LABORATORIO 7 - Statistiche Campionarie

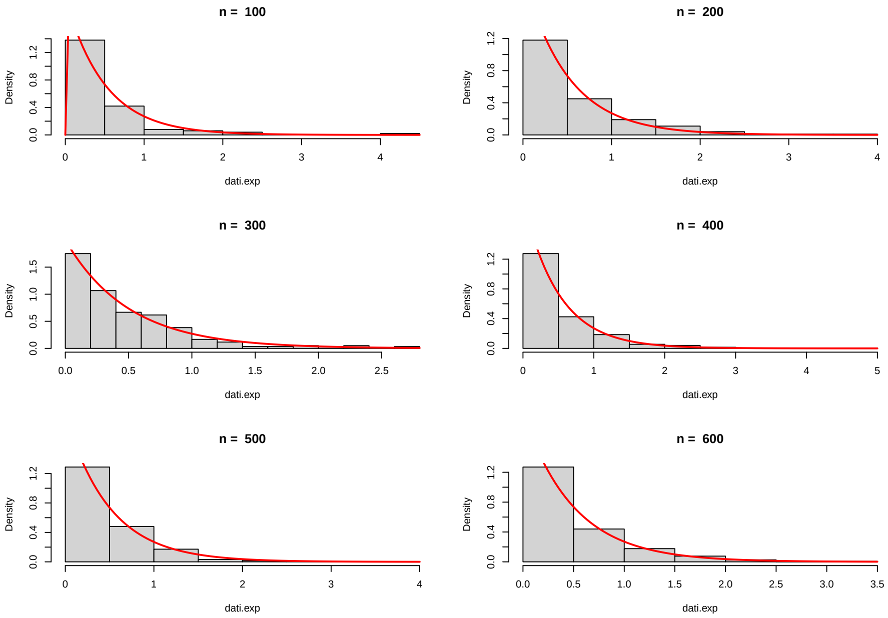
STATISTICA E LABORATORIO (CDL in INTERNET OF THINGS, BIG DATA, MACHINE LEARNING)

Anno Accademico 2021-2022

Modello esponenziale

Si consideri una variabile aleatoria di legge esponenziale di parametro 2. Si vuole verificare che, all'aumentare della numerosità del campione n, la forma dell'istogramma dei dati generati si avvicina al grafico della densità teorica della variabile aleatoria da cui i dati sono generati.

```r
par(mfrow=c(3,2))
for (n in seq(100, 600, by=100)) {

  lambda = 2

  dati.exp = rexp(n, rate=lambda)
  hist(dati.exp, prob=T, main=print(paste("n = ", n)))
  curve(dexp(x,rate = lambda),col='red',lwd=2,add=T)

}
```

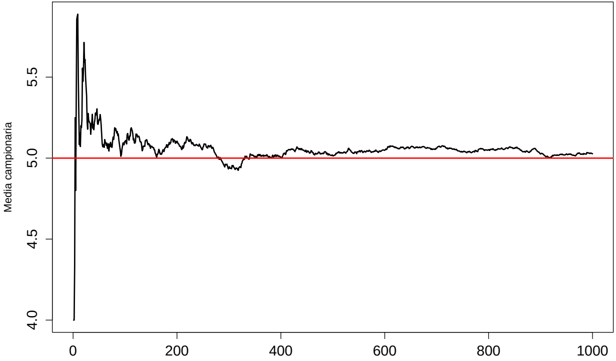

n = 100
n = 200
n = 300
n = 400
n = 500
n = 600
Density
dati.exp

Media di conteggi

Si consideri la sequenza dei valori osservati delle medie campionarie di $n = 1, \ldots, 1000$ variabili casuali indipendenti con distribuzione $P(5)$.

```
set.seed(1)
num_sim = 1000
x <- rpois(num_sim,5)
medie <- rep(0,num_sim)
# per calcolare la succesione delle medie campionarie
i <- 1
for(i in 1:num_sim){
  medie[i]<-mean(x[1:i])
  i<-i+1
}
# in alternativa nn<-1:length(x); medie<-cumsum(x)/nn
```

```
plot(1:num_sim, medie, xlim = c(0,num_sim), type = 'l', lty = 1, lwd
     cex.axis = 1.3, xlab = "", ylab = "Media campionaria")
abline(h = 5, lwd = 2, col = 'red')
```

