

# Package ‘estimators’

May 8, 2026

**Title** Parameter Estimation

**Version** 0.8.5

**Description** Implements estimation methods for parameters of common distribution families. The common `d`, `p`, `q`, `r` function family for each distribution is enriched with the `ll`, `e`, and `v` counterparts, computing the log-likelihood, performing estimation, and calculating the asymptotic variance - covariance matrix, respectively. Parameter estimation is performed analytically whenever possible.

**License** GPL (>= 3)

**URL** <https://thechibo.github.io/estimators/>

**BugReports** <https://github.com/thechibo/estimators/issues>

**Depends** R (>= 4.0.0)

**Imports** extraDistr, ggh4x, ggplot2, grDevices, Matrix, methods, progress, stats, utils

**Suggests** covr, knitr, rmarkdown, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Encoding** UTF-8

**Language** en-US

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Ioannis Oikonomidis [aut, cre] (ORCID:  
<<https://orcid.org/0000-0001-8130-2104>>),  
Samis Trevezas [aut, ths] (ORCID:  
<<https://orcid.org/0000-0003-2262-8299>>)

**Maintainer** Ioannis Oikonomidis <goikon@math.uoa.gr>

**Repository** CRAN

**Date/Publication** 2024-05-16 15:00:10 UTC

## Contents

avar	3
avar_me	5
avar_mle	6
avar_same	8
Bern	9
Beta	11
Binom	13
calculus	15
Cat	16
Cauchy	18
Chisq	19
Dir	20
Distributions	22
dpqr	22
estim	23
Exp	25
Fisher	27
Gam	28
Geom	30
idigamma	32
Laplace	33
large_metrics	35
ll	36
Lnorm	37
me	39
mle	41
moments	42
Multinom	44
Nbinom	45
Norm	47
plot_large_metrics	49
plot_small_metrics	50
Pois	51
same	53
small_metrics	55
Stud	56
Unif	58
Weib	59

## Index

62

---

avar

*Asymptotic Variance*

---

### Description

Calculates the asymptotic variance (or variance - covariance matrix in the multidimensional case) of an estimator, given a specified family of distributions and the true parameter values.

### Usage

```
avar(distr, type, ...)  
  
vbern(prob, type = "mle")  
  
vbinom(size, prob, type = "mle")  
  
vcat(prob, type = "mle")  
  
vdirichlet(alpha, type = "mle")  
  
vexp(rate, type = "mle")  
  
vgamma(shape, scale, type = "mle")  
  
vgeom(prob, type = "mle")  
  
vlaplace(mu, sigma, type = "mle")  
  
vmultinom(size, prob, type = "mle")  
  
vnbinom(size, prob, type = "mle")  
  
vnorm(mean, sd, type = "mle")  
  
vpois(lambda, type = "mle")  
  
vweib(shape, scale, type = "mle")
```

### Arguments

distr	A subclass of Distribution. The distribution family assumed.
type	character, case ignored. The estimator type (mle, me, or same).
...	extra arguments.
alpha, mu, sigma, size, prob, shape, rate, scale, mean, sd, lambda	numeric. Distribution parameters.

**Value**

A named matrix. The asymptotic covariance matrix of the estimator.

**References**

Ye, Z.-S. & Chen, N. (2017), Closed-form estimators for the gamma distribution derived from likelihood equations, *The American Statistician* 71(2), 177–181.

Van der Vaart, A. W. (2000), *Asymptotic statistics*, Vol. 3, Cambridge university press.

Tamae, H., Irie, K. & Kubokawa, T. (2020), A score-adjusted approach to closed-form estimators for the gamma and beta distributions, *Japanese Journal of Statistics and Data Science* 3, 543–561.

Mathal, A. & Moschopoulos, P. (1992), A form of multivariate gamma distribution, *Annals of the Institute of Statistical Mathematics* 44, 97–106.

Oikonomidis, I. & Trevezas, S. (2023), Moment-Type Estimators for the Dirichlet and the Multivariate Gamma Distributions, arXiv, <https://arxiv.org/abs/2311.15025>

**See Also**

[avar\\_mle](#), [avar\\_me](#), [avar\\_same](#)

**Examples**

```
# -----
# Beta Distribution Example
# -----

# Simulation
set.seed(1)
shape1 <- 1
shape2 <- 2
D <- Beta(shape1, shape2)
x <- r(D)(100)

# Likelihood - The ll Functions

llbeta(x, shape1, shape2)
ll(x, c(shape1, shape2), D)
ll(x, c(shape1, shape2), "beta")

# Point Estimation - The e Functions

ebeta(x, type = "mle")
ebeta(x, type = "me")
ebeta(x, type = "same")

mle(x, D)
me(x, D)
same(x, D)

estim(x, D, type = "mle")
```

```
# Asymptotic Variance - The v Functions

vbeta(shape1, shape2, type = "mle")
vbeta(shape1, shape2, type = "me")
vbeta(shape1, shape2, type = "same")

avar_mle(D)
avar_me(D)
avar_same(D)

avar(D, type = "mle")
```

---

avar_me	<i>ME Asymptotic Variance</i>
---------	-------------------------------

---

### Description

Calculates the asymptotic variance (or variance - covariance matrix in the multidimensional case) of the ME, given a specified family of distributions and the true parameter values.

### Usage

```
avar_me(distr, ...)
```

### Arguments

distr	A subclass of Distribution. The distribution family assumed.
...	extra arguments.

### Value

A named matrix. The asymptotic covariance matrix of the estimator.

### References

- Ye, Z.-S. & Chen, N. (2017), Closed-form estimators for the gamma distribution derived from likelihood equations, *The American Statistician* 71(2), 177–181.
- Van der Vaart, A. W. (2000), *Asymptotic statistics*, Vol. 3, Cambridge university press.
- Tamae, H., Irie, K. & Kubokawa, T. (2020), A score-adjusted approach to closed-form estimators for the gamma and beta distributions, *Japanese Journal of Statistics and Data Science* 3, 543–561.
- Mathal, A. & Moschopoulos, P. (1992), A form of multivariate gamma distribution, *Annals of the Institute of Statistical Mathematics* 44, 97–106.
- Oikonomidis, I. & Trevezas, S. (2023), Moment-Type Estimators for the Dirichlet and the Multivariate Gamma Distributions, arXiv, <https://arxiv.org/abs/2311.15025>

### See Also

[avar](#), [avar\\_mle](#), [avar\\_same](#)

**Examples**

```

# -----
# Beta Distribution Example
# -----

# Simulation
set.seed(1)
shape1 <- 1
shape2 <- 2
D <- Beta(shape1, shape2)
x <- r(D)(100)

# Likelihood - The ll Functions

llbeta(x, shape1, shape2)
ll(x, c(shape1, shape2), D)
ll(x, c(shape1, shape2), "beta")

# Point Estimation - The e Functions

ebeta(x, type = "mle")
ebeta(x, type = "me")
ebeta(x, type = "same")

mle(x, D)
me(x, D)
same(x, D)

estim(x, D, type = "mle")

# Asymptotic Variance - The v Functions

vbeta(shape1, shape2, type = "mle")
vbeta(shape1, shape2, type = "me")
vbeta(shape1, shape2, type = "same")

avar_mle(D)
avar_me(D)
avar_same(D)

avar(D, type = "mle")

```

---

avar\_mle

*MLE Asymptotic Variance*


---

**Description**

Calculates the asymptotic variance (or variance - covariance matrix in the multidimensional case) of the MLE, given a specified family of distributions and the true parameter values.

**Usage**

```
avar_mle(distr, ...)
```

**Arguments**

```
distr      A subclass of Distribution. The distribution family assumed.
...        extra arguments.
```

**Value**

A named matrix. The asymptotic covariance matrix of the estimator.

**See Also**

[avar](#), [avar\\_me](#), [avar\\_same](#)

**Examples**

```
# -----
# Beta Distribution Example
# -----

# Simulation
set.seed(1)
shape1 <- 1
shape2 <- 2
D <- Beta(shape1, shape2)
x <- r(D)(100)

# Likelihood - The ll Functions

llbeta(x, shape1, shape2)
ll(x, c(shape1, shape2), D)
ll(x, c(shape1, shape2), "beta")

# Point Estimation - The e Functions

ebeta(x, type = "mle")
ebeta(x, type = "me")
ebeta(x, type = "same")

mle(x, D)
me(x, D)
same(x, D)

estim(x, D, type = "mle")

# Asymptotic Variance - The v Functions

vbeta(shape1, shape2, type = "mle")
vbeta(shape1, shape2, type = "me")
```

```

vbeta(shape1, shape2, type = "same")

avar_mle(D)
avar_me(D)
avar_same(D)

avar(D, type = "mle")

```

---

 avar\_same

*SAME Asymptotic Variance*


---

### Description

Calculates the asymptotic variance (or variance - covariance matrix in the multidimensional case) of the SAME, given a specified family of distributions and the true parameter values.

### Usage

```
avar_same(distr, ...)
```

### Arguments

distr	A subclass of <code>Distribution</code> . The distribution family assumed.
...	extra arguments.

### Value

A named matrix. The asymptotic covariance matrix of the estimator.

