Package ‘lestat’

October 13, 2022

Version  1.9
Date  2018-06-12
Title  A Package for Learning Statistics
Author  Petter Mostad <mostad@chalmers.se>
Maintainer  Petter Mostad <mostad@chalmers.se>
Depends  R (>= 1.8.0), stats, MASS
Description  Some simple objects and functions to do statistics using linear models and a Bayesian framework.
License  GPL-2
LazyLoad  yes
NeedsCompilation  no
Repository  CRAN
Date/Publication  2018-06-12 18:29:29 UTC

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>lestat-package</td>
<td>2</td>
</tr>
<tr>
<td>anovatable</td>
<td>3</td>
</tr>
<tr>
<td>betabinomial</td>
<td>4</td>
</tr>
<tr>
<td>betadistribution</td>
<td>5</td>
</tr>
<tr>
<td>binomialbeta</td>
<td>6</td>
</tr>
<tr>
<td>binomialdistribution</td>
<td>7</td>
</tr>
<tr>
<td>cdf</td>
<td>7</td>
</tr>
<tr>
<td>compose</td>
<td>8</td>
</tr>
<tr>
<td>conditional</td>
<td>9</td>
</tr>
<tr>
<td>contrast</td>
<td>10</td>
</tr>
<tr>
<td>credibilityinterval</td>
<td>10</td>
</tr>
<tr>
<td>designBalanced</td>
<td>11</td>
</tr>
<tr>
<td>designFactorial</td>
<td>12</td>
</tr>
<tr>
<td>designManyGroups</td>
<td>13</td>
</tr>
<tr>
<td>designOneGroup</td>
<td>14</td>
</tr>
<tr>
<td>designTwoGroups</td>
<td>14</td>
</tr>
<tr>
<td>Function</td>
<td>Page</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>difference</td>
<td>15</td>
</tr>
<tr>
<td>discretedistribution</td>
<td>16</td>
</tr>
<tr>
<td>expectation</td>
<td>17</td>
</tr>
<tr>
<td>expgamma</td>
<td>17</td>
</tr>
<tr>
<td>fdistribution</td>
<td>18</td>
</tr>
<tr>
<td>fittedvalues</td>
<td>19</td>
</tr>
<tr>
<td>gammadistribution</td>
<td>20</td>
</tr>
<tr>
<td>invcdf</td>
<td>21</td>
</tr>
<tr>
<td>leastsquares</td>
<td>21</td>
</tr>
<tr>
<td>linearmodel</td>
<td>22</td>
</tr>
<tr>
<td>linearpredict</td>
<td>23</td>
</tr>
<tr>
<td>marginal</td>
<td>25</td>
</tr>
<tr>
<td>mdiscretedistribution</td>
<td>26</td>
</tr>
<tr>
<td>mnormal</td>
<td>26</td>
</tr>
<tr>
<td>mnormalexpgamma</td>
<td>27</td>
</tr>
<tr>
<td>mnormalgamma</td>
<td>28</td>
</tr>
<tr>
<td>mtdistribution</td>
<td>29</td>
</tr>
<tr>
<td>muniformdistribution</td>
<td>30</td>
</tr>
<tr>
<td>normal</td>
<td>31</td>
</tr>
<tr>
<td>normalexpgamma</td>
<td>32</td>
</tr>
<tr>
<td>normalgamma</td>
<td>33</td>
</tr>
<tr>
<td>p.value</td>
<td>34</td>
</tr>
<tr>
<td>plot.normal</td>
<td>35</td>
</tr>
<tr>
<td>poissondistribution</td>
<td>36</td>
</tr>
<tr>
<td>posteriornormal1</td>
<td>36</td>
</tr>
<tr>
<td>posteriornormal2</td>
<td>37</td>
</tr>
<tr>
<td>precision</td>
<td>38</td>
</tr>
<tr>
<td>print.normal</td>
<td>39</td>
</tr>
<tr>
<td>probability</td>
<td>39</td>
</tr>
<tr>
<td>probabilitydensity</td>
<td>40</td>
</tr>
<tr>
<td>simulate.normal</td>
<td>41</td>
</tr>
<tr>
<td>summary.normal</td>
<td>42</td>
</tr>
<tr>
<td>tdistribution</td>
<td>43</td>
</tr>
<tr>
<td>uniformdistribution</td>
<td>44</td>
</tr>
<tr>
<td>variance</td>
<td>44</td>
</tr>
</tbody>
</table>

**Index**

2

---

**lestat-package**

*LEarning STATistics using Bayesian object oriented computation*

---

**Description**

The package contains a number of simple functions which can be combined to implement simple Bayesian computations.

**Details**
With this package, functions can be used to create objects representing probability distributions of many different types. These distributions can then be transformed and combined in different ways, representing statistical modelling. The result is an object-oriented way to do Bayesian computation with R.

Author(s)

Petter Mostad <mostad@chalmers.se>

References

Please see more information at www.math.chalmers.se/~mostad/

Examples

prior <- normalexpgamma() #Generate a two-parameter flat prior
full <- linearpredict(prior, rep(1, 7)) #Extend with normal distribution
data <- runif(7) #Generate data
posterior <- conditional(full, 1:7, data) #Condition on parameters
credibilityinterval(marginal(posterior, 1)) #Investigate posterior

Description

Given data and a matrix describing a design for a linear model, the function creates an ANOVA table, using sums of squares based on a subdivision of the columns of the design matrix given as the third argument for the function.

Usage

anovatable(data, design, subdivisions = c(1, dim(design)[2] - 1))
**betabinomial**

**Arguments**
- **data** A vector with data values
- **design** A matrix with the same number of rows as there are data values. The matrix represents the design matrix for the linear model the ANOVA table is based on.
- **subdivisions** A vector of integers summing to the number of columns in the design matrix. The number of rows of the ANOVA table will be equal to the length of this vector.

**Value**
An ANOVA table.

**Author(s)**
Petter Mostad <mostad@chalmers.se>

**Examples**
```r
data1 <- simulate(normal(2.7, log(0.7)), 3)
data2 <- simulate(normal(4.0, log(0.7)), 5)
data3 <- simulate(normal(3.2, log(0.7)), 3)
data4 <- simulate(normal(4.1, log(0.7)), 4)
anovatable(c(data1, data2, data3, data4), designManyGroups(c(3,5,3,4)))
```

---

**betabinomial**

*Create an Object Representing a Beta-Binomial Distribution*

**Description**
Create an object representing a Beta-Binomial distribution. This can be used for a Binomial distribution where there is uncertainty about the probability of success, and this uncertainty is represented by a Beta distribution.

