.0625
##
## $error
## [1] 1.387779e-17
##
## $functionEvaluations
## [1] 33
##
## $returnCode
## [1] 0
```


```
library(RSocrata)
library(stringr)
token ="ew2rEMuESuzWPqMkyPfOSGJgE"
Colombia= read.socrata("https://www.datos.go
Colombia$sexo=str_to_lower(Colombia$sexo)
Colombia$estado[Colombia$estado=="N/A"]="NA"
Colombia$estado=str_to_lower(Colombia$estado
Colombia$recuperado[Colombia$recuperado=="N/
Colombia$recuperado=str_to_lower(Colombia$re
Colombia$fuente_tipo_contagio[Colombia$fuent
Colombia$fuente_tipo_contagio=str_to_lower(C
Colombia$ubicacion[Colombia$ubicacion=="N/A"
Colombia$ubicacion=str_to_lower(Colombia$ubi
 saveRDS(Colombia,"data/Colombia202105.RDS")
```

```
Colombia=readRDS("data/Colombia202105.RDS")
```

```
par(cex=0.8, cex.axis=1, cex.lab=1, cex.main
t=table(Colombia$ubicacion) # table en frecu
t=prop.table(t)*100         # tabla en porce
t=round(t,2)                # tabla en porce
labs=names(t)               # nombres de las
barplot(t,main=" Distribución contagiados de
legend("topright", inset = 0.1,labs,fill =c(
```




```
t=prop.table(table(Colombia$ubicacion))
```

Primer dia de clases

```
library(RColorBrewer) # paquete  colores
library(readxl) # paquete leer archivos exce
bd0052 <- read_excel("bd0052.xlsx",sheet = "
bd0052$carrera[bd0052$carrera=="Biologia"]="
bd0052$carrera[bd0052$carrera=="Ingenieria C
```

```
attach(bd0052)
t1011=table(carrera,grupo)
knitr::kable(t1011)
barplot(t1011,col = brewer.pal(6,"Set1"))
```

```
p1012<-ggplot(bd005NA, aes(x=grupo, y=promac
  geom_jitter(color="black", size=0.4, alpha
p1012
```

Problema de los dados

```
sample(1:6,1)
```

```
## [1] 6
```

```
sample(1:6, 10, replace = TRUE)
```

```
##  [1] 3 4 6 6 5 1 3 3 5 2
```

```
dd=sample(1:6, 20, replace = TRUE)
mdd=matrix(dd,ncol = 2)
apply(mdd, 1,sum)
```

```
##  [1]  6  6  5  8  5 11 11  5  9 10
```

```
#  ?sample
sample(x,size,replace=FALSE,prob=NULL)
```

[*] tomado de: https://weloversize.com/compras-por-menos-de-10e-para-mejorar-tus-polvazos/

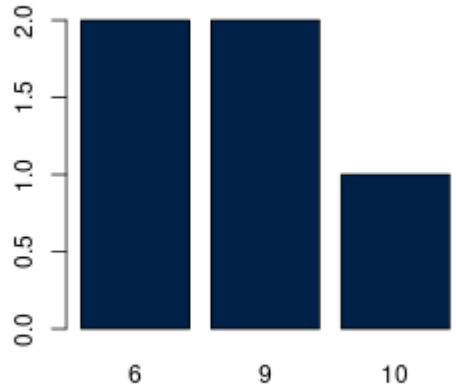
Problema de los dados

n