### References

- Ye, Z.-S. & Chen, N. (2017), Closed-form estimators for the gamma distribution derived from likelihood equations, *The American Statistician* 71(2), 177–181.
- Van der Vaart, A. W. (2000), *Asymptotic statistics*, Vol. 3, Cambridge university press.
- Tamae, H., Irie, K. & Kubokawa, T. (2020), A score-adjusted approach to closed-form estimators for the gamma and beta distributions, *Japanese Journal of Statistics and Data Science* 3, 543–561.
- Mathal, A. & Moschopoulos, P. (1992), A form of multivariate gamma distribution, *Annals of the Institute of Statistical Mathematics* 44, 97–106.
- Oikonomidis, I. & Trevezas, S. (2023), Moment-Type Estimators for the Dirichlet and the Multivariate Gamma Distributions, arXiv, <https://arxiv.org/abs/2311.15025>

### See Also

[avar](#), [avar\\_mle](#), [avar\\_me](#)

**Examples**

```

# -----
# Beta Distribution Example
# -----

# Simulation
set.seed(1)
shape1 <- 1
shape2 <- 2
D <- Beta(shape1, shape2)
x <- r(D)(100)

# Likelihood - The ll Functions

llbeta(x, shape1, shape2)
ll(x, c(shape1, shape2), D)
ll(x, c(shape1, shape2), "beta")

# Point Estimation - The e Functions

ebeta(x, type = "mle")
ebeta(x, type = "me")
ebeta(x, type = "same")

mle(x, D)
me(x, D)
same(x, D)

estim(x, D, type = "mle")

# Asymptotic Variance - The v Functions

vbeta(shape1, shape2, type = "mle")
vbeta(shape1, shape2, type = "me")
vbeta(shape1, shape2, type = "same")

avar_mle(D)
avar_me(D)
avar_same(D)

avar(D, type = "mle")

```

---

Bern

*Bernoulli Distribution*


---

**Description**

Bernoulli Distribution

**Usage**

```
Bern(prob = 0.5)

## S4 method for signature 'Bern'
d(x)

## S4 method for signature 'Bern'
p(x)

## S4 method for signature 'Bern'
qn(x)

## S4 method for signature 'Bern'
r(x)

## S4 method for signature 'Bern'
mean(x)

## S4 method for signature 'Bern'
median(x)

## S4 method for signature 'Bern'
mode(x)

## S4 method for signature 'Bern'
var(x)

## S4 method for signature 'Bern'
sd(x)

## S4 method for signature 'Bern'
skew(x)

## S4 method for signature 'Bern'
kurt(x)

## S4 method for signature 'Bern'
entro(x)

## S4 method for signature 'Bern'
finf(x)

## S4 method for signature 'numeric,numeric,Bern'
ll(x, prm, distr)

## S4 method for signature 'numeric,Bern'
mle(x, distr)
```

```
## S4 method for signature 'numeric,Bern'
me(x, distr)

## S4 method for signature 'Bern'
avar_mle(distr)

## S4 method for signature 'Bern'
avar_me(distr)
```

### Arguments

<code>x</code>	an object of class <code>Bern</code> . If the function also has a <code>distr</code> argument, <code>x</code> is a numeric vector, a sample of observations.
<code>prm, prob</code>	numeric. The distribution parameter.
<code>distr</code>	an object of class <code>Bern</code> .

### Value

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on `Distribution` objects returns a `Distribution` object of the appropriate class and with the appropriate parameters.

### See Also

[dpqr, moments](#)

---

Beta

*Beta Distribution*

---

### Description

Beta Distribution

### Usage

```
Beta(shape1 = 1, shape2 = 1, ncp = 0)

## S4 method for signature 'Beta'
d(x)

## S4 method for signature 'Beta'
p(x)
```

```
## S4 method for signature 'Beta'  
qn(x)  
  
## S4 method for signature 'Beta'  
r(x)  
  
## S4 method for signature 'Beta'  
mean(x)  
  
## S4 method for signature 'Beta'  
median(x)  
  
## S4 method for signature 'Beta'  
mode(x)  
  
## S4 method for signature 'Beta'  
var(x)  
  
## S4 method for signature 'Beta'  
sd(x)  
  
## S4 method for signature 'Beta'  
skew(x)  
  
## S4 method for signature 'Beta'  
kurt(x)  
  
## S4 method for signature 'Beta'  
entro(x)  
  
## S4 method for signature 'Beta'  
finf(x)  
  
llbeta(x, shape1, shape2)  
  
## S4 method for signature 'numeric,numeric,Beta'  
ll(x, prm, distr)  
  
## S4 method for signature 'numeric,Beta'  
mle(x, distr, par0 = "same", method = "L-BFGS-B", lower = 1e-05, upper = Inf)  
  
## S4 method for signature 'numeric,Beta'  
me(x, distr)  
  
## S4 method for signature 'numeric,Beta'  
same(x, distr)
```

```

vbeta(shape1, shape2, type = "mle")

## S4 method for signature 'Beta'
avar_mle(distr)

## S4 method for signature 'Beta'
avar_me(distr)

## S4 method for signature 'Beta'
avar_same(distr)

```

### Arguments

shape1, shape2, ncp	numeric. The distribution parameters.
x	an object of class Beta. If the function also has a <code>distr</code> argument, x is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class Beta.
par0, method, lower, upper	arguments passed to <code>optim</code> .
type	character, case ignored. The estimator type (mle, me, or same).

### Value

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

 Binom

*Binomial Distribution*


---

### Description

Binomial Distribution

**Usage**

```
Binom(size = 1, prob = 0.5)

## S4 method for signature 'Binom'
d(x)

## S4 method for signature 'Binom'
p(x)

## S4 method for signature 'Binom'
qn(x)

## S4 method for signature 'Binom'
r(x)

## S4 method for signature 'Binom'
mean(x)

## S4 method for signature 'Binom'
var(x)

## S4 method for signature 'Binom'
sd(x)

## S4 method for signature 'Binom'
skew(x)

## S4 method for signature 'Binom'
kurt(x)

## S4 method for signature 'Binom'
finf(x)

## S4 method for signature 'numeric,numeric,Binom'
ll(x, prm, distr)

## S4 method for signature 'numeric,Binom'
mle(x, distr)

## S4 method for signature 'numeric,Binom'
me(x, distr)

## S4 method for signature 'Binom'
avar_mle(distr)

## S4 method for signature 'Binom'
avar_me(distr)
```

**Arguments**

size, prob	numeric. The distribution parameters.
x	an object of class Binom. If the function also has a distr argument, x is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class Binom.

**Value**

The dpqr family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The ll function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The estim family of functions return the estimated parameters of the distribution, given a sample. The avar family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

calculus

*Distribution Calculus*


---

**Description**

Distribution Calculus

**Usage**

```
## S4 method for signature 'Norm, Norm'
e1 + e2

## S4 method for signature 'numeric, Norm'
e1 + e2

## S4 method for signature 'Norm, numeric'
e1 + e2

## S4 method for signature 'Norm, Norm'
e1 - e2

## S4 method for signature 'numeric, Norm'
e1 - e2

## S4 method for signature 'Norm, numeric'
e1 - e2

## S4 method for signature 'numeric, Norm'
```

```

e1 * e2

## S4 method for signature 'Norm,numeric'
e1 * e2

## S4 method for signature 'Norm,numeric'
e1 / e2

## S4 method for signature 'Norm,logical'
sum(x, ..., na.rm = FALSE)

## S4 method for signature 'Norm'
exp(x)

```

### Arguments

x, e1, e2	objects of subclass Distribution.
...	extra arguments.
na.rm	logical. Should missing values be removed?

### Value

All calculations return Distribution objects (specifically, objects of a class that is a subclass of Distribution), accordingly to the property at hand.

### Examples

```

# -----
# Distribution Calculus Example
# -----

library(estimators)

# Normal location - scale transformation
x <- Norm(mean = 2, sd = 3)
y <- 3 * x + 1 # Norm(mean = 7, sd = 9)

# Addition of two independent Normal random variables
x1 <- Norm(mean = 1, sd = 3)
x2 <- Norm(mean = 2, sd = 4)
x3 <- x1 + x2 # Norm(mean = 3, sd = 5)

```

---

Cat

*Categorical Distribution*

---

### Description

Categorical Distribution

**Usage**

```

Cat(prob = c(0.5, 0.5))

## S4 method for signature 'Cat'
d(x)

## S4 method for signature 'Cat'
r(x)

## S4 method for signature 'Cat'
mean(x)

## S4 method for signature 'Cat'
var(x)

## S4 method for signature 'Cat'
finf(x)

## S4 method for signature 'numeric,numeric,Cat'
ll(x, prm, distr)

## S4 method for signature 'numeric,Cat'
mle(x, distr)

## S4 method for signature 'numeric,Cat'
me(x, distr)

## S4 method for signature 'Cat'
avar_mle(distr)

## S4 method for signature 'Cat'
avar_me(distr)

```

**Arguments**

prob	numeric. The distribution parameters.
x	an object of class Cat. If the function also has a distr argument, x is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class Cat.

**Value**

The dpqr family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The ll function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The estim family of functions return

the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

 Cauchy

*Cauchy Distribution*


---

### Description

Cauchy Distribution

### Usage

```
Cauchy(location = 1, scale = 1)
```

```
## S4 method for signature 'Cauchy'
d(x)
```

```
## S4 method for signature 'Cauchy'
p(x)
```

```
## S4 method for signature 'Cauchy'
qn(x)
```

```
## S4 method for signature 'Cauchy'
r(x)
```

```
## S4 method for signature 'Cauchy'
median(x)
```

```
## S4 method for signature 'Cauchy'
mode(x)
```

```
## S4 method for signature 'Cauchy'
entro(x)
```

```
## S4 method for signature 'Cauchy'
finf(x)
```

```
## S4 method for signature 'numeric,numeric,Cauchy'
ll(x, prm, distr)
```

### Arguments

`location`, `scale` numeric. The distribution parameters.

x	an object of class Cauchy. If the function also has a <code>distr</code> argument, x is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class Cauchy.

**Value**

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The `moments` family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

Chisq	<i>Chi-Square Distribution</i>
-------	--------------------------------

---

**Description**

Chi-Square Distribution

**Usage**

```
Chisq(df = 1, ncp = 0)

## S4 method for signature 'Chisq'
d(x)

## S4 method for signature 'Chisq'
p(x)

## S4 method for signature 'Chisq'
qn(x)

## S4 method for signature 'Chisq'
r(x)

## S4 method for signature 'Chisq'
mean(x)

## S4 method for signature 'Chisq'
var(x)

## S4 method for signature 'Chisq'
```

```
sd(x)

## S4 method for signature 'Chisq'
skew(x)

## S4 method for signature 'Chisq'
kurt(x)
```

### Arguments

df, ncp	numeric. The distribution parameters.
x	an object of class Chisq. If the function also has a distr argument, x is a numeric vector, a sample of observations.