**Usage**
```r
betabinomial(n, alpha, beta)
```

**Arguments**
- **n** the number of trials in the binomial distribution (a positive integer).
- **alpha** the alpha parameter of the underlying Beta distribution.
- **beta** the beta parameter of the underlying Beta distribution.

**Value**
An object of class "betabinomial" and class "probabilitydistribution".
betadistribution

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

betadistribution, binomialdistribution, binomialbeta

Examples

dist <- betabinomial(10, 5.5, 3.3)
cdf(dist, 3)

Description

Create an object representing a Beta distribution.

Usage

betadistribution(alpha, beta)

Arguments

alpha The alpha parameter.

beta The beta parameter.

Value

A Beta probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

betabinomial, binomialbeta

Examples

dist <- betadistribution(4, 6)
plot(dist)
binomialbeta

Create an Object Representing a bivariate Binomial Beta Distribution

Description

Create an object representing a bivariate distribution, where the first variable is marginally Beta distributed, and the second variable is binomially distributed with probability given by the first variable.

Usage

binomialbeta(n, alpha, beta)

Arguments

n the number of trials in the binomial distribution (a positive integer).
alpha the alpha parameter of the Binomial distribution.
beta the beta parameter of the Binomial distribution.

Value

An object of class "binomialbeta" and class "probabilitydistribution".

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

betadistribution, binomialdistribution, betabinomial

Examples

dist <- binomialbeta(10, 5.5, 12.3)
plot(dist)
Create an Object Representing a Binomial Distribution

Description

Create an object representing a Binomial distribution

Usage

binomialdistribution(ntrials, probability)

Arguments

ntrials the number of trials in the binomial distribution (a positive integer).
probability the probability for success in each trial (a number between 0 and 1).

Value

An object of class "binomialdistribution" and class "probabilitydistribution".

Author(s)

Petter Mostad <mostad@chalmers.se>

Examples

dist <- binomialdistribution(10, 0.4)
cdf(dist, 3)

Compute Cumulative Distribution Function

Description

Compute the value of the cumulative distribution function for univariate distributions.

Usage

cdf(object, val)

Arguments

object A univariate probability distribution.
val The probability less than or equal to val is computed.
Value

The probability that a variable with distribution object is less than or equal to val.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

invcdf

Examples

cdf(normal(3, 2), 1)

compose
Building a new probability distribution from an old.

Description

The command can be used to generate a new distribution from an old, which is given as the first argument. The new distribution has the old as the marginal for the first variable. The conditional distribution for the second variable is specified with the remaining arguments.

Usage

compose(object, type, ...)

Arguments

object A probability distribution

type A text string specifying the type of the conditional distribution given the old distribution.

... Additional arguments specifying the conditional distribution.

Value

Depends on the input; may be a multivariate discrete distribution, or a Binomialbeta distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

Examples

joint <- compose(uniformdistribution(), "binomialdistribution", 5)
joint2 <- compose(discretedistribution(1:6), "discretedistribution",
1:6, matrix(c(1:36), 6, 6))
The Conditional Distribution

Description

Given a multivariate distribution, the conditional distribution is computed when the variables with the given indices are set to the given values.

Usage

```
conditional(object, v, val)
```

Arguments

- `object`: A multivariate probability distribution.
- `v`: A vector of the indices of the variables whose values should be fixed.
- `val`: A vector, of the same length as `v`, with the values at which these variables should be fixed.

Value

An object representing the conditional probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

`marginal`

Examples

```
prior <- normalexpgamma() # Generate a two-parameter flat prior
full <- linearpredict(prior, rep(1, 7)) # Normal extension
data <- simulate(uniformdistribution(), 7) # Generate data
posterior <- conditional(full, 1:7, data) # Condition on parameters
credibilityinterval(marginal(posterior, 1)) # Investigate posterior
```
Description

For some distributions, like the multivariate Normal-ExpGamma and the multivariate Normal-Gamma, a new distribution is constructed from a linear combination of all but the last variables, and the last variable.

Usage

contrast(object, v)

Arguments

- **object**: A multivariate Normal-ExpGamma distribution or multivariate Normal-Gamma distribution.
- **v**: A vector specifying the linear combination.

Value

A Normal-ExpGamma distribution or a Normal-Gamma distribution, depending on the input.

Author(s)

Petter Mostad <mostad@chalmers.se>

Examples

```r
data1 <- simulate(normal(13, log(0.4)), 3)
data2 <- simulate(normal(14, log(0.4)), 5)
data3 <- simulate(normal(12, log(0.4)), 6)
dist <- linearmodel(c(data1, data2, data3), designManyGroups(c(3,5,6)))
diff <- contrast(dist, c(0, 1, -1))
credibilityinterval(marginal(diff, 1))
```

---

credibilityinterval  
Compute Credibility Interval for a Univariate Distribution

Description

Given a univariate continuous distribution, a credibility interval is computed. Note that the interval is constructed so that there is an equal probability to be above or below the interval.

Usage

credibilityinterval(object, prob = 0.95)
**Arguments**

- **object** A univariate continuous distribution.
- **prob** The probability inside the credibility interval.

**Value**

A vector of length two, specifying the interval.

**Author(s)**

Petter Mostad <mostad@chalmers.se>

**See Also**

- `p.value`

**Examples**

```r
credibilityinterval(normal())
```

---

**Description**

The function creates a design matrix suitable for analyzing results from an experiment where a set of factors are analysed in a balanced design: The argument `factors` lists the number of levels of each factor, and each possible combination of levels of factors is tried out a number of times given by `replications`.

**Usage**

```r
designBalanced(factors, replications = 1, interactions = FALSE)
```

**Arguments**

- **factors** A vector of integers, listing the number of levels of each of the factors.
- **replications** An integer: The number of times each combination of factor levels is tried out.
- **interactions** If true, the design matrix will include columns for all possible interactions of the factors.