```
n=5
dd=sample(1:6, n*2, replace = TRUE)
mdd=matrix(dd,ncol = 2)
mdd
```

```
##      [,1] [,2]
## [1,]    3    3
## [2,]    6    4
## [3,]    3    6
## [4,]    4    5
## [5,]    3    3
```

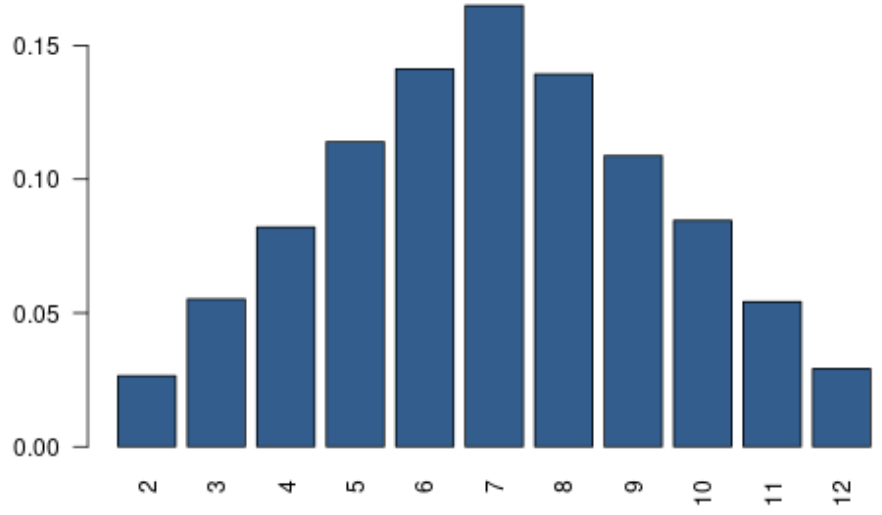
```
n=5
dd=sample(1:6, n*2, replace = TRUE)
mdd=matrix(dd,ncol = 2)
sdd=apply(mdd, 1,sum)
barplot(table(sdd), las=1)
prop.table(table(sdd))
```



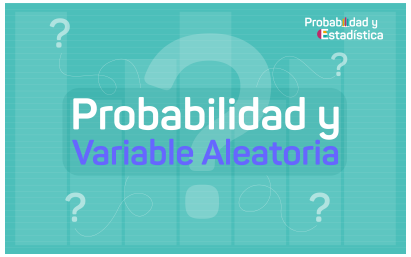

```
## sdd
##   6   9  10
## 0.4 0.4 0.2
```

Problema de los dados

```
n=10000
dd=sample(1:6, n*2, replace = TRUE)
mdd=matrix(dd,ncol = 2)
sdd=apply(mdd, 1,sum)
barplot(table(sdd))
prop.table(table(sdd))
```




```
## sdd
##      2      3      4      5      6      7      8      9     10     11     12
## 0.0284 0.0566 0.0794 0.1132 0.1401 0.1652 0.1405 0.1062 0.0838 0.0569 0.0297
```



Funciones en R

modelo	$F(x)$	X_p	$f(x)$	aleatorio
Binomial	pbinom	qbinom	dbinom	rbinom
Geometrica	pgeom	qgeom	dgeom	rgeom
Hypergeometrica	phyper	qhyper	dhyper	rhyper
Poisson	ppois	qpois	dpois	rpois
Beta	pbeta	qbeta	dbeta	rbeta
Cauchy	pcauchy	qcauchy	dcauchy	rcauchy
Chi-cuadrado	pchisq	qchisq	dchisq	rchisq
Exponencial	pexp	qexp	dexp	rexp
F	pf	qf	df	rf
Gamma	pgamma	qgamma	dgamma	rgamma
Logistic	plogis	qlogis	dlogis	rlogis
Log Normal	plnorm	qlnorm	dlnorm	rlnorm
Binomial Negativa	pnbinom	qnbinom	dnbinom	rnbinom
Normal	pnorm	qnorm	dnorm	rnorm
t-Student	pt	qt	dt	rt
Uniforme	punif	qunif	dunif	runif
Weibull	pweibull	qweibull	dweibull	rweibull

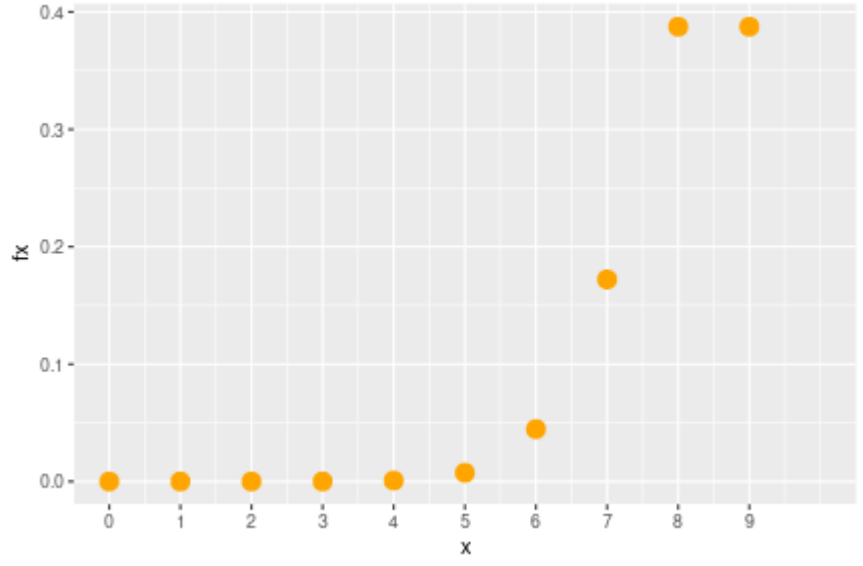
```r
# binomial
dbinom(x, size, prob)
pbinom(q, size, prob, lower.tail = TRUE)
qbinom(p, size, prob, lower.tail = TRUE)
rbinom(n, size, prob)
# Poisson
dpois(x, lambda)
ppois(q, lambda, lower.tail = TRUE)
qpois(p, lambda, lower.tail = TRUE)
rpois(n, lambda)
# geometrica
dgeom(x, prob)
pgeom(q, prob, lower.tail = TRUE)
qgeom(p, prob, lower.tail = TRUE)
rgeom(n, prob)