### Value

The dpqr family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The ll function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The estim family of functions return the estimated parameters of the distribution, given a sample. The avar family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

 Dir

---

*Dirichlet Distribution*


---

### Description

Dirichlet Distribution

### Usage

```
Dir(alpha = c(1, 1))

## S4 method for signature 'Dir'
d(x)

## S4 method for signature 'Dir'
r(x)

## S4 method for signature 'Dir'
mean(x)

## S4 method for signature 'Dir'
mode(x)
```

```

## S4 method for signature 'Dir'
var(x)

## S4 method for signature 'Dir'
entro(x)

## S4 method for signature 'Dir'
finf(x)

## S4 method for signature 'matrix,numeric,Dir'
ll(x, prm, distr)

## S4 method for signature 'matrix,Dir'
mle(x, distr, par0 = "same", method = "L-BFGS-B", lower = 1e-05, upper = Inf)

## S4 method for signature 'matrix,Dir'
me(x, distr)

## S4 method for signature 'matrix,Dir'
same(x, distr)

## S4 method for signature 'Dir'
avar_mle(distr)

## S4 method for signature 'Dir'
avar_me(distr)

## S4 method for signature 'Dir'
avar_same(distr)

```

### Arguments

alpha	numeric. The distribution parameters.
x	an object of class <code>Dir</code> . If the function also has a <code>distr</code> argument, <code>x</code> is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class <code>Dir</code> .
par0, method, lower, upper	arguments passed to <code>optim</code> .

### Value

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return

the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

Distributions

*Distribution S4 Classes*

---

### Description

A collection of classes that provide a flexible and structured way to work with probability distributions.

### Value

The dpqr family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The ll function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The estim family of functions return the estimated parameters of the distribution, given a sample. The avar family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

dpqr

*The d p q r Functions*

---

### Description

Four generic functions that take a distribution object (e.g. Bern) and return the density, cumulative probability, quantile, and random generator functions, respectively.

### Usage

`d(x, ...)`

`p(x, ...)`

`qn(x, ...)`

`r(x, ...)`

### Arguments

`x` an object of subclass Distribution.

`...` extra arguments.

**Value**

The `d` `p` `q` `r` functions return the density, cumulative probability, quantile, and random generator functions, respectively.

**Examples**

```
# -----  
# Beta Distribution Example  
# -----  
  
library(estimators)  
  
# Create the distribution  
x <- Beta(3, 5)  
  
# Density function  
df <- d(x)  
df(c(0.3, 0.8, 0.5))  
  
# Probability function  
pf <- p(x)  
pf(c(0.3, 0.8, 0.5))  
  
# Density function  
qf <- qn(x)  
qf(c(0.3, 0.8, 0.5))  
  
# Random Generator function  
rf <- r(x)  
rf(5)
```

---

estim

*Parameter Estimation*

---

**Description**

Estimates the parameters of a random sample according to a specified family of distributions.

**Usage**

```
estim(x, distr, type = "mle", ...)
```

```
ebern(x, type = "mle", ...)
```

```
ebeta(x, type = "mle", ...)
```

```
ebinom(x, type = "mle", ...)
```

```
ecat(x, type = "mle", ...)
```

```

edirichlet(x, type = "mle", ...)
eexp(x, type = "mle", ...)
egamma(x, type = "mle", ...)
egeom(x, type = "mle", ...)
elaplace(x, type = "mle", ...)
elnorm(x, type = "mle", ...)
emultinom(x, type = "mle", ...)
enbinom(x, type = "mle", ...)
enorm(x, type = "mle", ...)
epois(x, type = "mle", ...)
eunif(x, type = "mle", ...)
eweib(x, type = "mle", ...)

```

### Arguments

<code>x</code>	numeric. A sample under estimation.
<code>distr</code>	A subclass of <code>Distribution</code> . The distribution family assumed.
<code>type</code>	character, case ignored. The estimator type (mle, me, or same).
<code>...</code>	extra arguments.

### Value

numeric. The estimator produced by the sample.

### References

- Ye, Z.-S. & Chen, N. (2017), Closed-form estimators for the gamma distribution derived from likelihood equations, *The American Statistician* 71(2), 177–181.
- Van der Vaart, A. W. (2000), *Asymptotic statistics*, Vol. 3, Cambridge university press.
- Tamae, H., Irie, K. & Kubokawa, T. (2020), A score-adjusted approach to closed-form estimators for the gamma and beta distributions, *Japanese Journal of Statistics and Data Science* 3, 543–561.
- Mathal, A. & Moschopoulos, P. (1992), A form of multivariate gamma distribution, *Annals of the Institute of Statistical Mathematics* 44, 97–106.
- Oikonomidis, I. & Trevezas, S. (2023), Moment-Type Estimators for the Dirichlet and the Multivariate Gamma Distributions, arXiv, <https://arxiv.org/abs/2311.15025>

**See Also**

[mle](#), [me](#), [same](#)

**Examples**

```
# -----  
# Beta Distribution Example  
# -----  
  
# Simulation  
set.seed(1)  
shape1 <- 1  
shape2 <- 2  
D <- Beta(shape1, shape2)  
x <- r(D)(100)  
  
# Likelihood - The ll Functions  
  
llbeta(x, shape1, shape2)  
ll(x, c(shape1, shape2), D)  
ll(x, c(shape1, shape2), "beta")  
  
# Point Estimation - The e Functions  
  
ebeta(x, type = "mle")  
ebeta(x, type = "me")  
ebeta(x, type = "same")  
  
mle(x, D)  
me(x, D)  
same(x, D)  
  
estim(x, D, type = "mle")  
  
# Asymptotic Variance - The v Functions  
  
vbeta(shape1, shape2, type = "mle")  
vbeta(shape1, shape2, type = "me")  
vbeta(shape1, shape2, type = "same")  
  
avar_mle(D)  
avar_me(D)  
avar_same(D)  
  
avar(D, type = "mle")
```

**Description**

Exponential Distribution

**Usage**

```
Exp(rate = 1)

## S4 method for signature 'Exp'
d(x)

## S4 method for signature 'Exp'
p(x)

## S4 method for signature 'Exp'
qn(x)

## S4 method for signature 'Exp'
r(x)

## S4 method for signature 'Exp'
mean(x)

## S4 method for signature 'Exp'
median(x)

## S4 method for signature 'Exp'
mode(x)

## S4 method for signature 'Exp'
var(x)

## S4 method for signature 'Exp'
sd(x)

## S4 method for signature 'Exp'
skew(x)

## S4 method for signature 'Exp'
kurt(x)

## S4 method for signature 'Exp'
entro(x)

## S4 method for signature 'Exp'
finf(x)

## S4 method for signature 'numeric,numeric,Exp'
ll(x, prm, distr)
```

```
## S4 method for signature 'numeric,Exp'
mle(x, distr)

## S4 method for signature 'numeric,Exp'
me(x, distr)

## S4 method for signature 'Exp'
avar_mle(distr)

## S4 method for signature 'Exp'
avar_me(distr)
```

### Arguments

rate	numeric. The distribution parameters.
x	an object of class Exp. If the function also has a <code>distr</code> argument, x is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class Exp.

### Value

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The `moments` family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on `Distribution` objects returns a `Distribution` object of the appropriate class and with the appropriate parameters.

---

Fisher

*Fisher Distribution*

---

### Description

Fisher Distribution

### Usage

```
Fisher(df1 = 1, df2 = 1, ncp = 0)

## S4 method for signature 'Fisher'
d(x)

## S4 method for signature 'Fisher'
```

```

p(x)

## S4 method for signature 'Fisher'
qn(x)

## S4 method for signature 'Fisher'
r(x)

## S4 method for signature 'Fisher'
mean(x)

## S4 method for signature 'Fisher'
mode(x)

## S4 method for signature 'Fisher'
var(x)

## S4 method for signature 'Fisher'
sd(x)

## S4 method for signature 'Fisher'
skew(x)

```

### Arguments

df1, df2, ncp      numeric. The distribution parameters.

x                    an object of class Fisher. If the function also has a `distr` argument, x is a numeric vector, a sample of observations.

### Value

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The `moments` family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

Gam

*Gamma Distribution*

---

### Description

Gamma Distribution

**Usage**

```
Gam(shape = 1, scale = 1)

## S4 method for signature 'Gam'
d(x)

## S4 method for signature 'Gam'
p(x)

## S4 method for signature 'Gam'
qn(x)

## S4 method for signature 'Gam'
r(x)

## S4 method for signature 'Gam'
mean(x)

## S4 method for signature 'Gam'
var(x)

## S4 method for signature 'Gam'
sd(x)

## S4 method for signature 'Gam'
skew(x)

## S4 method for signature 'Gam'
kurt(x)

## S4 method for signature 'Gam'
entro(x)

## S4 method for signature 'Gam'
finf(x)

## S4 method for signature 'numeric,numeric,Gam'
ll(x, prm, distr)

## S4 method for signature 'numeric,Gam'
mle(x, distr, par0 = "same", method = "L-BFGS-B", lower = 1e-05, upper = Inf)

## S4 method for signature 'numeric,Gam'
me(x, distr)

## S4 method for signature 'numeric,Gam'
same(x, distr)
```

```
## S4 method for signature 'Gam'
avar_mle(distr)
```

```
## S4 method for signature 'Gam'
avar_me(distr)
```

```
## S4 method for signature 'Gam'
avar_same(distr)
```

### Arguments

shape, scale	numeric. The distribution parameters.
x	an object of class Gam. If the function also has a distr argument, x is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class Gam.
par0, method, lower, upper	arguments passed to optim.