**Value**

A matrix where the number of rows equals the product of `factors` and `replications`. The matrix will have only 0’s and 1’s as values.
**Author(s)**

Petter Mostad <mostad@chalmers.se>

**See Also**

designFactorial, designOneGroup, designTwoGroups, designManyGroups

**Examples**

designFactorial(3, 2)
Create a Design Matrix for Several Groups of Normal Observations

Description

A design matrix is created, to be used for the analysis of data assumed to come from several normal distributions.

Usage

designManyGroups(v)

Arguments

v        A vector of integers, indicating how many observations there are in each group.

Value

A matrix consisting of 0’s and 1’s. The number of columns is equal to the length of v. The number of rows is equal to the sum of the values of v.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

designOneGroup, designTwoGroups, designBalanced, designFactorial

Examples

data1 <- simulate(normal(3.3, log(2)), 9)
data2 <- simulate(normal(4.5, log(2)), 8)
data3 <- simulate(normal(2.9, log(2)), 7)
design <- designManyGroups(c(9,8,7))
posterior <- linearmodel(c(data1, data2, data3), design)
plot(posterior)
designOneGroup  Create a Design Matrix for One Group of Observations

Description
A design matrix is created, to be used for the analysis of data assumed to come from one normal distribution.

Usage
designOneGroup(n)

Arguments
n The number of data values.

Value
A matrix consisting only of 1’s, with one column and with the number of rows given by n.

Author(s)
Petter Mostad <mostad@chalmers.se>

See Also
designTwoGroups, designManyGroups, designBalanced, designFactorial

Examples
data <- simulate(normal(4, log(1.3)), 9)
design <- designOneGroup(9)
posterior <- linearmodel(data, design)
credibilityinterval(marginal(posterior, 1))


designTwoGroups  Create a Design Matrix for Two Groups of Observations

Description
A design matrix is created, to be used for the analysis of data assumed to come from two normal distributions.

Usage
designTwoGroups(n, m)
difference

Arguments

n  The number of data values in the first group.
m  The number of data values in the second group.

Value

A matrix consisting of 1’s and 0’s, with two columns, and with the number of rows given by $n + m$.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

designOneGroup, designManyGroups, designBalanced, designFactorial

Examples

data1 <- simulate(normal(3, log(2)), 7)
data2 <- simulate(normal(5, log(2)), 9)
design <- designTwoGroups(7, 9)
posterior <- linearmodel(c(data1, data2), design)
credibilityinterval(marginal(posterior, 1))

Description

Given two univariate distributions, an attempt is made to create the (approximate) difference between these.

Usage

difference(object1, object2)

Arguments

object1  A univariate normal or tdistribution.
object2  A univariate normal or tdistribution.

Value

A univariate normal or tdistribution, as appropriate.

Author(s)

Petter Mostad <mostad@chalmers.se>
Examples

data1 <- simulate(normal(8, log(1.5)), 6)
prior1 <- marginal(linearmodel(data1, designOneGroup(6)), 1)
data2 <- simulate(normal(10, log(2.8)), 7)
prior2 <- marginal(linearmodel(data2, designOneGroup(7)), 1)
prior <- difference(prior1, prior2)
credibilityinterval(prior)

discretedistribution

Create Object Representing a Discrete Distribution

Description

An object representing a discrete distribution is created, based on explicitly given possible values and probabilities for these.

Usage

discretedistribution(vals, probs = rep(1, length(vals)))

Arguments

vals

A vector listing the possible values of the discrete distribution.

probs

If given, probs must have the same length as vals, and should list the probabilities of the possible values. If not given, all possible values are assigned equal probabilities.

Value

A discrete probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

Examples

dist <- discretedistribution(1:10)
expectation(dist)
variance(dist)
expectation

Compute Expectation

Description
Compute the expectation of a probability distribution.

Usage
expectation(object)

Arguments
object A probability distribution.

Value
The expectation of the probability distribution.

Author(s)
Petter Mostad <mostad@chalmers.se>

See Also
variance

Examples
expectation(normal(3, log(2)))
expectation(binomialdistribution(7, 0.3))

expgamma
Create an ExpGamma distribution

Description
Create an ExpGamma Distribution: If a variable has a Gamma distribution with parameters alpha and beta, then its logarithm has an ExpGamma distribution with parameters alpha, beta, and gamma = 1.

Usage
expgamma(alpha = 1, beta = 1, gamma = -2)
Arguments
alpha  The "shape" parameter of the corresponding Gamma distribution.
beta   The "rate" parameter of the corresponding Gamma distribution.
gamma  The scale parameter for the logarithmic scale. By default, gamma = -2.

Details
The ExpGamma has probability density function

\[ f(x|\alpha, \beta, \gamma) = \exp(\alpha \gamma x - \beta \exp(\gamma x)) \]

Value
An ExpGamma distribution.

Author(s)
Petter Mostad <mostad@chalmers.se>

See Also
gammadistribution

Examples
```r
dist <- expgamma(4, 6)
plot(dist)
```

---

**fdistribution**

Create an F distribution

Description
Create a univariate F distribution.

Usage
```r
fdistribution(df1 = 1, df2 = 1)
```

Arguments
df1  The first degree of freedom: Should be a positive number.
df2  The second degree of freedom: Should be a positive number.

Value
An F distribution.
fittedvalues  

Author(s)

Petter Mostad <mostad@chalmers.se>

Examples

dist <- fdistribution(10, 3)
cdf(dist, 4)
expectation(dist)

---

fittedvalues  

Compute Fitted Values for a Linear Model

Description

Given a vector of data values and a design matrix, the fitted values for a linear model is computed.

Usage

fittedvalues(data, design)

Arguments

data  
A data vector.
design  
A design matrix. The number of rows must be equal to the length of the data vector.

Details

The fitted values represent the expected values all but the last variables in the posterior for the linear model.

Value

A vector of values of length equal to the number of columns in the design matrix.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

linearmodel, leastsquares, linearpredict

Examples

xdata <- simulate(uniformdistribution(), 14)
ydata <- xdata + 4 + simulate(normal(), 14)*0.1
plot(xdata, ydata)
design <- cbind(1, xdata)
lines(xdata, fittedvalues(ydata, design))
gammadistribution  

*Create a Gamma Distribution*

**Description**
Create a Gamma distribution.

**Usage**
gammadistribution(alpha = 1, beta = 1)

**Arguments**

- **alpha**: The first parameter of the Gamma distribution: The "shape" parameter.
- **beta**: The second parameter of the Gamma distribution: The "rate" parameter.