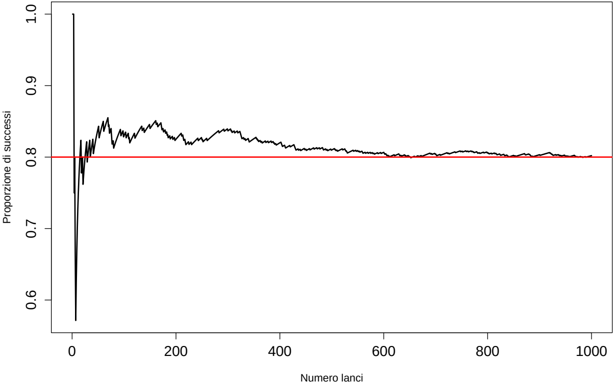


Lanci di una moneta truccata

Si vogliono simulare 1000 lanci di una moneta truccata in cui la probabilità di ottenere testa è pari a 0.8, e si intende studiare la proporzione di casi in cui si ottiene testa cambi al'aumentare del numero di lanci.

```
num_sim = 1000
n = 1
p = 0.8
set.seed(1)
dati.ber = rbinom(num_sim,n, p)
#somma delle bernulliane
somma.ber = cumsum(dati.ber)
#media dei primi k lanci
k <- 1:num_sim
medie <- somma.ber/k
```

```
plot(1:num_sim, medie, xlim = c(0,num_sim), type = 'l', lty = 1,
     lwd = 2, cex.axis = 1.3, xlab = "Numero lanci",
     ylab = "Proporzione di successi")
abline(h=p,lwd=2,col='red')
```

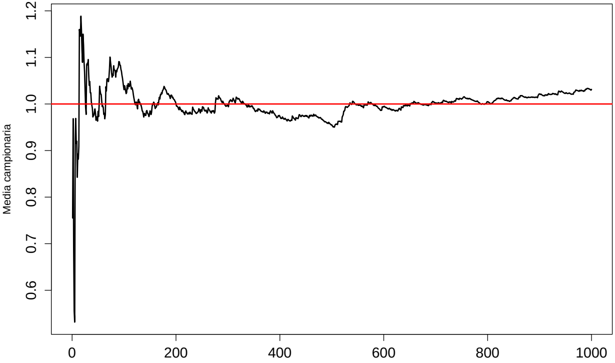


Media di una popolazione esponenziale

Si consideri la sequenza dei valori osservati delle medie campionarie di $n = 1, \ldots, 1000$ variabili casuali indipendenti con distribuzione $Exp(\lambda)$, con $\lambda = 1$.

```
set.seed(1)
num_sim = 1000
x <- rexp(num_sim,rate=1)
medie <- rep(0,num_sim)
# per calcolare la succesione delle medie campionarie
i <- 1
for(i in 1:num_sim){
  medie[i]<-mean(x[1:i])
  i<-i+1
}
# in alternativa nn<-1:length(x); medie<-cumsum(x)/nn
```

```r
plot(1:num_sim, medie, xlim = c(0,num_sim), type = 'l', lty = 1,
     lwd = 2, cex.axis = 1.3, xlab = "", ylab = "Media campionaria")
abline(h = 1, lwd = 2, col = 'red')
```