### Value

The dpqr family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The ll function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The estim family of functions return the estimated parameters of the distribution, given a sample. The avar family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

Geom

*Geometric Distribution*

---

### Description

Geometric Distribution

### Usage

```
Geom(prob = 0.5)
```

```
## S4 method for signature 'Geom'
d(x)
```

```
## S4 method for signature 'Geom'
p(x)
```

```
## S4 method for signature 'Geom'  
qn(x)  
  
## S4 method for signature 'Geom'  
r(x)  
  
## S4 method for signature 'Geom'  
mean(x)  
  
## S4 method for signature 'Geom'  
mode(x)  
  
## S4 method for signature 'Geom'  
var(x)  
  
## S4 method for signature 'Geom'  
sd(x)  
  
## S4 method for signature 'Geom'  
skew(x)  
  
## S4 method for signature 'Geom'  
kurt(x)  
  
## S4 method for signature 'Geom'  
entro(x)  
  
## S4 method for signature 'Geom'  
finf(x)  
  
## S4 method for signature 'numeric,numeric,Geom'  
ll(x, prm, distr)  
  
## S4 method for signature 'numeric,Geom'  
mle(x, distr)  
  
## S4 method for signature 'numeric,Geom'  
me(x, distr)  
  
## S4 method for signature 'Geom'  
avar_mle(distr)  
  
## S4 method for signature 'Geom'  
avar_me(distr)
```

### Arguments

prob            numeric. The distribution parameters.

<code>x</code>	an object of class <code>Geom</code> . If the function also has a <code>distr</code> argument, <code>x</code> is a numeric vector, a sample of observations.
<code>prm</code>	numeric. A vector including the distribution parameters.
<code>distr</code>	an object of class <code>Geom</code> .

**Value**

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The `moments` family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on `Distribution` objects returns a `Distribution` object of the appropriate class and with the appropriate parameters.

---

idigamma                      *Polygamma Functions*

---

**Description**

This set of functions revolve around the polygamma functions.

**Usage**

```
idigamma(x)
Ddigamma(x, y)
Dtrigamma(x, y)
gammap(x, p, log = FALSE)
```

**Arguments**

<code>x, y</code>	numeric. The points to evaluate the function.
<code>p</code>	integer. The p-variate Gamma function.
<code>log</code>	logical. Should the logarithm of the result be returned?

**Value**

numeric. The evaluated function.

**Functions**

- `idigamma()`: inverse digamma function.
- `Ddigamma()`: digamma difference function.
- `Dtrigamma()`: trigamma difference function.
- `gammap()`: p-variate gamma function

**Examples**

```
idigamma(2)
Ddigamma(2, 3)
Dtrigamma(2, 3)
gammap(1:3, 3)
```

---

Laplace

*Laplace Distribution*

---

**Description**

Laplace Distribution

**Usage**

```
Laplace(mu = 0, sigma = 1)

## S4 method for signature 'Laplace'
d(x)

## S4 method for signature 'Laplace'
p(x)

## S4 method for signature 'Laplace'
qn(x)

## S4 method for signature 'Laplace'
r(x)

## S4 method for signature 'Laplace'
mean(x)

## S4 method for signature 'Laplace'
median(x)

## S4 method for signature 'Laplace'
mode(x)

## S4 method for signature 'Laplace'
```

```

var(x)

## S4 method for signature 'Laplace'
sd(x)

## S4 method for signature 'Laplace'
skew(x)

## S4 method for signature 'Laplace'
kurt(x)

## S4 method for signature 'Laplace'
entro(x)

## S4 method for signature 'Laplace'
finf(x)

## S4 method for signature 'numeric,numeric,Laplace'
ll(x, prm, distr)

## S4 method for signature 'numeric,Laplace'
mle(x, distr)

## S4 method for signature 'numeric,Laplace'
me(x, distr)

## S4 method for signature 'Laplace'
avar_mle(distr)

## S4 method for signature 'Laplace'
avar_me(distr)

```

### Arguments

<code>mu, sigma</code>	numeric. The distribution parameters.
<code>x</code>	an object of class <code>Laplace</code> . If the function also has a <code>distr</code> argument, <code>x</code> is a numeric vector, a sample of observations.
<code>prm</code>	numeric. A vector including the distribution parameters.
<code>distr</code>	an object of class <code>Laplace</code> .

### Value

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the

corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

large_metrics	<i>Large Sample Metrics</i>
---------------	-----------------------------

---

## Description

This function performs Monte Carlo simulations to estimate the asymptotic variance - covariance matrix, characterizing the large sample behavior of an estimator. The function evaluates the metrics as a function of a single parameter, keeping the other ones constant. See Details.

## Usage

```
large_metrics(D, prm, est = c("same", "me", "mle"), ...)
```

## Arguments

D	A subclass of Distribution. The distribution family of interest.
prm	A list containing three elements (name, pos, val). See Details.
est	character. The estimator of interest. Can be a vector.
...	extra arguments.

## Details

The distribution D is used to specify an initial distribution. The list prm contains details concerning a single parameter that is allowed to change values. The quantity of interest is evaluated as a function of this parameter.

Specifically, prm includes three elements named "name", "pos", and "val". The first two elements determine the exact parameter that changes, while the third one is a numeric vector holding the values it takes. For example, in the case of the Multivariate Gamma distribution, `D <- MGamma(shape = c(1, 2), scale = 3)` and `prm <- list(name = "shape", pos = 2, val = seq(1, 1.5, by = 0.1))` means that the evaluation will be performed for the MGamma distributions with shape parameters (1, 1), (1, 1.1), ..., (1, 1.5) and scale 3. Notice that the initial shape parameter 2 in D is not utilized in the function.

## Value

A data.frame with columns "Row", "Col", "Parameter", "Estimator", and "Value".

## See Also

[small\\_metrics](#), [plot\\_small\\_metrics](#), [plot\\_large\\_metrics](#)

**Examples**

```
D <- Beta(shape1 = 1, shape2 = 2)

prm <- list(name = "shape1",
           pos = NULL,
           val = seq(0.5, 2, by = 0.5))

x <- large_metrics(D, prm,
                  est = c("mle", "me", "same"))

plot_large_metrics(x)
```

**Description**

These functions calculate the log-likelihood of an IID sample for specific values of the distribution parameters. See Details.

**Usage**

```
ll(x, prm, distr, ...)

## S4 method for signature 'ANY,ANY,character'
ll(x, prm, distr, ...)

llbern(x, prob)

llbinom(x, size, prob)

llcat(x, prob)

llcauchy(x, location, scale)

lldirichlet(x, alpha)

llexp(x, rate)

llgamma(x, shape, scale)

llgeom(x, prob)

lllaplace(x, mu, sigma)

lllnorm(x, meanlog, sdlog)
```

```
llMultinom(x, size, prob)
```

```
llnbinom(x, size, prob)
```

```
llnorm(x, mean, sd)
```

```
llpois(x, lambda)
```

```
llunif(x, min, max)
```

```
llweib(x, shape, scale)
```

### Arguments

x	numeric. A sample under estimation.
prm	numeric. A vector of the distribution parameters.
distr	A subclass of <code>Distribution</code> . The distribution family assumed.
...	extra arguments.
location, alpha, mu, sigma, meanlog, sdlog, min, max, size, prob, shape, rate, scale, mean, sd, lambda	numeric. Distribution parameters.

### Details

The log-likelihood functions are provided in two forms: the `ll<name>` distribution-specific version that follows the base R conventions, and the S4 generic `ll`.

### Value

Numeric. The value of the log-likelihood function.

---

Lnorm	<i>Lnorm Distribution</i>
-------	---------------------------

---

### Description

Lnorm Distribution

### Usage

```
Lnorm(meanlog = 0, sdlog = 1)
```

```
## S4 method for signature 'Lnorm'  
d(x)
```

```
## S4 method for signature 'Lnorm'
```

```
p(x)

## S4 method for signature 'Lnorm'
qn(x)

## S4 method for signature 'Lnorm'
r(x)

## S4 method for signature 'Lnorm'
mean(x)

## S4 method for signature 'Lnorm'
median(x)

## S4 method for signature 'Lnorm'
mode(x)

## S4 method for signature 'Lnorm'
var(x)

## S4 method for signature 'Lnorm'
sd(x)

## S4 method for signature 'Lnorm'
skew(x)

## S4 method for signature 'Lnorm'
kurt(x)

## S4 method for signature 'Lnorm'
entro(x)

## S4 method for signature 'Lnorm'
finf(x)

## S4 method for signature 'numeric,numeric,Lnorm'
ll(x, prm, distr)

## S4 method for signature 'numeric,Lnorm'
mle(x, distr)

## S4 method for signature 'numeric,Lnorm'
me(x, distr)

## S4 method for signature 'Lnorm'
avar_mle(distr)

## S4 method for signature 'Lnorm'
```

```
avar_me(distr)
```

### Arguments

meanlog, sdlog    numeric. The distribution parameters.

x                    an object of class Lnorm. If the function also has a distr argument, x is a numeric vector, a sample of observations.

prm                  numeric. A vector including the distribution parameters.

distr                an object of class Lnorm.

### Value

The dpqr family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The ll function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The estim family of functions return the estimated parameters of the distribution, given a sample. The avar family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

 me

*Moment Estimation*


---

### Description

Calculates the ME under the assumption the sample observations are independent and identically distributed (iid) according to a specified family of distributions.

### Usage

```
me(x, distr, ...)

## S4 method for signature 'ANY,character'
me(x, distr, ...)
```

### Arguments

x                    numeric. A sample under estimation.

distr                A subclass of Distribution. The distribution family assumed.

...                  extra arguments.

### Value

numeric. The estimator produced by the sample.

