# Package 'bgumbel'

October 12, 2022

**Title** Bimodal Gumbel Distribution

**Version** 0.0.3

**Description** Bimodal Gumbel distribution. General functions for performing extreme value analysis.

**Imports** MCMCpack, MASS, quantreg, SparseM, coda, stats

**License** MIT + file LICENSE

**SystemRequirements** gcc (>= 4.0), gfortran, clang++

**NeedsCompilation** yes

**URL** https://CRAN.R-project.org/package=bgumbel

**Language** en-US

**Encoding** UTF-8

**LazyData** false

**Date** 2021-03-31

**RoxygenNote** 7.1.1

**Author** Pedro C. Brom [aut, cre, cph] (<https://orcid.org/0000-0002-1288-7695>, http://lattes.cnpq.br/0154064396756002), Cira E. G. Otiniano [aut, cph] (<https://orcid.org/0000-0002-5619-0478>, http://lattes.cnpq.br/0307717595727716), Roberto Vila [aut, cph] (<https://orcid.org/0000-0003-1073-0114>, http://lattes.cnpq.br/4978745622057574), Marcelo B. Pereira [aut, cph] (<https://orcid.org/0000-0002-1182-5193>, https://lattes.cnpq.br/9358366674842900)

**Maintainer** Pedro C. Brom <pcbrom@gmail.com>

**Repository** CRAN

**Date/Publication** 2021-03-31 22:10:07 UTC

## R topics documented:

dbgumbel . . . . . 2
m1bgumbel . . . . . 3

m2bgumbel . . . . . 4
mlebgumbel . . . . . 5
pbgumbel . . . . . 6
qbgumbel . . . . . 7
rbgumbel . . . . . 8

**Index** **9**

---

`dbgumbel` *Bimodal Gumbel: Density Function*

---

## Description

Bimodal Gumbel: Density Function

## Usage

```
dbgumbel(x, mu, sigma, delta)
```

## Arguments

| | |
|---|---|
| `x` | Domain. |
| `mu` | First location parameter. |
| `sigma` | Scale parameter. |
| `delta` | Second location parameter. |

## Value

Vector.

## Examples

```
dbgumbel(x = 0, mu = -2, sigma = 1, delta = -1)
curve(dbgumbel(x, mu = -2, sigma = 1, delta = -1), xlim = c(-5, 10), ylim= c(0, .4))
integrate(dbgumbel, mu = -2, sigma = 1, delta = -1, lower = -5, upper = 0)
```

m1bgumbel *Bimodal Gumbel: Theoretical E(X)*

## Description

Bimodal Gumbel: Theoretical E(X)

## Usage

```
m1bgumbel(mu, sigma, delta)
```

## Arguments

| | |
|---|---|
| mu | First location parameter. |
| sigma | Scale parameter. |
| delta | Second location parameter. |

## Value

Vector.

## Examples

```
(EX <- m1bgumbel(mu = -2, sigma = 1, delta = -1))


# Comparison: Theoretical E(X) and empirical mean

x <- rbgumbel(100000, mu = -2, sigma = 1, delta = -1)
mean(x)
abs(EX - mean(x))/abs(EX) # relative error

# grid 1

mu <- seq(-5, 5, length.out = 100)
delta <- seq(-5, 5, length.out = 100)
z <- outer(
  X <- mu,
  Y <- delta,
  FUN = function(x, y) m1bgumbel(mu = x, sigma = 1, delta = y)
)

persp(x = mu, y = delta, z = z, theta = -60, ticktype = 'detailed')

# grid 2

mu <- seq(-5, 5, length.out = 100)
delta <- seq(-5, 5, length.out = 100)
sigmas <- seq(.1, 10, length.out = 20)
```

```
for (sigma in sigmas) {
 z <- outer(
   X <- mu,
   Y <- delta,
    FUN = function(x, y) m1bgumbel(mu = x, sigma = sigma, delta = y)
 )
 persp(x = mu, y = delta, z = z, theta = -60, zlab = 'E(X)')
 Sys.sleep(.5)
}
```

---

m2bgumbel *Bimodal Gumbel: Theoretical E(X^2)*

---

## Description

Bimodal Gumbel: Theoretical E(X^2)

## Usage

```
m2bgumbel(mu, sigma, delta)
```

## Arguments

| | |
|---|---|
| mu | First location parameter. |
| sigma | Scale parameter. |
| delta | Second location parameter. |

## Value

Vector.