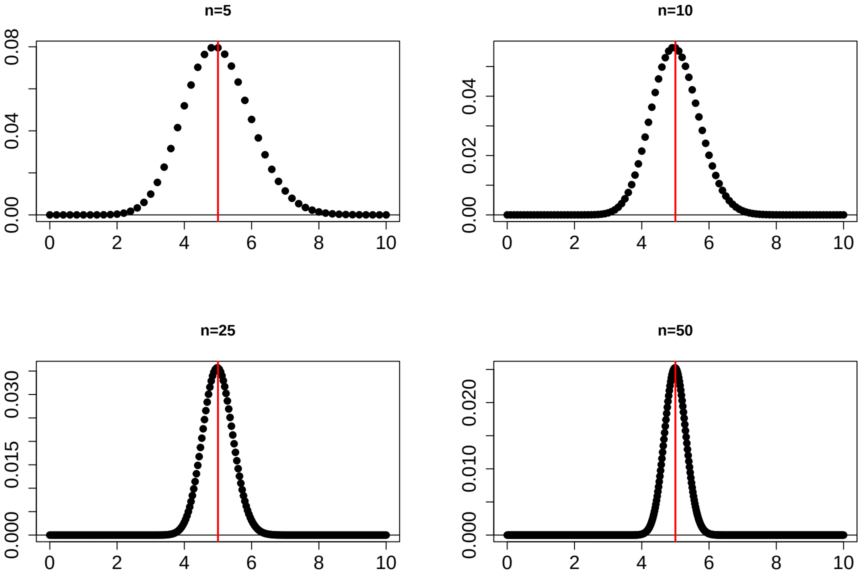


Si considerano la funzioni di probabilità di $\bar{X}_n$ per $n = 5, 10, 25, 50$ variabili casuali indipendenti con distribuzione $P(\lambda)$.

```
par(mfrow=c(2,2))
# si rappresentano le funzioni di probabilita' della media
#campionaria di n P(lambda) che corrispondono
# alle funzioni di probabilita' di una P(n*lambda) valutata in x/n
xx <- seq(0,50,1)
plot(xx/5, dpois(xx,5*5),pch=19,lwd=2,cex.axis=1.5,xlab=" ",
     ylab=" ",main = "n = 5")
abline(0,0,lwd=1);
abline(v=5,lwd=2,col='red')
xx <- seq(0,100,1)
plot(xx/10,dpois(xx,10*5),pch=19,lwd=2,cex.axis=1.5,xlab=" ",
     ylab=" ",main="n=10")
abline(0,0,lwd=1);abline(v=5,lwd=2,col='red')
xx<-seq(0,250,1)
plot(xx/25,dpois(xx,25*5),pch=19,lwd=2,cex.axis=1.5,xlab=" ",
     ylab=" ",main="n=25")
abline(0,0,lwd=1);
abline(v=5,lwd=2,col='red')
```

```
xx<-seq(0,500,1)
plot(xx/50,dpois(xx,50*5),pch=19,lwd=2,cex.axis=1.5,xlab=" ",
     ylab=" ",main="n=50")
abline(0,0,lwd=1)
abline(v=5,lwd=2,col='red')
```

```
par(mfrow=c(1,1))
```


n=5
n=10
n=25
n=50
0.00
0.04
0.08
0.02
0.000
0.015
0.030
0.010
0.020
0
2
4
6
8
10

Somma di conteggi

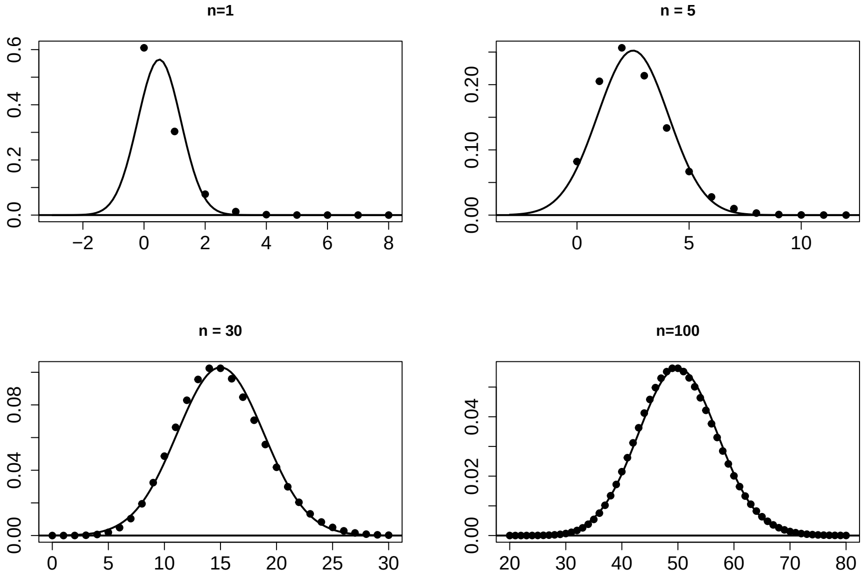
Si consideri una successione $\{X_n\}_{n\geq 1}$ di variabili casuali X_n, $n \geq 1$, indipendenti con distribuzione $P(\lambda)$.

```
par(mfrow=c(2,2))
xx<-seq(0,8,1)
plot(xx,dpois(xx,0.5),pch=19,lwd=2,xlim=c(-3,8),cex.axis=1.5,
     xlab=" ",ylab=" ",main="n=1")
# funzione di probabilita' esatta P(n*lambda)
curve(dnorm(x,0.5,sqrt(0.5)),-3,8,lwd=2,add=T)
# funzione di densita' normale approssimante N(n*lambda,n*lambda)
abline(0,0,lwd=2)
xx<-seq(0,12,1)
plot(xx,dpois(xx,0.5*5),pch=19,lwd=2,xlim=c(-3,12),cex.axis=1.5,
     xlab=" ",ylab=" ",main="n=5")
# funzione di probabilita' esatta P(n*lambda)
curve(dnorm(x,0.5*5,sqrt(0.5*5)),-3,12,lwd=2,add=T)
# funzione di densita' normale approssimante N(n*lambda,n*lambda)
abline(0,0,lwd=2)
```

```r
xx<-seq(0,30,1)
plot(xx,dpois(xx,0.5*30),pch=19,lwd=2,xlim=c(0,30),cex.axis=1.5,
     xlab=" ",ylab=" ",main="n=30")
# funzione di probabilita' esatta P(n*lambda)
curve(dnorm(x,0.5*30,sqrt(0.5*30)),0,30,lwd=2,add=T)
# funzione di densita' normale approssimante N(n*lambda,n*lambda)
abline(0,0,lwd=2)
```

```r
xx<-seq(20,80,1)
plot(xx,dpois(xx,0.5*100),pch=19,lwd=2,xlim=c(20,80),cex.axis=1.5,
     xlab=" ",ylab=" ",main="n=100")
# funzione di probabilita' esatta P(n*lambda)
curve(dnorm(x,0.5*100,sqrt(0.5*100)),20,80,lwd=2,add=T)
# funzione di densita' normale approssimante N(n*lambda,n*lambda)
abline(0,0,lwd=2)
```