## References

- Ye, Z.-S. & Chen, N. (2017), Closed-form estimators for the gamma distribution derived from likelihood equations, *The American Statistician* 71(2), 177–181.
- Van der Vaart, A. W. (2000), *Asymptotic statistics*, Vol. 3, Cambridge university press.
- Tamae, H., Irie, K. & Kubokawa, T. (2020), A score-adjusted approach to closed-form estimators for the gamma and beta distributions, *Japanese Journal of Statistics and Data Science* 3, 543–561.
- Mathal, A. & Moschopoulos, P. (1992), A form of multivariate gamma distribution, *Annals of the Institute of Statistical Mathematics* 44, 97–106.
- Oikonomidis, I. & Trevezas, S. (2023), Moment-Type Estimators for the Dirichlet and the Multivariate Gamma Distributions, arXiv, <https://arxiv.org/abs/2311.15025>

## See Also

[estim](#), [mle](#), [same](#)

## Examples

```
# -----
# Beta Distribution Example
# -----

# Simulation
set.seed(1)
shape1 <- 1
shape2 <- 2
D <- Beta(shape1, shape2)
x <- r(D)(100)

# Likelihood - The ll Functions

llbeta(x, shape1, shape2)
ll(x, c(shape1, shape2), D)
ll(x, c(shape1, shape2), "beta")

# Point Estimation - The e Functions

ebeta(x, type = "mle")
ebeta(x, type = "me")
ebeta(x, type = "same")

mle(x, D)
me(x, D)
same(x, D)

estim(x, D, type = "mle")

# Asymptotic Variance - The v Functions

vbeta(shape1, shape2, type = "mle")
vbeta(shape1, shape2, type = "me")
```

```

vbeta(shape1, shape2, type = "same")

avar_mle(D)
avar_me(D)
avar_same(D)

avar(D, type = "mle")

```

---

mle

*Maximum Likelihood Estimation*


---

### Description

Calculates the MLE under the assumption the sample observations are independent and identically distributed (iid) according to a specified family of distributions.

### Usage

```

mle(x, distr, ...)

## S4 method for signature 'ANY,character'
mle(x, distr, ...)

```

### Arguments

x	numeric. A sample under estimation.
distr	A subclass of <code>Distribution</code> . The distribution family assumed.
...	extra arguments.

### Value

numeric. The estimator produced by the sample.

### References

- Ye, Z.-S. & Chen, N. (2017), Closed-form estimators for the gamma distribution derived from likelihood equations, *The American Statistician* 71(2), 177–181.
- Van der Vaart, A. W. (2000), *Asymptotic statistics*, Vol. 3, Cambridge university press.
- Tamae, H., Irie, K. & Kubokawa, T. (2020), A score-adjusted approach to closed-form estimators for the gamma and beta distributions, *Japanese Journal of Statistics and Data Science* 3, 543–561.
- Mathal, A. & Moschopoulos, P. (1992), A form of multivariate gamma distribution, *Annals of the Institute of Statistical Mathematics* 44, 97–106.
- Oikonomidis, I. & Trevezas, S. (2023), Moment-Type Estimators for the Dirichlet and the Multivariate Gamma Distributions, arXiv, <https://arxiv.org/abs/2311.15025>

**See Also**[estim](#), [me](#), [same](#)**Examples**

```
# -----  
# Beta Distribution Example  
# -----  
  
# Simulation  
set.seed(1)  
shape1 <- 1  
shape2 <- 2  
D <- Beta(shape1, shape2)  
x <- r(D)(100)  
  
# Likelihood - The ll Functions  
  
llbeta(x, shape1, shape2)  
ll(x, c(shape1, shape2), D)  
ll(x, c(shape1, shape2), "beta")  
  
# Point Estimation - The e Functions  
  
ebeta(x, type = "mle")  
ebeta(x, type = "me")  
ebeta(x, type = "same")  
  
mle(x, D)  
me(x, D)  
same(x, D)  
  
estim(x, D, type = "mle")  
  
# Asymptotic Variance - The v Functions  
  
vbeta(shape1, shape2, type = "mle")  
vbeta(shape1, shape2, type = "me")  
vbeta(shape1, shape2, type = "same")  
  
avar_mle(D)  
avar_me(D)  
avar_same(D)  
  
avar(D, type = "mle")
```

**Description**

A set of functions that calculate the theoretical moments (expectation, variance, skewness, excess kurtosis) and other important parametric functions (median, mode, entropy, Fisher information) of a distribution.

**Usage**

```
moments(x)

mean(x, ...)

median(x, na.rm = FALSE, ...)

mode(x)

var(x, y = NULL, na.rm = FALSE, use)

sd(x, na.rm = FALSE)

skew(x, ...)

kurt(x, ...)

entro(x, ...)

finf(x, ...)
```

**Arguments**

<code>x</code>	an object of a Distribution subclass.
<code>...</code>	extra arguments.
<code>y, use, na.rm</code>	arguments in mean and var standard methods from the stats package not used here.

**Details**

The `moments()` function automatically finds the available methods for a given distribution and results all of the results in a list.

Not all functions are available for distributions; for example, the `sd()` is available only for univariate distributions.

**Value**

Numeric, either vector or matrix depending on the moment and the distribution. Function `moments()` returns a list of all available methods.

**Examples**

```

# -----
# Beta Distribution Example
# -----

library(estimators)

# Create the distribution
x <- Beta(3, 5)

# List of all available moments
mom <- moments(x)

# Expectation
mean(x)
mom$mean

# Variance and Standard Deviation
var(x)
sd(x)

# Skewness and Excess Kurtosis
skew(x)
kurt(x)

# Entropy
entro(x)

# Fisher Information Matrix
finf(x)

```

---

Multinom

*Multinomial Distribution*


---

**Description**

Multinomial Distribution

**Usage**

```

Multinom(size = 1, prob = c(0.5, 0.5))

## S4 method for signature 'Multinom'
d(x)

## S4 method for signature 'Multinom'
r(x)

## S4 method for signature 'Multinom'

```

```

mean(x)

## S4 method for signature 'Multinom'
var(x)

## S4 method for signature 'Multinom'
finf(x)

## S4 method for signature 'matrix,numeric,Multinom'
ll(x, prm, distr)

## S4 method for signature 'matrix,Multinom'
mle(x, distr)

## S4 method for signature 'matrix,Multinom'
me(x, distr)

## S4 method for signature 'Multinom'
avar_mle(distr)

## S4 method for signature 'Multinom'
avar_me(distr)

```

### Arguments

size, prob	numeric. The distribution parameters.
x	an object of class <code>Multinom</code> . If the function also has a <code>distr</code> argument, <code>x</code> is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class <code>Multinom</code> .

### Value

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The `moments` family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on `Distribution` objects returns a `Distribution` object of the appropriate class and with the appropriate parameters.

**Description**

Negative Binomial Distribution

**Usage**

```
Nbinom(size = 1, prob = 0.5)

## S4 method for signature 'Nbinom'
d(x)

## S4 method for signature 'Nbinom'
p(x)

## S4 method for signature 'Nbinom'
qn(x)

## S4 method for signature 'Nbinom'
r(x)

## S4 method for signature 'Nbinom'
mean(x)

## S4 method for signature 'Nbinom'
mode(x)

## S4 method for signature 'Nbinom'
var(x)

## S4 method for signature 'Nbinom'
sd(x)

## S4 method for signature 'Nbinom'
skew(x)

## S4 method for signature 'Nbinom'
kurt(x)

## S4 method for signature 'Nbinom'
finf(x)

## S4 method for signature 'numeric,numeric,Nbinom'
ll(x, prm, distr)

## S4 method for signature 'numeric,Nbinom'
mle(x, distr)

## S4 method for signature 'numeric,Nbinom'
me(x, distr)
```

```
## S4 method for signature 'Nbinom'
avar_mle(distr)
```

```
## S4 method for signature 'Nbinom'
avar_me(distr)
```

### Arguments

size, prob	numeric. The distribution parameters.
x	an object of class Nbinom. If the function also has a distr argument, x is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class Nbinom.

### Value

The dpqr family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The ll function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The estim family of functions return the estimated parameters of the distribution, given a sample. The avar family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

Norm

*Normal Distribution*

---

### Description

Normal Distribution

### Usage

```
Norm(mean = 0, sd = 1)
```

```
## S4 method for signature 'Norm'
d(x)
```

```
## S4 method for signature 'Norm'
p(x)
```

```
## S4 method for signature 'Norm'
qn(x)
```

```
## S4 method for signature 'Norm'
```

```
r(x)

## S4 method for signature 'Norm'
mean(x)

## S4 method for signature 'Norm'
median(x)

## S4 method for signature 'Norm'
mode(x)

## S4 method for signature 'Norm'
var(x)

## S4 method for signature 'Norm'
sd(x)

## S4 method for signature 'Norm'
skew(x)

## S4 method for signature 'Norm'
kurt(x)

## S4 method for signature 'Norm'
entro(x)

## S4 method for signature 'Norm'
finf(x)

## S4 method for signature 'numeric,numeric,Norm'
ll(x, prm, distr)

## S4 method for signature 'numeric,Norm'
mle(x, distr)

## S4 method for signature 'numeric,Norm'
me(x, distr)

## S4 method for signature 'Norm'
avar_mle(distr)

## S4 method for signature 'Norm'
avar_me(distr)
```

### Arguments

mean, sd	numeric. The distribution parameters.
x	an object of class Norm. If the function also has a <code>distr</code> argument, x is a numeric

	vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class Norm.

**Value**

The dpqr family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The ll function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The estim family of functions return the estimated parameters of the distribution, given a sample. The avar family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

---

plot\_large\_metrics      *Plot Large Sample Metrics*

---

**Description**

This function provides an easy way to illustrate the output of large\_metrics(), using the ggplot2 package. A grid of line charts is created for each element of the asymptotic variance - covariance matrix. Each estimator is plotted with a different color and linetype. The plot can be saved in pdf format.

**Usage**

```
plot_large_metrics(
  x,
  colors = NULL,
  title = NULL,
  save = FALSE,
  path = NULL,
  name = "myplot.pdf",
  width = 15,
  height = 8
)
```

**Arguments**

x	A data.frame. The result of small_metrics().
colors	character. The colors to be used in the plot.
title	character. The plot title.
save	logical. Should the plot be saved?
path	A path to the directory in which the plot will be saved.

name	character. The name of the output pdf file.
width	numeric. The plot width in inches.
height	numeric. The plot height in inches.