?rnorm
```

Modelos especiales - gráficos

binomial | Poisson | uniforme c. | normal | exponencial | Weibull | gamma

```
library(ggplot2)
x=0:9
fx=dbinom(x,9,0.90)
dat=data.frame(x,fx)
ggplot(dat) +
  geom_point(aes(x, fx),
             colour = "orange", size = 4) +
  scale_x_contins(limits = c(0, 10),
             breaks = c(0,1,2,3,4,5,6,7,8,9)
             labels = c('0','1','2','3','4',
```



Modelos especiales

En **R** los nombres de las funciones diseñadas para los cálculos requeridos están conformadas por dos partes:

- La primera parte con el propósito de la función (primera letra)
- La segunda parte hace referencia al modelo a utilizar (en el caso binomial binom)

letra	detalle
p	función de distribución acumulada $F(x)$
q	percentil
d	densidad de probabilidad $P(X = x)$
r	variable aleatoria

Para variables aleatoria discreta con distribución binomial $X \sim b(x; 20, 0.30)$

$$P(X = 7) = \binom{20}{7} 0.30^7 (10.30)^{(20-7)}$$

```
dbinom(7, 20, 0.30) #<<  # P(X = 7)
pbinom(7, 20, 0.30) #<<  # P(X  <= 7)
qbinom(0.25, 20, 0.30) #<<  Percentil 25
```

Experimento de Montecarlo

Es un método no determinista o estadístico numérico, usado para aproximar expresiones matemáticas complejas y costosas de evaluar con exactitud. El método se llamó así en referencia al Casino de Montecarlo (Mónaco) por ser “la capital del juego de azar”, al ser la ruleta un generador simple de números aleatorios. El nombre y el desarrollo sistemático de los métodos de Montecarlo datan aproximadamente de 1944 y se mejoraron enormemente con el desarrollo de la computadora.

