Laboratorio di ST1
Lezione 4

Claudia Abundo

Dipartimento di Matematica
Università degli Studi Roma Tre

26 Marzo 2010

Gamma

```
?dgamma
```

The Gamma Distribution
Description
Density, distribution function, quantile function and random generation for the Gamma distribution with parameters shape and scale.

Usage
dgamma(x, shape, rate = 1, scale = 1/rate, log = FALSE)
pgamma(q, shape, rate = 1, scale = 1/rate, lower.tail = TRUE, log.p = FALSE)
qgamma(p, shape, rate = 1, scale = 1/rate, lower.tail = TRUE, log.p = FALSE)
rgamma(n, shape, rate = 1, scale = 1/rate)

Gamma 2

Arguments
x, q vector of quantiles.
p vector of probabilities.
n number of observations. If length(n) $>$ 1, the length is taken to be the number required.
rate an alternative way to specify the scale.
shape, scale shape and scale parameters. Must be positive, scale strictly.
log, log.p logical; if TRUE, probabilities/densities p are returned as log(p).
lower.tail logical; if TRUE (default), probabilities are P[X $<=$ x], otherwise, P[X $>$ x].

Details
If scale is omitted, it assumes the default value of 1.
The Gamma distribution with parameters shape = a and scale = s has density

$$f(x) = 1/(s^a Gamma(a)) x^(a-1) e^-(x/s)$$

$$for x >= 0, a > 0 and\ s > 0.$$

Gamma 3

```
curve(dgamma(x, shape = 0.5, rate = 5), col = "red",
ylab = "Densità", from = 0, to = 2, main =
"Distribuzione Gamma")
curve(dgamma(x, shape = 1, rate = 5), from = 0, to = 2,
add = T, lty = 2)
curve(dgamma(x, shape = 2, rate = 5), col = "blue",
from = 0, to = 2, add = T, lty = 3)
curve(dgamma(x, shape = 3, rate = 5), col = "green",
from = 0, to = 2, add = T, lty = 4)
legend(1.2,6, c("shape = 1/2", "shape = 1", "shape =
2", "shape = 3"), lty = c(1, 2, 3, 4))
```

Realizzazioni di Gamma

```
s<-c()
for (i in 1:10000) s[i]<-mean(rgamma(n=1,1))
hist(s,prob=T,xlim=c(0,4),ylim=c(0,2))
for (i in 1:10000) s[i]<-mean(rgamma(n=5,1))
hist(s,prob=T,xlim=c(0,4),ylim=c(0,2))
for (i in 1:10000) s[i]<-mean(rgamma(n=10,1))
hist(s,prob=T,xlim=c(0,4),ylim=c(0,2))
for (i in 1:10000) s[i]<-mean(rgamma(n=20,1))
hist(s,prob=T,xlim=c(0,4),ylim=c(0,2))
```

Chi Quadrato

Per $n > 0$, la distribuzione gamma con parametro di forma k = n / 2 e parametro di scala 2 è detta distribuzione chi-square con n gradi di libertà.

The (non-central) Chi-Squared Distribution
Density, distribution function, quantile function and random generation for the chi-squared distribution with df degrees of freedom and optional non-centrality parameter ncp.

```
dchisq(x, df, ncp=0, log = FALSE)
pchisq(q, df, ncp=0, lower.tail = TRUE, log.p = FALSE)
qchisq(p, df, ncp=0, lower.tail = TRUE, log.p = FALSE)
rchisq(n, df, ncp=0)
```

Chi Quadrato 2

```
curve(dchisq(x, df = 3), 0.,20, ylab = "Densità", col =
"red", main = "Distribuzione CHI-Quadrato")

curve(dchisq(x, df = 5), 0.,20, ylab = "Densità", col =
"blue", lty = 2, add = T)

curve(dchisq(x, df = 7), 0.,20, ylab = "Densità", col =
"green", lty = 2, add = T)

legend(10,0.2, c("gdl = 3", "gdl = 5", "gdl = 7"), lty
= c(1, 2, 3))
```

Chi Quadrato 3: 1 g.d.l.