**Details**
The density of the distribution is proportional to

\[ f(x) = x^{\alpha-1} \exp(-\beta x) \]

**Value**
A Gamma probability distribution.

**Author(s)**
Petter Mostad <mostad@chalmers.se>

**See Also**

- expgamma

**Examples**
dist <- gammadistribution(4, 2)
plot(dist)
**invcdf**  

*Compute the Inverse Cumulative Distribution Function*

**Description**

Compute the inverse of the cumulative distribution function for a univariate probability distribution.

**Usage**

`invcdf(object, val)`

**Arguments**

- `object`  
  A univariate probability distribution.

- `val`  
  A value between 0 and 1.

**Value**

A value `v` such that the probability that `x ≤ v` is given by `val`.

**Author(s)**

Petter Mostad <mostad@chalmers.se>

**See Also**

cdf

**Examples**

```
invcdf(normal(), 0.975)
invcdf(binomialdistribution(10, 0.4), 0.5)
```

---

**leastsquares**  

*Find the Least Squares Solution in a Linear Model*

**Description**

Given a vector of data and a design matrix, the least squares estimates for a linear model is computed.

**Usage**

`leastsquares(data, design)`
Arguments

- `data`: A data vector.
- `design`: A design matrix. The number of rows must be equal to the length of the data vector.

Details

The fitted values represent the expected values all but the last variables in the posterior for the linear model.

Value

A vector of values of length equal to the number of columns in the design matrix.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

`linearmodel`, `fittedvalues`, `linearpredict`

Examples

```r
xdata <- simulate(uniformdistribution(), 14)
ydata <- xdata + 4 + simulate(normal(), 14)*0.1
plot(xdata,ydata)
design <- cbind(1, xdata)
leastsquares(ydata, design)
```

Description

Given a vector of data and a design matrix, describing how these data are thought to relate to some predictors in a linear model, the posterior for the parameters of this linear model is found, using a flat prior.

Usage

`linearmodel(data, design)`

Arguments

- `data`: A vector with data values.
- `design`: A design matrix. The number of rows must be equal to the length of the data vector. The number of columns corresponds to the number of explanatory variables.
Details
If $y_i$ is the i’th data value and $\beta_j$ is the j’th unknown parameter, and if $x_{ij}$ is the value in the i’th row and j’th column of the design matrix, then one assumes that $y_i$ is normally distributed with expectation

$$x_{i1}\beta_1 + x_{i2}\beta_2 + \ldots + x_{ik}\beta_k$$

and logged standard deviation $\lambda$. The computed probability distribution is then the posterior for the joint distribution of

$$(\beta_1, \beta_2, \ldots, \beta_k, \lambda)$$

Value
If $k$ is the number of columns in the design matrix and if $k > 1$, then the output is a multivariate Normal-ExpGamma distribution representing the posterior for the corresponding $k$ values and the logged scale parameter in the linear model. If $k = 1$, the output is a Normal-ExpGamma distribution representing the posterior.

Author(s)
Petter Mostad <mostad@chalmers.se>

See Also
fittedvalues, leastsquares, linearpredict

Examples
```r
data1 <- simulate(normal(3.3, log(2)), 9)
data2 <- simulate(normal(4.5, log(2)), 8)
data3 <- simulate(normal(2.9, log(2)), 7)
design <- designManyGroups(c(9,8,7))
posterior <- linearmodel(c(data1, data2, data3), design)
plot(posterior)
```

linearpredict Create a Linear Extension of a Distribution

Description
Extends the given probability distribution with new variables which are (multivariate) normally distributed with parameters based on the values of the given probability distribution and values given to the function.

Usage
linearpredict(object, ...)

Note
Arguments

- **object**: The probability distribution to be extended. Currently, it should be either a (multivariate) normal distribution, or a (multivariate) normal distribution extended with an extra parameter with either a Gamma or an ExpGamma distribution.

... A second optional argument may be given, which should then be a matrix with the same number of columns as there are normally distributed variables in the input object. If the matrix has only one row or column, it may be given as a vector. The default is a matrix with one row, consisting of only 1’s.

A third optional argument may also be given, which is then the precision matrix of the new normally distributed variables. By default, this matrix is the identity. If the input object contains a Gamma-distributed variable, its value is multiplied with the precision matrix. If the input object contains an ExpGamma-distributed variable $$y$$, the value $$e^{-2y}$$ is multiplied with the precision matrix.

Details

The input is either a (multivariate) variable $$x$$ with a normal distribution, or a joint distribution consisting of a Gamma- or ExpGamma-distributed variable $$y$$, and conditionally on this a (multivariate) normally distributed $$x$$. The output is a joint distribution for $$(z, x)$$ or $$(z, x, y)$$, where the marginal distribution for $$x$$ or $$(x, y)$$ is unchanged, while the conditional distribution for $$z$$ given $$x$$ or $$(x, y)$$ is (multivariate) normal. The expectation and precision for this conditional distribution is $$X \mu$$ and $$P \tau$$, respectively. Here, $$\mu$$ is the expectation of $$x$$, while $$X$$ is the optional second argument. The matrix $$P$$ is the optional third argument, while $$\tau$$ is either equal to $$y$$, when $$y$$ has a Gamma distribution, or equal to $$e^{-2y}$$, when $$y$$ has an ExpGamma distribution.

Value

A multivariate normal, multivariate Normal-Gamma, or multivariate Normal-ExpGamma distribution, depending on the input.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

- contrast

Examples

```r
prior <- normalgamma()
full <- linearpredict(prior, rep(1, 7))
data <- simulate(normal(), 7)
posterior <- conditional(full, 1:7, data)
plot(posterior)
```
Description

Given a multivariate distribution, one of its marginal distributions is computed.

Usage

marginal(object, v)

Arguments

object The multivariate probability distribution whose marginal should be computed.
v A vector of indices, indicating which parts of the object should be kept after marginalisation.

Details

The index or indices of the parameter(s) whose marginal distribution is computed is given in v.