El uso de los métodos de Montecarlo como herramienta de investigación, proviene del trabajo realizado en el desarrollo de la bomba atómica durante la Segunda Guerra Mundial en el Laboratorio Nacional de Los Álamos en EE. UU. Este trabajo conllevaba la simulación de problemas probabilísticos de hidrodinámica concernientes a la difusión de neutrones en el material de fisión. Esta difusión posee un comportamiento eminentemente aleatorio.
(tomado de Wikipedia)

Ejemplo : Se fabrican placas rectangulares cuyas longitudes en pulgadas se distribuyen como $N(2.0; 0.01)$ y cuyos anchos se distribuyen $N(3.0; 0.04)$. Suponga que las longitudes y los anchos son independientes. El área de una placa esta dada por $A = XY$. (Problema 3 capitulo 4 Navidi(2006))

- [a.] Utilice una muestra simulada de tamaño 1000 para estimar la media y la varianza de A.
- [b.] Estime la probabilidad de que $P(5.9 < A < 6.1)$.
- [c.] Construya una gráfica de distribución normal $(qqplot)$ para el área. ¿El área de una placa sigue una distribución normal?

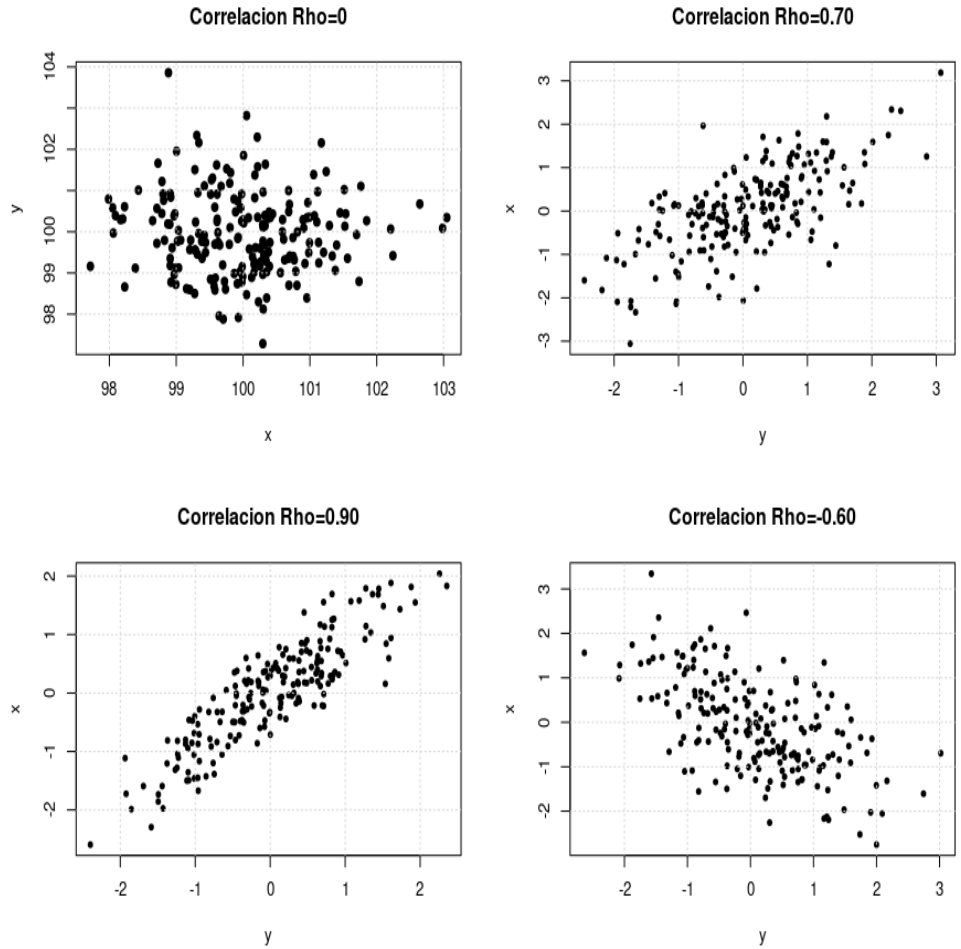
```r
X2=rnorm(1000,mean=2.0,sd=0.1)     #  generación de numeros aleatorios  de X
Y2=rnorm(1000,mean=3.0,sd=0.2)     #  generacion de numeros aleatorios  deY
Z2=data.frame(X2,Y2)               #  generacion de matriz de X,Y
A2=apply(Z2,1,prod)                #  area de la placa A=XY
mediaA=mean(A2)                    #  media del vector de areas
varianzaA=var(A2)                  #  varianza del vector de areas
B2=as.numeric(A2>5.9 & A2<6.1)     #  generacion de variable de 0,1,
#  con 1 donde cumplecondicion
Pro3c=sum(B2)/1000                 #  calculo de la  probabilidad
hist(A2)                           # histograma del valor de las areas
plot(density(A2))                  # grafico de la distribucion empirica de A2
qqnorm(A2)                         # grafico QQ de A2
summarytools::descr(A2)
```

Concepto de correlación

