

Normal {stats}

R Documentation

The Normal Distribution

Description

Density, distribution function, quantile function and random generation for the normal distribution with mean equal to mean and standard deviation equal to sd.

Usage

```
dnorm(x, mean=0, sd=1, log = FALSE)
pnorm(q, mean=0, sd=1, lower.tail = TRUE, log.p = FALSE)
qnorm(p, mean=0, sd=1, lower.tail = TRUE, log.p = FALSE)
rnorm(n, mean=0, sd=1)
```

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.

mean

vector of means.

sd

vector of standard deviations.

log, log.p

logical; if TRUE, probabilities p are given as $\log(p)$.

lower.tail

logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.

Details

If mean or sd are not specified they assume the default values of 0 and 1, respectively.

The normal distribution has density

$$f(x) = 1/(\text{sqrt}(2 \pi) \text{sigma}) e^{-((x - \text{mu})^2/(2 \text{sigma}^2))}$$

where *mu* is the mean of the distribution and *sigma* the standard deviation.

qnorm is based on Wichura's algorithm AS 241 which provides precise results up to about 16 digits.

Value

dnorm gives the density, pnorm gives the distribution function, qnorm gives the quantile function, and rnorm generates random deviates.

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

Wichura, M. J. (1988) Algorithm AS 241: The Percentage Points of the Normal Distribution. *Applied Statistics*, **37**, 477–484.

See Also

[runif](#) and [.Random.seed](#) about random number generation, and [dlnorm](#) for the *Lognormal* distribution.

Examples

```

dnorm(0) == 1/ sqrt(2*pi)
dnorm(1) == exp(-1/2)/ sqrt(2*pi)
dnorm(1) == 1/ sqrt(2*pi*exp(1))

## Using "log = TRUE" for an extended range :
par(mfrow=c(2,1))
plot(function(x)dnorm(x, log=TRUE), -60, 50, main = "log
{ Normal density }")
curve(log(dnorm(x)), add=TRUE, col="red",lwd=2)
mtext("dnorm(x, log=TRUE)", adj=0);
mtext("log(dnorm(x))", col="red", adj=1)

plot(function(x)pnorm(x, log=TRUE), -50, 10, main = "log
{ Normal Cumulative }")
curve(log(pnorm(x)), add=TRUE, col="red",lwd=2)
mtext("pnorm(x, log=TRUE)", adj=0);
mtext("log(pnorm(x))", col="red", adj=1)

## if you want the so-called 'error function'
erf <- function(x) 2 * pnorm(x * sqrt(2)) - 1
## (see Abrahamowitz and Stegun 29.2.29)
## and the so-called 'complementary error function'
erfc <- function(x) 2 * pnorm(x * sqrt(2), lower=FALSE)

```

[Package *stats* version 2.0.1 [Index](#)]