**Value**

The plot is returned invisibly in the form of a ggplot object.

**See Also**

[small\\_metrics](#), [large\\_metrics](#), [plot\\_small\\_metrics](#)

**Examples**

```
D <- Beta(shape1 = 1, shape2 = 2)

prm <- list(name = "shape1",
           pos = NULL,
           val = seq(0.5, 2, by = 0.5))

x <- small_metrics(D, prm,
                  est = c("mle", "me", "same"),
                  obs = c(20, 50),
                  sam = 1e2,
                  seed = 1)

plot_small_metrics(x)
```

---

plot\_small\_metrics      *Plot Small Sample Metrics*

---

**Description**

This function provides an easy way to illustrate the output of `small_metrics()`, using the `ggplot2` package. A grid of line charts is created for each metric and sample size. Each estimator is plotted with a different color and linetype. The plot can be saved in pdf format.

**Usage**

```
plot_small_metrics(
  x,
  colors = NULL,
  title = NULL,
  save = FALSE,
  path = NULL,
  name = "myplot.pdf",
  width = 15,
  height = 8
)
```

**Arguments**

x	A data.frame. The result of <code>small_metrics()</code> .
colors	character. The colors to be used in the plot.
title	character. The plot title.
save	logical. Should the plot be saved?
path	A path to the directory in which the plot will be saved.
name	character. The name of the output pdf file.
width	numeric. The plot width in inches.
height	numeric. The plot height in inches.

**Value**

The plot is returned invisibly in the form of a ggplot object.

**See Also**

[small\\_metrics](#), [large\\_metrics](#), [plot\\_large\\_metrics](#)

**Examples**

```
D <- Beta(shape1 = 1, shape2 = 2)

prm <- list(name = "shape1",
           pos = NULL,
           val = seq(0.5, 2, by = 0.5))

x <- small_metrics(D, prm,
                  est = c("mle", "me", "same"),
                  obs = c(20, 50),
                  sam = 1e2,
                  seed = 1)

plot_small_metrics(x)
```

---

Pois

*Poisson Distribution*

---

**Description**

Poisson Distribution

**Usage**

```
Pois(lambda = 1)

## S4 method for signature 'Pois'
d(x)

## S4 method for signature 'Pois'
p(x)

## S4 method for signature 'Pois'
qn(x)

## S4 method for signature 'Pois'
r(x)

## S4 method for signature 'Pois'
mean(x)

## S4 method for signature 'Pois'
var(x)

## S4 method for signature 'Pois'
sd(x)

## S4 method for signature 'Pois'
skew(x)

## S4 method for signature 'Pois'
kurt(x)

## S4 method for signature 'Pois'
finf(x)

## S4 method for signature 'numeric,numeric,Pois'
ll(x, prm, distr)

## S4 method for signature 'numeric,Pois'
mle(x, distr)

## S4 method for signature 'numeric,Pois'
me(x, distr)

## S4 method for signature 'Pois'
avar_mle(distr)

## S4 method for signature 'Pois'
avar_me(distr)
```

**Arguments**

<code>lambda</code>	numeric. The distribution parameters.
<code>x</code>	an object of class <code>Pois</code> . If the function also has a <code>distr</code> argument, <code>x</code> is a numeric vector, a sample of observations.
<code>prm</code>	numeric. A vector including the distribution parameters.
<code>distr</code>	an object of class <code>Pois</code> .

**Value**

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on `Distribution` objects returns a `Distribution` object of the appropriate class and with the appropriate parameters.

---

same

---

*Score - Adjusted Moment Estimation*


---

**Description**

Calculates the SAME under the assumption the sample observations are independent and identically distributed (iid) according to a specified family of distributions.

**Usage**

```
same(x, distr, ...)

## S4 method for signature 'ANY,character'
same(x, distr, ...)
```

**Arguments**

<code>x</code>	numeric. A sample under estimation.
<code>distr</code>	A subclass of <code>Distribution</code> . The distribution family assumed.
<code>...</code>	extra arguments.

**Value**

numeric. The estimator produced by the sample.

## References

- Ye, Z.-S. & Chen, N. (2017), Closed-form estimators for the gamma distribution derived from likelihood equations, *The American Statistician* 71(2), 177–181.
- Van der Vaart, A. W. (2000), *Asymptotic statistics*, Vol. 3, Cambridge university press.
- Tamae, H., Irie, K. & Kubokawa, T. (2020), A score-adjusted approach to closed-form estimators for the gamma and beta distributions, *Japanese Journal of Statistics and Data Science* 3, 543–561.
- Mathal, A. & Moschopoulos, P. (1992), A form of multivariate gamma distribution, *Annals of the Institute of Statistical Mathematics* 44, 97–106.
- Oikonomidis, I. & Trevezas, S. (2023), Moment-Type Estimators for the Dirichlet and the Multivariate Gamma Distributions, arXiv, <https://arxiv.org/abs/2311.15025>

## See Also

[estim](#), [mle](#), [me](#)

## Examples

```
# -----
# Beta Distribution Example
# -----

# Simulation
set.seed(1)
shape1 <- 1
shape2 <- 2
D <- Beta(shape1, shape2)
x <- r(D)(100)

# Likelihood - The ll Functions

llbeta(x, shape1, shape2)
ll(x, c(shape1, shape2), D)
ll(x, c(shape1, shape2), "beta")

# Point Estimation - The e Functions

ebeta(x, type = "mle")
ebeta(x, type = "me")
ebeta(x, type = "same")

mle(x, D)
me(x, D)
same(x, D)

estim(x, D, type = "mle")

# Asymptotic Variance - The v Functions

vbeta(shape1, shape2, type = "mle")
vbeta(shape1, shape2, type = "me")
```

```

vbeta(shape1, shape2, type = "same")

avar_mle(D)
avar_me(D)
avar_same(D)

avar(D, type = "mle")

```

---

small\_metrics                      *Small Sample Metrics*

---

### Description

This function performs Monte Carlo simulations to estimate the main metrics (bias, variance, and RMSE) characterizing the small sample behavior of an estimator. The function evaluates the metrics as a function of a single parameter, keeping the other ones constant. See Details.

### Usage

```

small_metrics(
  D,
  prm,
  est = c("same", "me", "mle"),
  obs = c(20, 50, 100),
  sam = 10000,
  seed = 1,
  ...
)

```

### Arguments

D	A subclass of <code>Distribution</code> . The distribution family of interest.
prm	A list containing three elements (name, pos, val). See Details.
est	character. The estimator of interest. Can be a vector.
obs	numeric. The size of each sample. Can be a vector.
sam	numeric. The number of Monte Carlo samples used to estimate the metrics.
seed	numeric. Passed to <code>set.seed()</code> for reproducibility.
...	extra arguments.

### Details

The distribution `D` is used to specify an initial distribution. The list `prm` contains details concerning a single parameter that is allowed to change values. The quantity of interest is evaluated as a function of this parameter.

Specifically, `prm` includes three elements named "name", "pos", and "val". The first two elements determine the exact parameter that changes, while the third one is a numeric vector holding the values it takes. For example, in the case of the Multivariate Gamma distribution, `D <- MGamma(shape = c(1, 2), scale = 3)` and `prm <- list(name = "shape", pos = 2, val = seq(1, 1.5, by = 0.1))` means that the evaluation will be performed for the MGamma distributions with shape parameters (1, 1), (1, 1.1), ..., (1, 1.5) and scale 3. Notice that the initial shape parameter 2 in `D` is not utilized in the function.

### Value

For the small sample, a data.frame with columns named "Parameter", "Observations", "Estimator", "Metric", and "Value". For the large sample, a data.frame with columns "Row", "Col", "Parameter", "Estimator", and "Value".

### See Also

[plot\\_small\\_metrics](#) [large\\_metrics](#), [plot\\_large\\_metrics](#)

### Examples

```
D <- Beta(shape1 = 1, shape2 = 2)

prm <- list(name = "shape1",
           pos = NULL,
           val = seq(0.5, 2, by = 0.5))

x <- small_metrics(D, prm,
                 est = c("mle", "me", "same"),
                 obs = c(20, 50),
                 sam = 1e2,
                 seed = 1)

plot_small_metrics(x)
```

---

Stud

*Student Distribution*

---

### Description

Student Distribution

### Usage

```
Stud(df = 1, ncp = 0)

## S4 method for signature 'Stud'
d(x)
```

```
## S4 method for signature 'Stud'  
p(x)  
  
## S4 method for signature 'Stud'  
qn(x)  
  
## S4 method for signature 'Stud'  
r(x)  
  
## S4 method for signature 'Stud'  
mean(x)  
  
## S4 method for signature 'Stud'  
median(x)  
  
## S4 method for signature 'Stud'  
mode(x)  
  
## S4 method for signature 'Stud'  
var(x)  
  
## S4 method for signature 'Stud'  
sd(x)  
  
## S4 method for signature 'Stud'  
skew(x)  
  
## S4 method for signature 'Stud'  
kurt(x)
```

### Arguments

<code>df, ncp</code>	numeric. The distribution parameters.
<code>x</code>	an object of class <code>Stud</code> . If the function also has a <code>distr</code> argument, <code>x</code> is a numeric vector, a sample of observations.

### Value

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The `moments` family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on `Distribution` objects returns a `Distribution` object of the appropriate class and with the appropriate parameters.