Value

A probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

conditional

Examples

data <- simulate(normal(3, log(3)), 11)
posterior <- linearmodel(data, designOneGroup(11))
credibilityinterval(marginal(posterior, 1))
credibilityinterval(marginal(posterior, 2))
mdiscretedistribution  

Create Object Representing a Multivariate Discrete Distribution

Description

An object representing a multivariate discrete distribution is created, based on explicitly given possible values and probabilities for these.

Usage

mdiscretedistribution(probs, nms=NULL)

Arguments

probs  
This must be a matrix, or more generally an array with the same number of dimensions as the desired variable. The values in the matrix must be non-negative and represent the probabilities of the variable.

nms  
If given, nms should be a list with the same number of items as there are dimensions of probs. Each item in the list should be a vector with the names of the possible values of the variable representing this dimension of the multivariate variable. If not given, integers are used as variable names.

Value

A multivariate discrete probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

Examples

```
> dist <- mdiscretedistribution(array(1:24, c(2,3,4)))
> expectation(dist)
> variance(dist)
```

mnormal  

A Multivariate Normal Distribution

Description

Creates an object representing a multivariate normal distribution.

Usage

mnormal(expectation = c(0,0), P = diag(length(expectation)))
mnormalexpgamma

Arguments

- **expectation**: A vector of length at least 2 specifying the expectation of the distribution. By default, the vector (0,0).
- **P**: A matrix of size \( k \times k \), where \( k \) is the length of the expectation vector. \( P \) specifies the precision matrix, i.e., the inverse of the covariance matrix.

Details

If \( \mu \) is the expectation vector and \( P \) is the precision matrix, then the probability density function is proportional to

\[
f(x) = \exp(-0.5(x - \mu)^t P (x - \mu))
\]

Value

A multivariate normal probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

- normal

Examples

- plot(mnormal())
- plot(mnormal(c(1,2,3)))
- plot(mnormal(c(1,2), matrix(c(1, 0.5, 0.5, 1), 2, 2)))

mnormalexpgamma

A Multivariate Normal-ExpGamma Distribution

Description

Creates an object representing a multivariate Normal-ExpGamma distribution. If \((x, y)\) has a multivariate Normal-ExpGamma distribution, then the marginal distribution of \( y \) is an ExpGamma distribution, and the conditional distribution of \( x \) given \( y \) is multivariate normal.

Usage

mnormalexpgamma(mu=c(0,0), P, alpha, beta)
Arguments

- **mu**: The mu parameter. It must be a vector of length at least 2. The default value is (0,0).
- **P**: The P parameter.
- **alpha**: The alpha parameter.
- **beta**: The beta parameter.

Details

If \((x, y)\) has a multivariate Normal-ExpGamma distribution with parameters \(\mu, P, \alpha, \) and \(\beta\), then the marginal distribution of \(y\) has an ExpGamma distribution with parameters \(\alpha, \beta, \) and \(-2\), and conditionally on \(y\), \(x\) has a multivariate normal distribution with expectation \(\mu\) and precision matrix \(e^{-2y}P\). The probability density is proportional to

\[
    f(x, y) = \exp(-(2\alpha + k)y - e^{-2y}(\beta + (x - \mu)^t P(x - \mu)/2))
\]

where \(k\) is the dimension.

Value

A multivariate Normal-ExpGamma probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

- `gamma`, `normal`, `expgamma`, `normalgamma`, `normalexpgamma`, `mnormal`, `mnormalgamma`

Examples

```r
plot(mnormalexpgamma(alpha=3, beta=3))
```

---

**mnormalgamma**

A Multivariate Normal-Gamma Distribution

Description

Creates an object representing a multivariate Normal-Gamma distribution. If \((x, y)\) has a multivariate Normal-Gamma distribution, then the marginal distribution of \(y\) is an Gamma distribution, and the conditional distribution of \(x\) given \(y\) is multivariate normal.

Usage

```r
mnormalgamma(mu=c(0,0), P, alpha, beta)
```
Arguments

- **mu**
  The mu parameter. It must be a vector of length at least 2. The default value is (0,0).
- **P**
  The P parameter.
- **alpha**
  The alpha parameter.
- **beta**
  The beta parameter.

Details

If $(x, y)$ has a multivariate Normal-Gamma distribution with parameters $\mu$, $P$, $\alpha$, and $\beta$, then the marginal distribution of $y$ has a Gamma distribution with parameters $\alpha$, $\beta$, and conditionally on $y$, $x$ has a multivariate normal distribution with expectation $\mu$ and precision matrix $yP$. The probability density is proportional to

$$f(x, y) = y^{\alpha+k/2-1} \exp(-y(\beta + (x - \mu)^t P(x - \mu)/2))$$

where $k$ is the dimension.

Value

A multivariate Normal-Gamma probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

gamma, normal, expgamma, normalgamma, normalexpgamma, mnormal, mnormalexpgamma

Examples

plot(mnormalgamma(alpha=3, beta=3))

mtdistribution  A Multivariate t-Distribution

Description

Creates an object representing a multivariate non-centered t-distribution.

Usage

mtdistribution(expectation = c(0,0), degreesoffreedom = 10000, 
P = diag(length(expectation)))
Arguments

- **expectation**: A vector of length at least 2 specifying the expectation of the distribution. By default, the vector (0,0).
- **degreesoffreedom**: The degrees of freedom parameter.
- **P**: A matrix of size $k \times k$, where $k$ is the length of the expectation vector. $P$ plays a similar role in the multivariate t-distribution as the precision matrix does in the multivariate normal distribution. By default, $P$ is the identity matrix.

Details

If $\mu$ is the expectation, $\nu$ the degrees of freedom, $P$ is the last parameter, and $k$ the dimension, then the probability density function is proportional to

$$f(x) = \exp(\nu + (x - \mu)^t P (x - \mu))^{-(\nu+k)/2}$$

Value

A multivariate t-distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

tdistribution, mnormal

Examples

```r
plot(mtdistribution())
plot(mtdistribution(c(1,2,3), 3))
plot(mtdistribution(c(1,2), 3, matrix(c(1, 0.5, 0.5, 1), 2, 2)))
```

muniformdistribution  A Multivariate Uniform Distribution

Description

An object representing a multivariate univariate muniform distribution is created.