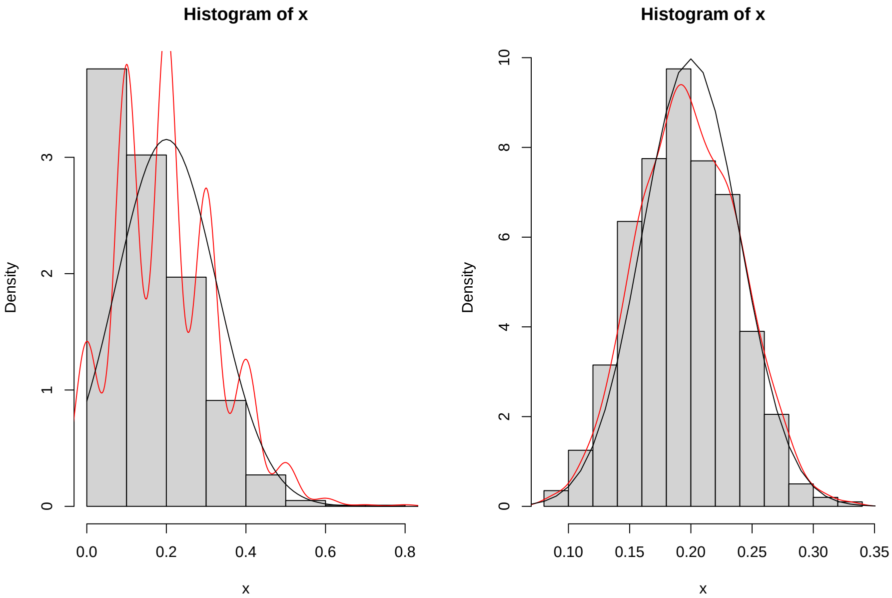
```r
par(mfrow=c(1,1))
```


n=1
0.0
0.2
0.4
0.6
−2
0
2
4
6
8
n = 5
0.00
0.10
0.20
0
5
10
n = 30
0.00
0.04
0.08
0
5
10
15
20
25
30
n=100
0.00
0.02
0.04
20
30
40
50
60
70
80

Media campionaria di osservazioni bernoulliane

```
set.seed(1)
par(mfrow=c(1,2))
x <-rbinom(1000,10,0.2) # 1000 valori simulati da una Bi(10,0.2),
# che corrisponde alla somma di n=10 Ber(0.2)
x <- x/10 # 1000 valori simulati per la media di n=10 Ber(0.2)
hist(x,probability = T) # istogramma
lines(density(x),col='red') # stima della densita' con il metodo
# del nucleo, densita' gaussiana approssimante
lines(seq(0,1,0.01),dnorm(seq(0,1,0.01),0.2,sqrt(0.2*0.8/10)))
x <- rbinom(1000,100,0.2) # 1000 valori simulati da una Bi(100,0.2),
# che corrisponde alla somma di n=100 Ber(0.2)
x <- x/100 # 1000 valori simulati per la media di n=100 Ber(0.2)
hist(x,probability = T) # istogramma
lines(density(x),col='red') # stima della densita' con il metodo
# del nucleo
lines(seq(0,1,0.01),dnorm(seq(0,1,0.01),0.2,sqrt(0.2*0.8/100)))
```

```
par(mfrow=c(1,1))
```