```r
gen.corr.data<- function(rho,n){
 x <- rnorm(n);
 z <- rnorm(n);
 y<- rho*x + sqrt(1-rho^2)*z ; result <-cbin
  return(result)
}
par(mfrow = c(2, 2)) # matriz de graficos 2x
muestra<-gen.corr.data(0,200);   plot(muestr
muestra<-gen.corr.data(0.7,200); plot(muestr
muestra<-gen.corr.data(0.9,200); plot(muestr
muestra<-gen.corr.data(-0.6,200); plot(muest
```

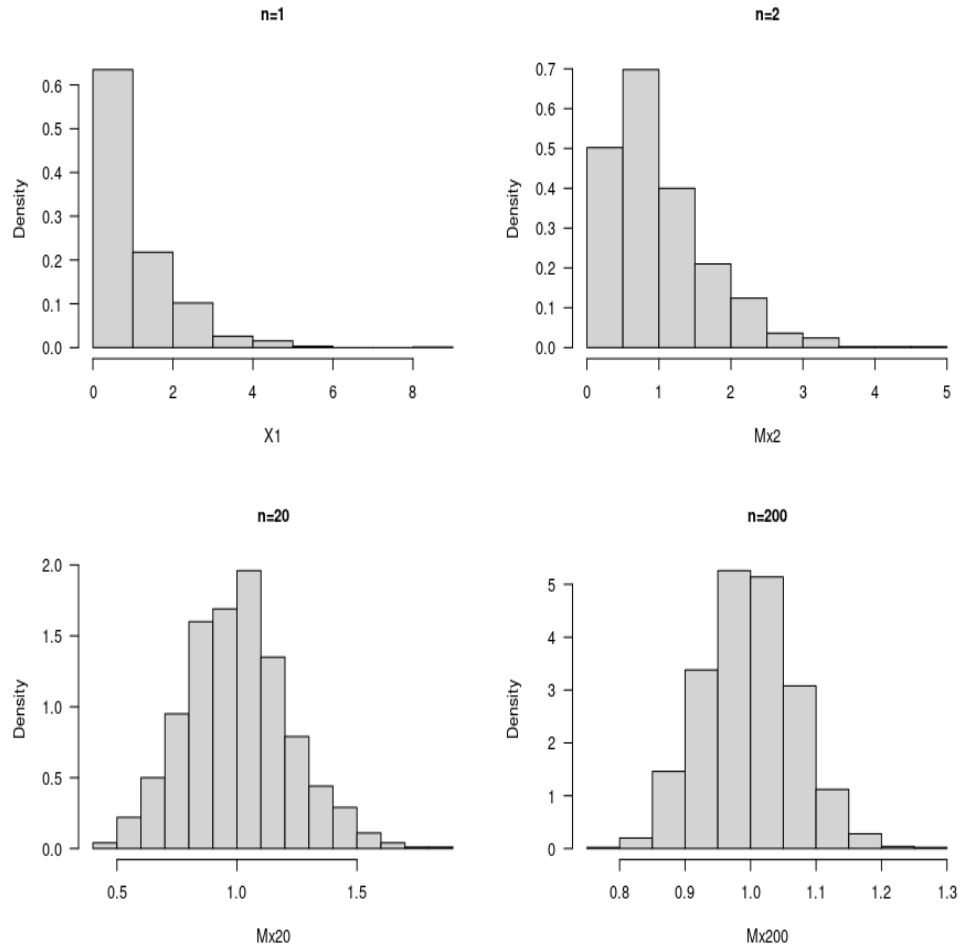
myCompiler

https://www.mycompiler.io/online-r-compiler



Teorema Central del Límite


```
n=200 ; m=1000*n
# distribucion exponencial-----------------
X=matrix(rexp(m,1),ncol=n)
# generacion de muestras-------------------
X1=X[ ,1]  ; X2=X[ ,1:2]; ; X20=X[ ,1:20] ;
# generacion de medias---------------
Mx2=apply(X2,1,mean); Mx20=apply(X20,1,mean)
# histogramas  de comparacion--------------
par(mfrow=c(2,2),cex=0.8, cex.axis=1, cex.la
hist(X1,  main = "n=1", freq=FALSE)
hist(Mx2, main ="n=2", freq=FALSE)
hist(Mx20, main = "n=20",freq=FALSE)
hist(Mx200, main = "n=200", freq=FALSE)
```



Pruebas de hipotesis

```r
# Codigos R
t.test(x, y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95, ...)

 prop.test(x, n, p = NULL,
          alternative = c("two.sided", "less", "greater"),
          conf.level = 0.95, correct = TRUE)

 var.test(x, y, ratio = 1,
         alternative = c("two.sided", "less", "greater"),
         conf.level = 0.95, ...)
```

Prueba t-Student para una media , con distribucion normal

$Ho : \mu \leq 1000$ $Ha : \mu > 1000$

```
x=c(11.1, 15.6, 11.1,  7.5,  7.9, 14.7,  6.3,  8.5,  8.0 , 7.6)
t.test(x,
       alternative = "less",
       mu = 10,
       conf.level = 0.95)
```

```
##
##     One Sample t-test
##
## data:  x
## t = -0.1682, df = 9, p-value = 0.4351
## alternative hypothesis: true mean is less than 10
## 95 percent confidence interval:
##      -Inf 11.68277
## sample estimates:
## mean of x
##      9.83
```

Prueba no parametrica de Signos

$Ho: Me \leq 15$

$Ha: Me > 15$

```
# install.packages("BSDA")
library(BSDA)
x=c(16,15,12,17,18,14,16,14,16,17,19,16,14,21,20,16,16,16)
SIGN.test(x,md=15,alternative = "greater")
```

```
    One-sample Sign-Test

data:  x
s = 13, p-value = 0.02452
alternative hypothesis: true median is greater than 15
95 percent confidence interval:
  16 Inf
sample estimates:
median of x
         16
```

Práctica