Se Z ha distribuzione normale standard, Z^2 ha una distribuzione chi-quadrato con 1 grado di libertà.

```
curve(dchisq(x, df = 1), 0.,20, ylab = "Densit`a",col =
+ "red", main = "Distribuzione CHI-Quadrato")

qchisq(0.95, df=1)

points(qchisq(0.95, df=1), 0, pch="|")
```

gli ultimi due comandi calcolano e individuano sul grafico il quantile 0.95.
Il quantile 0.95 è 3.84. Notare che la radice quadrata di 3.84 è 1.96:

$$P(Z^2 \leq 3.84) = 0.95 \Leftrightarrow P(|Z| \leq 1.96) = 0.95$$

Cioè la probabilità che il chi-quadrato con 1 g.l. sia minore di 3.84 è uguale alla probabilità che la normale standard sia compresa tra -1.96 e 1.96.

t di Student

Date la v.a. Z standardizzata e la v.a. X chi quadro con n g.d.l la v.a. $T = \frac{Z}{(X/n)^1/2}$ è una t con n g.d.l.

```
curve(dnorm(x), -5, 5, ylab = "Densità", col = "red",
main = "Distribuzione t di Student")

curve(dt(x, df = 1), -6, 6, lty = 2, col = "blue", add
= T)

curve(dt(x, df = 2), -6, 6, lty = 3, col = "green", add
= T)

legend(2,0.3, c("Z", "t, gdl = 1", "t, gdl = 2"), lty =
c(1, 2, 3))
```

df = degrees of freedom

F di Fisher

Date due v.a. chi quadro con rispettivamente n e m g.d.l. la v.a. $F = \frac{\chi^2_n/n}{\chi^2_m/m}$ è una F di Fisher con n e m g.d.l.
Density, distribution function, quantile function and random generation for the F distribution with df1 and df2 degrees of freedom (and optional non-centrality parameter ncp).

df(x, df1, df2, ncp, log = FALSE)
pf(q, df1, df2, ncp, lower.tail = TRUE, log.p = FALSE)
qf(p, df1, df2, ncp, lower.tail = TRUE, log.p = FALSE)
rf(n, df1, df2, ncp)

x, q vector of quantiles.
p vector of probabilities.
n number of observations. If length(n) $>$ 1, the length is taken to be the number required.
df1, df2 degrees of freedom. Inf is allowed.
ncp non-centrality parameter. If omitted the central F is assumed.
log, log.p logical; if TRUE, probabilities p are given as log(p).
lower.tail logical; if TRUE (default), probabilities are $P[X <= x]$, otherwise, $P[X > x]$.

F di Fisher 2

```
curve(df(x, df1 =3, df2 = 2), 0, 2, ylab = "Densità",
col = "red", main = "Distribuzione f di Fisher")

curve(df(x, df1 =4, df2 = 3), 0, 2, ylab = "Densità",
col = "blue", add = T)
```

Beta

```
curve(dbeta(x,5,3), ylim=c(0,3), xlim=c(0,1),
ylab="Densità Beta")
```

Beta 2

Cambio di parametri

```
curve(dbeta(x,1,1), ylim=c(0,3), xlim=c(0,1), ylab="Densità
Beta")

curve(dbeta(x,0.1,1), add = TRUE, lty = 3, col = "red")

curve(dbeta(x,1,0.1), add = TRUE, lty = 3, col = "red")

curve(dbeta(x,0.1,0.1), add = TRUE, lty = 2, lwd = 2, col =
"green")

curve(dbeta(x,4,4), add = TRUE, lty = 2, lwd = 2)

curve(dbeta(x,2,6), add = TRUE, lty = 2, lwd = 3, col = "blue")

curve(dbeta(x,6,2), add = TRUE, lty = 2, lwd = 3, col = "blue")

curve(dbeta(x,2,6), add = TRUE, lty = 2, lwd = 3, col = "blue")

curve(dbeta(x,2,2), add = TRUE, lty = 2, lwd = 3)
```

Below, we will show how to export a graph into a JPEG image file. The main trick is to call the function jpeg(). It has only one required argument, which is the filepath to the JPEG file. Once the function jpeg() has been called, all subsequent graphing will be done inside of the specified JPEG file. Therefore, the following code will create a line graph in a file named test.jpg. (The file will be saved in the directory where R was originally run.)

```
> data <- read.table("data/dow-jones.csv", header=TRUE)
# read data
> jpeg("test.jpg")
# graph to JPEG file
> plot(data$year, data$start, type="line",
# create graph
+ xlab="Year", ylab="DJIA", main="Dow Jones Industrial Average")
# label axes and add title
> dev.off()
# close file
> q()
```

Before quitting R, it is important to close the file with the command dev.off() to ensure that R does not continue to write to it in future sessions.