## Examples

```
(EX2 <- m2bgumbel(mu = -2, sigma = 1, delta = -1))


# Comparison: Theoretical E(X^2) and empirical second moment

x <- rbgumbel(100000, mu = -2, sigma = 1, delta = -1)
mean(x^2)
abs(EX2 - mean(x))/abs(EX2) # relative error

# Variance
EX <- m1bgumbel(mu = -2, sigma = 1, delta = -1)
EX2 - EX^2
var(x)
abs(EX2 - EX^2 - var(x))/abs(EX2 - EX^2) # relative error
```

```
# grid 1

mu <- seq(-5, 5, length.out = 100)
delta <- seq(-5, 5, length.out = 100)
z <- outer(
  X <- mu,
  Y <- delta,
  FUN = function(x, y) m2bgumbel(mu = x, sigma = 1, delta = y)
)
persp(x = mu, y = delta, z = z, theta = -30, ticktype = 'detailed')

# grid 2

mu <- seq(-5, 5, length.out = 100)
delta <- seq(-5, 5, length.out = 100)
sigmas <- seq(.1, 10, length.out = 20)
for (sigma in sigmas) {
  z <- outer(
    X <- mu,
    Y <- delta,
    FUN = function(x, y) m2bgumbel(mu = x, sigma = sigma, delta = y)
  )
  persp(x = mu, y = delta, z = z, theta = -45, zlab = 'E(X^2)')
  Sys.sleep(.5)
}
```

---

`mlebgumbel` *Bimodal Gumbel: Maximum Likelihood Estimation*

---

## Description

Bimodal Gumbel: Maximum Likelihood Estimation

## Usage

```
mlebgumbel(data, theta, auto = TRUE)
```

## Arguments

| | |
|---|---|
| `data` | A numeric vector. |
| `theta` | Vector. Starting parameter values for the minimization. Default: theta = c(1, 1, 1) |
| `auto` | Logical. Automatic search for theta initial condition. Default: TRUE |

## Value

List.

## Examples

```
# Let's generate some values

set.seed(123)
x <- rbgumbel(1000, mu = -2, sigma = 1, delta = -1)

# Look for these references in the figure:

hist(x, probability = TRUE)
lines(density(x), col = 'blue')
abline(v = c(-2.5, -.5), col = 'red')
text(x = c(c(-2.5, -.5)), y = c(.05, .05), c('mu\nnear here', 'delta\nnear here'))

# Time to fit!

# If argument auto = FALSE
fit <- mlebgumbel(
   data = x,
   # try some values near the region. Format: theta = c(mu, sigma, delta)
   theta = c(-3, 2, -2),
   auto = FALSE
)
print(fit)

# If argument auto = TRUE
fit <- mlebgumbel(
   data = x,
   auto = TRUE
)
print(fit)


# Kolmogorov-Smirnov Tests

mu.sigma.delta <- fit$estimate$estimate
ks.test(
  x,
  y = 'pbgumbel',
  mu = mu.sigma.delta[[1]],
  sigma = mu.sigma.delta[[2]],
  delta = mu.sigma.delta[[3]]
)
```

---

pbgumbel *Bimodal Gumbel: Distribution Function*

---

## Description

Bimodal Gumbel: Distribution Function

### Usage

```
pbgumbel(q, mu, sigma, delta, lower.tail = TRUE)
```

### Arguments

| | |
|---|---|
| q | Quantile. |
| mu | First location parameter. |
| sigma | Scale parameter. |
| delta | Second location parameter. |
| lower.tail | Logical; if TRUE (default), probabilities are P(X <= x) otherwise, P(X > x). |

### Value

Vector.

### Examples

```
pbgumbel(0, mu = -2, sigma = 1, delta = -1)
integrate(dbgumbel, mu = -2, sigma = 1, delta = -1, lower = -Inf, upper = 0)
pbgumbel(0, mu = -2, sigma = 1, delta = -1, lower.tail = FALSE)
curve(pbgumbel(x, mu = -2, sigma = 1, delta = -1), xlim = c(-5, 10))
```

---

## qbgumbel *Bimodal Gumbel: Quantile Function*

---

### Description

Bimodal Gumbel: Quantile Function

### Usage

```
qbgumbel(p, mu, sigma, delta, initial = -10, final = 10)
```

### Arguments

| | |
|---|---|
| p | Probability. |
| mu | First location parameter. |
| sigma | Scale parameter. |
| delta | Second location parameter. |
| initial | Starting point of range in desired quantile. |
| final | Starting point of range in desired quantile. |

### Value

Vector.

## Examples

```
# It is recommended to set up a pbgumbel
# graph to see the starting and ending
# range of the desired quantile.
curve(pbgumbel(x, mu = -2, sigma = 1, delta = -1), xlim = c(-5, 5))
(value <- qbgumbel(.25, mu = -2, sigma = 1, delta = -1, initial = -4, final = -2))
pbgumbel(value, mu = -2, sigma = 1, delta = -1)
```

---

rbgumbel *Bimodal Gumbel: Pseudo-Random Numbers Generator*

---

## Description

Bimodal Gumbel: Pseudo-Random Numbers Generator

## Usage

```
rbgumbel(n, mu, sigma, delta)
```

## Arguments

| | |
|---|---|
| n | Number of observations. If length(n) > 1, the length is taken to be the number required. |
| mu | First location parameter. |
| sigma | Scale parameter. |
| delta | Second location parameter. |

## Value

A matrix nx1.

## Examples

```
x <- rbgumbel(40000, mu = -2, sigma = 1, delta = -1)
hist(x, probability = TRUE)
curve(dbgumbel(x, mu = -2, sigma = 1, delta = -1), add = TRUE, col = 'blue')
lines(density(x), col = 'red')
```

# Index

dbgumbel, 2

m1bgumbel, 3
m2bgumbel, 4
mlebgumbel, 5

pbgumbel, 6

qbgumbel, 7

rbgumbel, 8