Histogram of x
Density
x
0.0
0.2
0.4
0.6
0.8
0
1
2
3
Histogram of x
Density
x
0.10
0.15
0.20
0.25
0.30
0.35
0
2
4
6
8
10

Procedura di controllo

Si è verificato un inconveniente su una linea di produzione che determina la presenza di $1/10$ di pezzi difettosi. La procedura di controllo della qualità prevede che, se si individuano almeno 5 pezzi difettosi su $n \geq 1$ scelti a caso, il processo viene posto in revisione. Sia S_n la somma di $n \geq 1$ variabili casuali $Ber(1/10)$ indipendenti. Si cerca il valore per n tale che ci sia una probabilità pari a 0.9 di porre il processo in revisione.

```
# Calcolo esatto di n
# funzione che fornisce P(S_n>=5)-0.9
probab<-function(x){
  xr<-round(x)
  # perche' in pbinom(q,size,prob) size deve essere intero
  1-pbinom(5,xr,1/10)-0.9}
# in alternativa pbinom(5,xr,1/10,lower.tail=FALSE)-0.9
uniroot(probab,c(0,100))$root
```

```
## [1] 90.50006
```

```
# la soluzione dell'equazione corrisponde al valore cercato di n
```

```r
# Calcolo approssimato di n
# funzione che fornisce l'approssimazione di P(S_n>=5)-0.9
# basata sulla distribuzione gaussiana N(n*(1/10),n*(1/10)*(9/10))

probab1<-function(x){1-pnorm(5,x*(1/10),sqrt(x*(1/10)*(9/10)))-0.9}

# in alternativa pnorm(5,x*(1/10),sqrt(x*(1/10)*(9/10),
# lower.tail=FALSE)-0.9
uniroot(probab1,c(0,100))$root
```

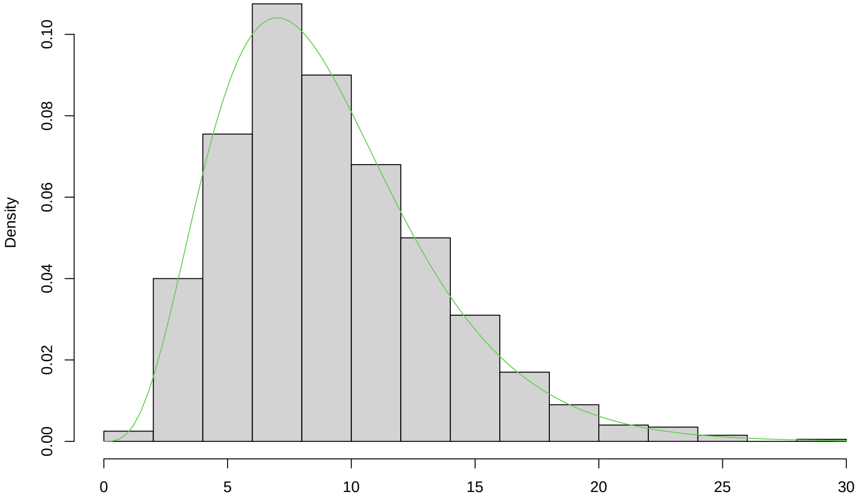
```
## [1] 85.56322
```

```r
# la soluzione dell'equazione
# corrisponde al valore cercato di n
```

Distribuzione della varianza campionaria

```
set.seed(1)
mu = 5
sigma = 1
n = 10
varianza = NULL

for(i in 1:1000){

  y <- rnorm(n, mu, sigma) # 10 valori simulati da una N(mu,sigma^2)

  varianza[i] <- (n-1)*var(y)/(sigma^2)

}

hist(varianza, freq = F,main="",xlab='') # istogramma
curve(dchisq(x,n-1),col=3,add=T) #densità chi-quadrato con n-1 gdl
```

```
par(mfrow=c(1,1))
```