Usage

muniformdistribution(startvec, stopvec)
Arguments

startvec A vector with the lower bounds for the distribution.
stopvec A vector with the upper bounds for the distribution.

Value

A multivariate uniform probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

Examples

dist <- muniformdistribution(rep(0, 5), rep(1, 5))
expectation(dist)
variance(dist)

---

normal A Normal Distribution

Description

Create an object representing a univariate normal distribution.

Usage

normal(expectation = 0, lambda, P = 1)

Arguments

expectation The expectation of the distribution.
lambda THE NATURAL LOGARITHM OF THE STANDARD DEVIATION OF THE DISTRIBUTION. Thus, if the desired standard deviation is s, the second argument should be log(s). If the desired variance is v, the second argument should be log(v)/2. The default is a standard deviation of 1. An alternative to specifying this argument is to specify the precision parameter P.
P If given, this argument specifies the precision of the distribution, i.e., the inverse of the variance.

Value

A univariate normal probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>
See Also

mnormal

Examples

dist <- normal(3, log(0.7))
variance(dist)
dist <- normal(5, log(0.49)/2)
variance(dist)
dist <- normal(7, P = 2)
variance(dist)

Description

Creates an object representing a Normal-ExpGamma distribution. If \((x, y)\) has a Normal-ExpGamma distribution, then the marginal distribution of \(y\) is an ExpGamma distribution, and the conditional distribution of \(x\) given \(y\) is normal.

Usage

normalexpgamma(mu, kappa, alpha, beta)

Arguments

mu The \(mu\) parameter.
kappa The \(kappa\) parameter.
alpha The \(alpha\) parameter.
beta The \(beta\) parameter.

Details

If \((x, y)\) has a Normal-ExpGamma distribution with parameters \(\mu, \kappa, \alpha, \) and \(\beta\), then the marginal distribution of \(y\) has an ExpGamma distribution with parameters \(\alpha, \beta,\) and \(-2\), and conditionally on \(y, x\) has a normal distribution with expectation \(\mu\) and logged standard deviation \(\kappa + y\). The probability density is proportional to
\[
f(x, y) = \exp(-(2\alpha + 1)y - e^{-2y}(\beta + e^{-2\kappa(x - \mu)^2}/2))
\]

Value

A Normal-ExpGamma probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>
normalgamma

See Also
gamma, normal, expgamma, normalgamma, mnormal, mnormalgamma, normalexpgamma

Examples
plot(normalexpgamma(3,4,5,6))

normalgamma

A Normal-Gamma Distribution

Description
Creates an object representing a Normal-Gamma distribution. If \((x, y)\) has a Normal-Gamma distribution, then the marginal distribution of \(y\) is a Gamma distribution, and the conditional distribution of \(x\) given \(y\) is normal.

Usage
normalgamma(mu, kappa, alpha, beta)

Arguments
mu The \(mu\) parameter.
kappa The \(kappa\) parameter.
alp \(pha\) The \(alpha\) parameter.
beta The \(beta\) parameter.

Details
If \((x, y)\) has a Normal-Gamma distribution with parameters \(\mu, \kappa, \alpha, \beta\), then the marginal distribution of \(y\) has a Gamma distribution with parameters \(\alpha\) and \(\beta\), and conditionally on \(y\), \(x\) has a normal distribution with expectation \(\mu\) and logged standard deviation \(\kappa - log(y)/2\). The probability density is proportional to
\[
f(x, y) = y^{\alpha-0.5} \exp(-y(\beta + e^{-2\kappa(x - \mu)^2/2}))
\]

Value
A Normal-Gamma probability distribution.

Author(s)
Petter Mostad <mostad@chalmers.se>

See Also
gamma, normal, expgamma, normalexpgamma, mnormal, mnormalgamma, normalexpgamma
Examples

```
plot(normalgamma(3,4,5,6))
```

---

**p.value**  
*Compute the p-value for a Distribution*

**Description**

The p-value of a distribution is here interpreted as the probability outside the smallest credibility interval or region containing a point; if no point is explicitly given, it is assumed to be zero, or the origin.

**Usage**

```
p.value(object, point)
```

**Arguments**

- `object`  
The probability distribution for which the p-value should be computed.
- `point`  
The point which should be included in the credibility interval or region.

**Value**

The probability outside the smallest credibility interval or region containing the point.

**Author(s)**

Petter Mostad <mostad@chalmers.se>

**See Also**

- `credibilityinterval`

**Examples**

```
data <- simulate(normal(3, log(2)), 10)
posterior <- linearmodel(data, designOneGroup(10))
p.value(marginal(posterior, 1))
```
Description

A plot is constructed covering the central part of a probability distribution. The purpose is simply to illustrate the properties of the distribution.

Usage

```r
## S3 method for class 'normal'
plot(x, ...)
## S3 method for class 'binomialdistribution'
plot(x, ...)
```

Arguments

- `x`: A probability distribution.
- `...`: Other arguments (not currently in use).

Value

For univariate discrete distributions, a plot is generated showing with a histogram the probabilities of each of the possible values of the distribution. For univariate continuous distributions, a plot is made of roughly the central 99 of the distribution. For multivariate distributions, a combined plot is made, where one can find the marginal distributions along the diagonal, and contour plots for bivariate marginal distributions off the diagonal.

Author(s)

Petter Mostad <mostad@chalmers.se>

Examples

```r
plot(normal())
plot(mnormal(c(3,4,5), diag(3)))
plot(poissondistribution(3))
```
**poissondistribution**  
*A Poissondistribution*

**Description**  
Create an object representing a Poisson distribution.

**Usage**  
poissondistribution(rate)

**Arguments**  
rate  
The rate parameter of the distribution.

**Value**  
An object representing a Poisson distribution.

**Author(s)**  
Petter Mostad <mostad@chalmers.se>

**See Also**  
binomialdistribution

**Examples**  
dist <- poissondistribution(4)  
cdf(dist, 3)

---

**posteriornormal1**  
*Compute the Posterior Distribution for Parameters of One Normal Distribution*

**Description**  
Given a vector of data, this function computes the bivariate posterior for the expectation parameter and the logged scale parameter of a normal distribution, assuming that the data represents independent observations from the normal distribution. One assumes a flat prior.

**Usage**  
posteriornormal1(data)
posteriornormal2

Arguments

data A vector with data values.