---

Unif

*Uniform Distribution*

---

### Description

Uniform Distribution

### Usage

```
Unif(min = 0, max = 1)
```

```
## S4 method for signature 'Unif'  
d(x)
```

```
## S4 method for signature 'Unif'  
p(x)
```

```
## S4 method for signature 'Unif'  
qn(x)
```

```
## S4 method for signature 'Unif'  
r(x)
```

```
## S4 method for signature 'Unif'  
mean(x)
```

```
## S4 method for signature 'Unif'  
var(x)
```

```
## S4 method for signature 'Unif'  
sd(x)
```

```
## S4 method for signature 'Unif'  
skew(x)
```

```
## S4 method for signature 'Unif'  
kurt(x)
```

```
## S4 method for signature 'Unif'  
entro(x)
```

```
## S4 method for signature 'numeric,numeric,Unif'  
ll(x, prm, distr)
```

```
## S4 method for signature 'numeric,Unif'  
mle(x, distr)
```

```
## S4 method for signature 'numeric,Unif'
me(x, distr)
```

### Arguments

min, max	numeric. The distribution parameters.
x	an object of class <code>Unif</code> . If the function also has a <code>distr</code> argument, <code>x</code> is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class <code>Unif</code> .

### Value

The `dpqr` family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The `ll` function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The `estim` family of functions return the estimated parameters of the distribution, given a sample. The `avar` family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on `Distribution` objects returns a `Distribution` object of the appropriate class and with the appropriate parameters.

---

Weib

*Weibull Distribution*

---

### Description

Weibull Distribution

### Usage

```
Weib(shape = 1, scale = 1)

## S4 method for signature 'Weib'
d(x)

## S4 method for signature 'Weib'
p(x)

## S4 method for signature 'Weib'
qn(x)

## S4 method for signature 'Weib'
r(x)

## S4 method for signature 'Weib'
mean(x)
```

```

## S4 method for signature 'Weib'
median(x)

## S4 method for signature 'Weib'
mode(x)

## S4 method for signature 'Weib'
var(x)

## S4 method for signature 'Weib'
sd(x)

## S4 method for signature 'Weib'
skew(x)

## S4 method for signature 'Weib'
kurt(x)

## S4 method for signature 'Weib'
entro(x)

## S4 method for signature 'numeric,numeric,Weib'
ll(x, prm, distr)

## S4 method for signature 'numeric,Weib'
mle(x, distr, par0 = "same", method = "L-BFGS-B", lower = 1e-05, upper = Inf)

## S4 method for signature 'numeric,Weib'
me(x, distr)

## S4 method for signature 'Weib'
avar_mle(distr)

## S4 method for signature 'Weib'
avar_me(distr)

```

### Arguments

shape, scale	numeric. The distribution parameters.
x	an object of class Weib. If the function also has a <code>distr</code> argument, x is a numeric vector, a sample of observations.
prm	numeric. A vector including the distribution parameters.
distr	an object of class Weib.
par0, method, lower, upper	arguments passed to <code>optim</code> .

**Value**

The dpqr family of functions return the evaluated density, cumulative probability, quantile, and random sample, respectively. The moments family of functions return the appropriate theoretical moment, as calculated by the distribution true parameters. The ll function returns the evaluated log-likelihood, given a sample and the theoretical parameters. The estim family of functions return the estimated parameters of the distribution, given a sample. The avar family of functions return the asymptotic variance or variance - covariance matrix (if there are two or more parameters) of the corresponding estimation method. Calculus performed on Distribution objects returns a Distribution object of the appropriate class and with the appropriate parameters.

# Index

- [\\*, Norm, numeric-method \(calculus\), 15](#)
  - [\\*, numeric, Norm-method \(calculus\), 15](#)
  - [+, Norm, Norm-method \(calculus\), 15](#)
  - [+, Norm, numeric-method \(calculus\), 15](#)
  - [+, numeric, Norm-method \(calculus\), 15](#)
  - [-, Norm, Norm-method \(calculus\), 15](#)
  - [-, Norm, numeric-method \(calculus\), 15](#)
  - [-, numeric, Norm-method \(calculus\), 15](#)
  - [/, Norm, numeric-method \(calculus\), 15](#)
  
- [avar, 3, 5, 7, 8](#)
- [avar\\_me, 4, 5, 7, 8](#)
- [avar\\_me, Bern-method \(Bern\), 9](#)
- [avar\\_me, Beta-method \(Beta\), 11](#)
- [avar\\_me, Binom-method \(Binom\), 13](#)
- [avar\\_me, Cat-method \(Cat\), 16](#)
- [avar\\_me, Dir-method \(Dir\), 20](#)
- [avar\\_me, Exp-method \(Exp\), 25](#)
- [avar\\_me, Gam-method \(Gam\), 28](#)
- [avar\\_me, Geom-method \(Geom\), 30](#)
- [avar\\_me, Laplace-method \(Laplace\), 33](#)
- [avar\\_me, Lnorm-method \(Lnorm\), 37](#)
- [avar\\_me, Multinom-method \(Multinom\), 44](#)
- [avar\\_me, Nbinom-method \(Nbinom\), 45](#)
- [avar\\_me, Norm-method \(Norm\), 47](#)
- [avar\\_me, Pois-method \(Pois\), 51](#)
- [avar\\_me, Weib-method \(Weib\), 59](#)
- [avar\\_mle, 4, 5, 6, 8](#)
- [avar\\_mle, Bern-method \(Bern\), 9](#)
- [avar\\_mle, Beta-method \(Beta\), 11](#)
- [avar\\_mle, Binom-method \(Binom\), 13](#)
- [avar\\_mle, Cat-method \(Cat\), 16](#)
- [avar\\_mle, Dir-method \(Dir\), 20](#)
- [avar\\_mle, Exp-method \(Exp\), 25](#)
- [avar\\_mle, Gam-method \(Gam\), 28](#)
- [avar\\_mle, Geom-method \(Geom\), 30](#)
- [avar\\_mle, Laplace-method \(Laplace\), 33](#)
- [avar\\_mle, Lnorm-method \(Lnorm\), 37](#)
- [avar\\_mle, Multinom-method \(Multinom\), 44](#)
- [avar\\_mle, Nbinom-method \(Nbinom\), 45](#)
- [avar\\_mle, Norm-method \(Norm\), 47](#)
- [avar\\_mle, Pois-method \(Pois\), 51](#)
- [avar\\_mle, Weib-method \(Weib\), 59](#)

- Distributions, 22
- dpqr, 11, 22
- Dtrigamma (idigamma), 32
- ebern (estim), 23
- ebeta (estim), 23
- ebinom (estim), 23
- ecat (estim), 23
- edirichlet (estim), 23
- eexp (estim), 23
- egamma (estim), 23
- egeom (estim), 23
- elaplace (estim), 23
- elnorm (estim), 23
- emultinom (estim), 23
- enbinom (estim), 23
- enorm (estim), 23
- entro (moments), 42
- entro, Bern-method (Bern), 9
- entro, Beta-method (Beta), 11
- entro, Cauchy-method (Cauchy), 18
- entro, Dir-method (Dir), 20
- entro, Exp-method (Exp), 25
- entro, Gam-method (Gam), 28
- entro, Geom-method (Geom), 30
- entro, Laplace-method (Laplace), 33
- entro, Lnorm-method (Lnorm), 37
- entro, Norm-method (Norm), 47
- entro, Unif-method (Unif), 58
- entro, Weib-method (Weib), 59
- epois (estim), 23
- estim, 23, 40, 42, 54
- eunif (estim), 23
- eweib (estim), 23
- Exp, 25
- exp, Norm-method (calculus), 15
- finf (moments), 42
- finf, Bern-method (Bern), 9
- finf, Beta-method (Beta), 11
- finf, Binom-method (Binom), 13
- finf, Cat-method (Cat), 16
- finf, Cauchy-method (Cauchy), 18
- finf, Dir-method (Dir), 20
- finf, Exp-method (Exp), 25
- finf, Gam-method (Gam), 28
- finf, Geom-method (Geom), 30
- finf, Laplace-method (Laplace), 33
- finf, Lnorm-method (Lnorm), 37
- finf, Multinom-method (Multinom), 44
- finf, Nbinom-method (Nbinom), 45
- finf, Norm-method (Norm), 47
- finf, Pois-method (Pois), 51
- Fisher, 27
- Gam, 28
- gammap (idigamma), 32
- Geom, 30
- idigamma, 32
- kurt (moments), 42
- kurt, Bern-method (Bern), 9
- kurt, Beta-method (Beta), 11
- kurt, Binom-method (Binom), 13
- kurt, Chisq-method (Chisq), 19
- kurt, Exp-method (Exp), 25
- kurt, Gam-method (Gam), 28
- kurt, Geom-method (Geom), 30
- kurt, Laplace-method (Laplace), 33
- kurt, Lnorm-method (Lnorm), 37
- kurt, Nbinom-method (Nbinom), 45
- kurt, Norm-method (Norm), 47
- kurt, Pois-method (Pois), 51
- kurt, Stud-method (Stud), 56
- kurt, Unif-method (Unif), 58
- kurt, Weib-method (Weib), 59
- Laplace, 33
- large\_metrics, 35, 50, 51, 56
- ll, 36
- ll, ANY, ANY, character-method (ll), 36
- ll, matrix, numeric, Dir-method (Dir), 20
- ll, matrix, numeric, Multinom-method (Multinom), 44
- ll, numeric, numeric, Bern-method (Bern), 9
- ll, numeric, numeric, Beta-method (Beta), 11
- ll, numeric, numeric, Binom-method (Binom), 13
- ll, numeric, numeric, Cat-method (Cat), 16
- ll, numeric, numeric, Cauchy-method (Cauchy), 18
- ll, numeric, numeric, Exp-method (Exp), 25
- ll, numeric, numeric, Gam-method (Gam), 28
- ll, numeric, numeric, Geom-method (Geom), 30
- ll, numeric, numeric, Laplace-method (Laplace), 33