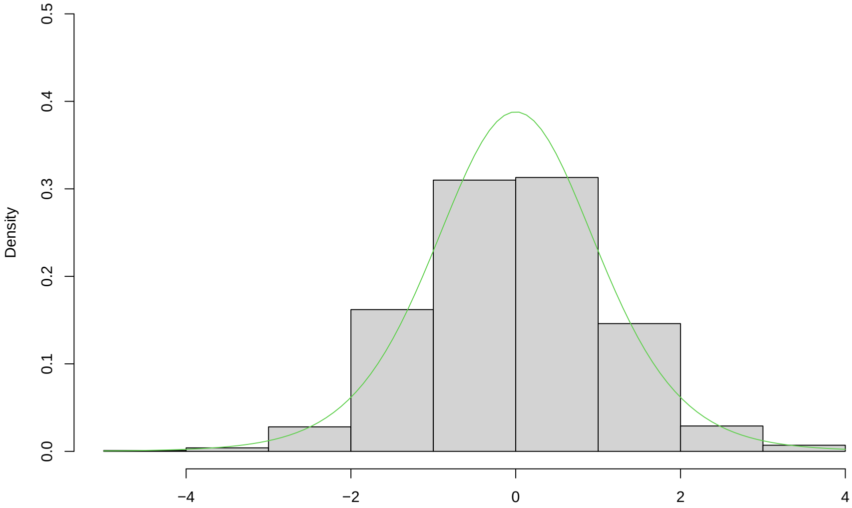


Density
0.00
0.02
0.04
0.06
0.08
0.10
0
5
10
15
20
25
30

Distribuzione della media campionaria Studentizzata

```
set.seed(1)
mu = 5
sigma = 1
n = 10
media_stud = NULL

for(i in 1:1000){
  y <- rnorm(n, mu, sigma) # 10 valori simulati da una N(mu,sigma^2,
  media_stud[i] <- (mean(y)-mu)/(sqrt(var(y)/n))
}
hist(media_stud, freq=F, main='',xlab='',ylim=c(0,0.5)) # istogramma
curve(dt(x,n-1),col=3,add=T) #densità t di Student con n-1 gdl
```

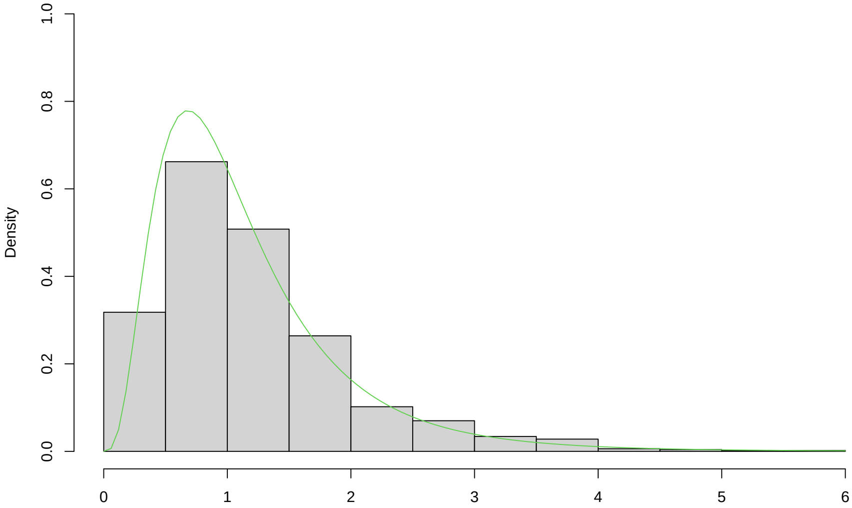
```
par(mfrow=c(1,1))
```


Density
0.0
0.1
0.2
0.3
0.4
0.5
−4
−2
0
2
4

Distribuzione del rapporto di due varianze

```
set.seed(1)
mu1 = 5;sigma1 = 1;mu2 = 4;sigma2 = 1;n1 = 10;n2 = 15;
rapp_v = NULL

for(i in 1:1000){
  y1 <- rnorm(n1, mu1, sigma1) # n1 valori simulati da
  #una N(mu1,sigma1^2),
  y2 <- rnorm(n2, mu2, sigma2) # n2 valori simulati da
  #una N(mu2,sigma2^2),
  varianza1 <- var(y1)/sigma1^2
  varianza2 <- var(y2)/sigma2^2
  rapp_v[i] <- varianza1/varianza2
}
hist(rapp_v, freq=F, main='',xlab='',ylim=c(0,1)) # istogramma
#densità F di Fisher con n1-1,n2-1 gdl
curve(df(x,n1-1,n2-1),col=3,add=T)
```

```
par(mfrow=c(1,1))
```


Density
0.0
0.2
0.4
0.6
0.8
1.0
0
1
2
3
4
5
6