Value

An object representing a bivariate Normal-ExpGamma distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

posteriornormal2, linearmodel

Examples

data <- simulate(normal(3.3, log(2)), 9)
posteriornormal1(data)
linearmodel(data, designOneGroup(length(data))) #Gives same result

posteriornormal2 Compute a Posterior Distribution for Parameters of Two Normal Distributions

Description

Given a vectors data1 and data2 of data, this function assumes data1 is a sample from one normal distribution while data2 is a sample from another, while both distributions are assumed to have the same logged scale. The bivariate posterior for the difference between the expectations of the two distributions and the common logged scale of the distributions is computed, assuming a flat prior.

Usage

posteriornormal2(data1, data2)

Arguments

data1 A vector with data values. Assumed to be a sample from the first normal distribution.

data2 Another vector with data values. Assumed to be a sample from the second normal distribution.

Value

An object representing a Normal-ExpGamma distribution.
**Description**

Compute the precision (i.e., the inverse of the variance) of a probability distribution.

**Usage**

```r
precision(object)
```

**Arguments**

- `object` : A probability distribution.

**Value**

The precision of the probability distribution: Either a number or a matrix.

**Examples**

```r
data1 <- simulate(normal(3.3, log(2)), 9)
data2 <- simulate(normal(5.7, log(2)), 4)
posteriornormal2(data1, data2)
marginal(linearmodel(c(data1, data2),
designTwoGroups(length(data1), length(data2))), 2:3) #Gives same result
```

```r
precision(normal(3, log(0.7)))
precision(binomialdistribution(7, 0.4))
```
Description

When a probability distribution is printed, its main features are listed.

Usage

```
## S3 method for class 'normal'
print(x, ...)  
```

Arguments

- `x` The object to be printed.
- `...` Other possible arguments (not currently implemented).

Value

Readable text describing the object.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

- `summary`

Examples

```
print(normal())  
```

---

probability

The Probability at a Value for a Discrete Distribution

Description

Given a possible value for a probability distribution, the probability at that value is computed.

Usage

```
probability(object, val)  
```
Arguments

- `object`: A discrete probability distribution.
- `val`: The value at which the probability should be computed.

Value

The probability at `val`.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

`probabilitydensity`

Examples

```r
probability(poissondistribution(3), 1)
probability(binomialedistribution(10, 0.24), 2)
```

---

**probabilitydensity**

The Probability Density at a Value for a Continuous Distribution

Description

Computes the probability density at a value for a continuous distribution.

Usage

```r
probabilitydensity(object, val, log = FALSE, normalize = TRUE)
```

Arguments

- `object`: A continuous probability distribution.
- `val`: The point at which the probability density should be computed.
- `log`: If TRUE, the logarithm of the probability density is returned.
- `normalize`: If FALSE, unnormalized densities are returned.

Value

The probability density at `val`.

Author(s)

Petter Mostad <mostad@chalmers.se>
simulate.normal

Examples

probabilitydensity(normal(), 1)
probabilitydensity(mnormal(c(0,0), diag(2)), c(1,1))

simulate.normal Simulate values from a Probability Distribution

Description

Simulate independent values from a given probability distribution.

Usage

## S3 method for class 'normal'
simulate(object, nsim = 1, ...)
## S3 method for class 'binomialdistribution'
simulate(object, nsim = 1, ...)
## S3 method for class 'discretedistribution'
simulate(object, nsim = 1, ...)
## S3 method for class 'expgamma'
simulate(object, nsim = 1, ...)
## S3 method for class 'fdistribution'
simulate(object, nsim = 1, ...)
## S3 method for class 'gammadistribution'
simulate(object, nsim = 1, ...)
## S3 method for class 'mnormalexpgamma'
simulate(object, nsim = 1, ...)
## S3 method for class 'mnormalgamma'
simulate(object, nsim = 1, ...)
## S3 method for class 'mnormal'
simulate(object, nsim = 1, ...)
## S3 method for class 'mtdistribution'
simulate(object, nsim = 1, ...)
## S3 method for class 'normalexpgamma'
simulate(object, nsim = 1, ...)
## S3 method for class 'normalgamma'
simulate(object, nsim = 1, ...)
## S3 method for class 'poissondistribution'
simulate(object, nsim = 1, ...)
## S3 method for class 'tdistribution'
simulate(object, nsim = 1, ...)
## S3 method for class 'uniformdistribution'
simulate(object, nsim = 1, ...)
Arguments

object  The probability distribution to be simulated from.
nsim   The number of simulated values. Default is 1.
...    Additional parameters. Currently not in use.

Value

For univariate distributions, a vector of length nsim is produced. For multivariate distributions, a matrix with nsim rows is produced.

Author(s)

Petter Mostad <mostad@chalmers.se>

Examples

simulate(normal())
simulate(normal(), 10)
simulate(mnormal(), 10)

summary.normal  Summary of a Probability Distribution Object

Description

Lists the main features of a probability distribution object.

Usage

## S3 method for class 'normal'
summary(object, ...)

Arguments

object  The object to be summarized.
...    Other possible arguments (not currently implemented).

Value

Readable text describing the object.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

print
tdistribution

Examples

summary(normal())

distribution A t-distribution

Description

Create an object representing a univariate non-centered t-distribution.

Usage

tdistribution(expectation = 0, degreesoffreedom = 1e+20,
lambda, P = 1)

Arguments

expectation The expectation of the distribution.

degreesoffreedom The degrees of freedom parameter.

lambda The natural logarithm of the scale $\sigma$ of the distribution. The standard t-distribution has scale 1, and the default for lambda is log(1) = 0.

P An alternative to specifying the logged scale log($\sigma$) with lambda is to specify P: It is defined as $P = 1/\sigma^2$.

Details

The probability density of a t-distribution with expectation $\mu$, degrees of freedom $\nu$, and logged scale $\lambda$ is proportional to

$$f(x) = (\nu + e^{-2\lambda}(x - \mu)^2)^{-(\nu+1)/2}$$

Value

A t-distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

mtdistribution

Examples

dist <- tdistribution(3)
plot(dist)
uniformdistribution A Uniform Distribution

Description
An object representing a univariate uniform distribution is created.