- ll,numeric,numeric,Lnorm-method (Lnorm), 37
- ll,numeric,numeric,Nbinom-method (Nbinom), 45
- ll,numeric,numeric,Norm-method (Norm), 47
- ll,numeric,numeric,Pois-method (Pois), 51
- ll,numeric,numeric,Unif-method (Unif), 58
- ll,numeric,numeric>Weib-method (Weib), 59
- llbern (ll), 36
- llbeta (Beta), 11
- llbinom (ll), 36
- llcat (ll), 36
- llcauchy (ll), 36
- lldirichlet (ll), 36
- llexp (ll), 36
- llgamma (ll), 36
- llgeom (ll), 36
- lllaplace (ll), 36
- lllnorm (ll), 36
- llMultinom (ll), 36
- llnbinom (ll), 36
- llnorm (ll), 36
- llpois (ll), 36
- llunif (ll), 36
- llweib (ll), 36
- Lnorm, 37
- me, 25, 39, 42, 54
- me,ANY,character-method (me), 39
- me,matrix,Dir-method (Dir), 20
- me,matrix,Multinom-method (Multinom), 44
- me,numeric,Bern-method (Bern), 9
- me,numeric,Beta-method (Beta), 11
- me,numeric,Binom-method (Binom), 13
- me,numeric,Cat-method (Cat), 16
- me,numeric,Exp-method (Exp), 25
- me,numeric,Gam-method (Gam), 28
- me,numeric,Geom-method (Geom), 30
- me,numeric,Laplace-method (Laplace), 33
- me,numeric,Lnorm-method (Lnorm), 37
- me,numeric,Nbinom-method (Nbinom), 45
- me,numeric,Norm-method (Norm), 47
- me,numeric,Pois-method (Pois), 51
- me,numeric,Unif-method (Unif), 58
- me,numeric>Weib-method (Weib), 59
- mean (moments), 42
- mean,Bern-method (Bern), 9
- mean,Beta-method (Beta), 11
- mean,Binom-method (Binom), 13
- mean,Cat-method (Cat), 16
- mean,Chisq-method (Chisq), 19
- mean,Dir-method (Dir), 20
- mean,Exp-method (Exp), 25
- mean,Fisher-method (Fisher), 27
- mean,Gam-method (Gam), 28
- mean,Geom-method (Geom), 30
- mean,Laplace-method (Laplace), 33
- mean,Lnorm-method (Lnorm), 37
- mean,Multinom-method (Multinom), 44
- mean,Nbinom-method (Nbinom), 45
- mean,Norm-method (Norm), 47
- mean,Pois-method (Pois), 51
- mean,Stud-method (Stud), 56
- mean,Unif-method (Unif), 58
- mean>Weib-method (Weib), 59
- median (moments), 42
- median,Bern-method (Bern), 9
- median,Beta-method (Beta), 11
- median,Cauchy-method (Cauchy), 18
- median,Exp-method (Exp), 25
- median,Laplace-method (Laplace), 33
- median,Lnorm-method (Lnorm), 37
- median,Norm-method (Norm), 47
- median,Stud-method (Stud), 56
- median>Weib-method (Weib), 59
- mle, 25, 40, 41, 54
- mle,ANY,character-method (mle), 41
- mle,matrix,Dir-method (Dir), 20
- mle,matrix,Multinom-method (Multinom), 44
- mle,numeric,Bern-method (Bern), 9
- mle,numeric,Beta-method (Beta), 11
- mle,numeric,Binom-method (Binom), 13
- mle,numeric,Cat-method (Cat), 16
- mle,numeric,Exp-method (Exp), 25
- mle,numeric,Gam-method (Gam), 28
- mle,numeric,Geom-method (Geom), 30
- mle,numeric,Laplace-method (Laplace), 33
- mle,numeric,Lnorm-method (Lnorm), 37
- mle,numeric,Nbinom-method (Nbinom), 45
- mle,numeric,Norm-method (Norm), 47
- mle,numeric,Pois-method (Pois), 51
- mle,numeric,Unif-method (Unif), 58

- mle, numeric, Weib-method (Weib), 59
- mode (moments), 42
- mode, Bern-method (Bern), 9
- mode, Beta-method (Beta), 11
- mode, Cauchy-method (Cauchy), 18
- mode, Dir-method (Dir), 20
- mode, Exp-method (Exp), 25
- mode, Fisher-method (Fisher), 27
- mode, Geom-method (Geom), 30
- mode, Laplace-method (Laplace), 33
- mode, Lnorm-method (Lnorm), 37
- mode, Nbinom-method (Nbinom), 45
- mode, Norm-method (Norm), 47
- mode, Stud-method (Stud), 56
- mode, Weib-method (Weib), 59
- moments, 11, 42
- Multinom, 44
  
- Nbinom, 45
- Norm, 47
  
- p (dpqr), 22
- p, Bern-method (Bern), 9
- p, Beta-method (Beta), 11
- p, Binom-method (Binom), 13
- p, Cauchy-method (Cauchy), 18
- p, Chisq-method (Chisq), 19
- p, Exp-method (Exp), 25
- p, Fisher-method (Fisher), 27
- p, Gam-method (Gam), 28
- p, Geom-method (Geom), 30
- p, Laplace-method (Laplace), 33
- p, Lnorm-method (Lnorm), 37
- p, Nbinom-method (Nbinom), 45
- p, Norm-method (Norm), 47
- p, Pois-method (Pois), 51
- p, Stud-method (Stud), 56
- p, Unif-method (Unif), 58
- p, Weib-method (Weib), 59
- plot\_large\_metrics, 35, 49, 51, 56
- plot\_small\_metrics, 35, 50, 50, 56
- Pois, 51
  
- q (dpqr), 22
- qn (dpqr), 22
- qn, Bern-method (Bern), 9
- qn, Beta-method (Beta), 11
- qn, Binom-method (Binom), 13
- qn, Cauchy-method (Cauchy), 18
- qn, Chisq-method (Chisq), 19
- qn, Exp-method (Exp), 25
- qn, Fisher-method (Fisher), 27
- qn, Gam-method (Gam), 28
- qn, Geom-method (Geom), 30
- qn, Laplace-method (Laplace), 33
- qn, Lnorm-method (Lnorm), 37
- qn, Nbinom-method (Nbinom), 45
- qn, Norm-method (Norm), 47
- qn, Pois-method (Pois), 51
- qn, Stud-method (Stud), 56
- qn, Unif-method (Unif), 58
- qn, Weib-method (Weib), 59
  
- r (dpqr), 22
- r, Bern-method (Bern), 9
- r, Beta-method (Beta), 11
- r, Binom-method (Binom), 13
- r, Cat-method (Cat), 16
- r, Cauchy-method (Cauchy), 18
- r, Chisq-method (Chisq), 19
- r, Dir-method (Dir), 20
- r, Exp-method (Exp), 25
- r, Fisher-method (Fisher), 27
- r, Gam-method (Gam), 28
- r, Geom-method (Geom), 30
- r, Laplace-method (Laplace), 33
- r, Lnorm-method (Lnorm), 37
- r, Multinom-method (Multinom), 44
- r, Nbinom-method (Nbinom), 45
- r, Norm-method (Norm), 47
- r, Pois-method (Pois), 51
- r, Stud-method (Stud), 56
- r, Unif-method (Unif), 58
- r, Weib-method (Weib), 59
  
- same, 25, 40, 42, 53
- same, ANY, character-method (same), 53
- same, matrix, Dir-method (Dir), 20
- same, numeric, Beta-method (Beta), 11
- same, numeric, Gam-method (Gam), 28
- sd (moments), 42
- sd, Bern-method (Bern), 9
- sd, Beta-method (Beta), 11
- sd, Binom-method (Binom), 13
- sd, Chisq-method (Chisq), 19
- sd, Exp-method (Exp), 25
- sd, Fisher-method (Fisher), 27
- sd, Gam-method (Gam), 28

- sd,Geom-method (Geom), 30
- sd,Laplace-method (Laplace), 33
- sd,Lnorm-method (Lnorm), 37
- sd,Nbinom-method (Nbinom), 45
- sd,Norm-method (Norm), 47
- sd,Pois-method (Pois), 51
- sd,Stud-method (Stud), 56
- sd,Unif-method (Unif), 58
- sd,Weib-method (Weib), 59
- skew (moments), 42
- skew,Bern-method (Bern), 9
- skew,Beta-method (Beta), 11
- skew,Binom-method (Binom), 13
- skew,Chisq-method (Chisq), 19
- skew,Exp-method (Exp), 25
- skew,Fisher-method (Fisher), 27
- skew,Gam-method (Gam), 28
- skew,Geom-method (Geom), 30
- skew,Laplace-method (Laplace), 33
- skew,Lnorm-method (Lnorm), 37
- skew,Nbinom-method (Nbinom), 45
- skew,Norm-method (Norm), 47
- skew,Pois-method (Pois), 51
- skew,Stud-method (Stud), 56
- skew,Unif-method (Unif), 58
- skew,Weib-method (Weib), 59
- small\_metrics, 35, 50, 51, 55
- Stud, 56
- sum,Norm,logical-method (calculus), 15
- Unif, 58
- var (moments), 42
- var,Bern-method (Bern), 9
- var,Beta-method (Beta), 11
- var,Binom-method (Binom), 13
- var,Cat-method (Cat), 16
- var,Chisq-method (Chisq), 19
- var,Dir-method (Dir), 20
- var,Exp-method (Exp), 25
- var,Fisher-method (Fisher), 27
- var,Gam-method (Gam), 28
- var,Geom-method (Geom), 30
- var,Laplace-method (Laplace), 33
- var,Lnorm-method (Lnorm), 37
- var,Multinom-method (Multinom), 44
- var,Nbinom-method (Nbinom), 45
- var,Norm-method (Norm), 47
- var,Pois-method (Pois), 51
- var,Stud-method (Stud), 56
- var,Unif-method (Unif), 58
- var,Weib-method (Weib), 59
- vbern (avar), 3
- vbeta (Beta), 11
- vbinom (avar), 3
- vcat (avar), 3
- vdirichlet (avar), 3
- vexp (avar), 3
- vgamma (avar), 3
- vgeom (avar), 3
- vlaplace (avar), 3
- vmultinom (avar), 3
- vnbinom (avar), 3
- vnorm (avar), 3
- vpois (avar), 3
- vweib (avar), 3
- Weib, 59