Usage
uniformdistribution(a = 0, b = 1)

Arguments
a The lower bound for the distribution. The default is 0.
b The upper bound for the distribution. The default is 1.

Value
A uniform probability distribution.

Author(s)
Petter Mostad <mostad@chalmers.se>

Examples
dist <- uniformdistribution()
expectation(dist)
variance(dist)

variance The Variance of a Distribution

Description
Compute the variance of a probability distribution.

Usage
variance(object)

Arguments
object A probability distribution.
variance

Value

The variance of the probability distribution.

Author(s)

Petter Mostad <mostad@chalmers.se>

See Also

expectation, variance

Examples

variance(normal(3, log(0.7)))
variance(binomialdistribution(7, 0.4))
Index

* ANOVA  
anovatable, 3  
* Conditional distribution  
conditional, 9  
* Contrast  
contrast, 10  
* Credibility Interval  
credibilityinterval, 10  
* Cumulative distribution function  
cdf, 7  
* Design Matrix  
designBalanced, 11  
  designFactorial, 12  
* F distribution  
fdistribution, 18  
* One-sample t test  
designOneGroup, 14  
* One-way ANOVA  
designManyGroups, 13  
* Probability density  
probabilitydensity, 40  
* Two-sample t test  
designTwoGroups, 14  
* compose  
  compose, 8  
* distribution  
betabinomial, 4  
  binomialbeta, 6  
  binomialdistribution, 7  
* probabilitydistribution  
betabinomial, 4  
  binomialbeta, 6  
  binomialdistribution, 7  
  anovatable, 3  
betabinomial, 4, 5, 6  
  betadistribution, 5, 5, 6  
  binomialbeta, 5, 6  
  binomialdistribution, 5, 6, 7, 36  
  cdf, 7, 21  
  compose, 8  
  conditional, 9, 25  
  contrast, 10, 24  
  credibilityinterval, 10, 34  
  designBalanced, 11, 12–15  
  designFactorial, 12, 12, 13–15  
  designManyGroups, 12, 13, 14, 15  
  designOneGroup, 12, 13, 14, 15  
  designTwoGroups, 12–14, 14  
  difference, 15  
  discretedistribution, 16  
  expectation, 17, 38, 45  
  expgamma, 17, 20, 28, 29, 33  
  fdistribution, 18  
  fittedvalues, 19, 22, 23  
  gamma, 28, 29, 33  
  gammadistribution, 18, 20  
  invcdf, 8, 21  
  leastsquares, 19, 21, 23  
  lestat (lestat-package), 2  
  lestat-package, 2  
  linearmodel, 19, 22, 22, 37, 38  
  linearpredict, 19, 22, 23, 23  
  marginal, 9, 25  
  mdiscretedistribution, 26  
  mnormal, 26, 28–30, 32, 33  
  mnormalexpgamma, 27, 29, 33  
  mnormalgamma, 28, 28, 33  
  mtdistribution, 29, 43  
  muniformdistribution, 30  
  normal, 27–29, 31, 33  
  normalexpgamma, 28, 29, 32, 33
normalgamma, 28, 29, 33, 33
p.value, 11, 34
plot.betabinomial (plot.normal), 35
plot.betadistribution (plot.normal), 35
plot.binomialbeta (plot.normal), 35
plot.binomialdistribution
(plot.normal), 35
plot.discretedistribution
(plot.normal), 35
plot.expgamma (plot.normal), 35
plot.fdistribution (plot.normal), 35
plot.gammadistribution
(plot.normal), 35
plot.mdiscretedistribution
(plot.normal), 35
plot.mnormal (plot.normal), 35
plot.mnormalexpgamma (plot.normal), 35
plot.mnormalgamma (plot.normal), 35
plot.mtdistribution (plot.normal), 35
plot.muniformdistribution
(plot.normal), 35
plot.normal (plot.normal), 35
plot.normalexpgamma (plot.normal), 35
plot.normalgamma (plot.normal), 35
plot.poissondistribution
(plot.normal), 35
plot.tdistribution (plot.normal), 35
plot.uniformdistribution
(plot.normal), 35
probability, 39
probabilitydensity, 40, 40
simulate.betabinomial
(simulate.normal), 41
simulate.betadistribution
(simulate.normal), 41
simulate.binomialbeta
(simulate.normal), 41
simulate.binomialdistribution
(simulate.normal), 41
simulate.discretedistribution
(simulate.normal), 41
simulate.expgamma (simulate.normal), 41
simulate.fdistribution
(simulate.normal), 41
simulate.gammadistribution
(simulate.normal), 41
simulate.mdiscretedistribution
(simulate.normal), 41
simulate.mnormal (simulate.normal), 41
simulate.mnormalexpgamma
(simulate.normal), 41
simulate.mnormalgamma
(simulate.normal), 41
simulate.mtdistribution
(simulate.normal), 41
simulate.muniformdistribution
(simulate.normal), 41
simulate.normal (simulate.normal), 41
simulate.normalexpgamma
(simulate.normal), 41
simulate.normalgamma (simulate.normal), 41
simulate.poissondistribution
(simulate.normal), 41
simulate.tdistribution
   (simulate.normal), 41
simulate.uniformdistribution
   (simulate.normal), 41
summary, 39
summary.betabinomial (summary.normal), 42
summary.betadistribution
   (summary.normal), 42
summary.binomialbeta (summary.normal), 42
summary.binomialdistribution
   (summary.normal), 42
summary.discretedistribution
   (summary.normal), 42
summary.expgamma (summary.normal), 42
summary.fdistribution (summary.normal), 42
summary.gammadistribution
   (summary.normal), 42
summary.mdiscretedistribution
   (summary.normal), 42
summary.mnormal (summary.normal), 42
summary.mnormalexpgamma
   (summary.normal), 42
summary.mnormalgamma (summary.normal), 42
summary.mtdistribution
   (summary.normal), 42
summary.muniformdistribution
   (summary.normal), 42
summary.normal, 42
summary.normalexpgamma
   (summary.normal), 42
summary.normalgamma (summary.normal), 42
summary.poissondistribution
   (summary.normal), 42
summary.tdistribution
   (summary.normal), 42
summary.uniformdistribution
   (summary.normal), 42
tdistribution, 30, 43
uniformdistribution, 44
variance, 17, 38, 44, 45