

Package ‘weibull4’

September 17, 2020

Type Package

Title Fits Data into 4-Parameters Weibull Distribution

Version 0.1.0

Author Vitor Hugo Moreau

Maintainer Vitor Hugo Moreau <vitorhmc@ufba.br>

Description Performs a curve fit to 4-parameters Weibull distribution using Metropolis algorithm - Markov chain-Monte Carlo method. Special usage for fitting COVID-19 epidemic data on daily new cases and deaths. Also, builds the 4-parameters Weibull distribution curve using given parameters (shape, scale, location and area).

License Creative Commons Attribution 4.0 International License

Encoding UTF-8

LazyData true

Depends R (>= 2.10)

RoxygenNote 7.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2020-09-17 12:50:03 UTC

R topics documented:

likelihood	2
posterior	3
prior	4
proposalfunction	5
run_metropolis_MCMC	5
weibull4	6
weibull4.fit	8
worldCOVID	9

Index	11
--------------	-----------

likelihood	<i>Calculates the Log Likelihood</i>
------------	--------------------------------------

Description

Likelihood is an Internal Function of the Weibull4 Package

Usage

```
likelihood(x, y, param)
```

Arguments

x	Vector with the x values
y	Vector with the y values
param	Vector with shape, scale, location, area and SD parameters for calculating the log of Likelihood for the weibull4 package

Value

Unitary vector with the sum of the likelihood

Author(s)

Florian Hartig - Theoretical Ecology

References

<https://theoreticalecology.wordpress.com/2010/09/17/metropolis-hastings-mcmc-in-r/>

Examples

```
function (param)
{
  shape <- param[1]
  scale <- param[2]
  loc <- param[3]
  area <- param[4]
  sd <- param[5]
  pred <- weibull4(xi, shape, scale, loc, area)
  singlelikelihoods <- dnorm(yi, mean = pred, sd = sd, log = T)
  sumll <- sum(singlelikelihoods, na.rm = T)
  return(sumll)
}
```

posterior	<i>Calculates the posterior distribution for Metropolis-MCMC</i>
-----------	--

Description

This is an internal function of the weibull4 package

Usage

```
posterior(x, y, param)
```

Arguments

x	Vector with the x values
y	Vector with the y values
param	Vector containing shape, scale, location, area and SD parameters

Value

Vector containing the posterior distribution for Metropolis-MCMC

Author(s)

Florian Hartig - Theoretical Ecology

References

<https://theoreticalecology.wordpress.com/2010/09/17/metropolis-hastings-mcmc-in-r/>

Examples

```
function (param)
{
  return(likelihood(param) + prior(param))
}
```

prior

Calculates the Prior Distribution for Metropolis-MCMC

Description

This is an internal function of the Weibull4 package

Usage

```
prior(param)
```

Arguments

param A vector with shape, scale, location, area and SD parameters

Value

A vector with prior distribution for Metropolis-MCMC

Author(s)

Florian Hartig - Theoretical Ecology

References

<https://theoreticalecology.wordpress.com/2010/09/17/metropolis-hastings-mcmc-in-r/>

Examples

```
function (param)
{
  shape <- param[1]
  scale <- param[2]
  loc <- param[3]
  area <- param[4]
  sd <- param[5]
  shapeprior <- dunif(shape, min = 1, max = 5, log = T)
  scaleprior <- dnorm(scale, sd = scale/2, log = T)
  locprior <- dunif(loc, min = 1, max = loc * 2, log = T)
  areaprior <- dunif(area, min = area/2, max = area * 2, log = T)
  sdprior <- dunif(sd, min = 1, max = sd * 2, log = T)
  return(shapeprior + scaleprior + locprior + areaprior + sdprior)
}
```

proposalfunction	<i>Proposal Distribution for Metropolis-MCMC</i>
------------------	--

Description

This is an internal function of the weibull4 package

Usage

```
proposalfunction(param)
```

Arguments

param Vector containing shape, scale, location, area and SD parameters

Value

Vector containing proposal values for shape, scale, location, area and SD

Author(s)

Florian Hartig - Theoretical Ecology

References

<https://theoreticalecology.wordpress.com/2010/09/17/metropolis-hastings-mcmc-in-r/>

Examples

```
function (param)
{
  return(rnorm(5, mean = param, sd = param * 0.015))
}
```

run_metropolis_MCMC	<i>Runs the Metropolis-MCMC algorithm for weibull4 package</i>
---------------------	--

Description

This is an internal function of the weibull4 package

Usage

```
run_metropolis_MCMC(x, y, startvalue, iterations)
```

Arguments

x	Vector with the x values
y	Vector with the y values
startvalue	Vector with starting shape, scale, location, area and SD values for Metropolis-MCMC calculations
iterations	Number of iterations to be performed in MCMC simulation

Value

Matrix with 5 columns and iterations rows with Markov chains for shape, scale, location, area and SD parameters)

Author(s)

Florian Hartig - Theoretical Ecology

References

<https://theoreticalecology.wordpress.com/2010/09/17/metropolis-hastings-mcmc-in-r/>

Examples

```
function (startvalue, iterations)
{
  chain <- array(dim = c(iterations + 1, 5))
  chain[1, ] <- startvalue
  for (i in 1:iterations) {
    proposal <- proposalfunction(chain[i, ])
    probab <- exp(posterior(proposal) - posterior(chain[i,
      ]))
    if (runif(1) < probab) {
      chain[i + 1, ] <- proposal
    }
    else {
      chain[i + 1, ] <- chain[i, ]
    }
  }
  return(chain)
}
```

weibull4

weibull 4-parameters distribution building function

Description

Builds data for a 4-parameters Weibull distribution of a given x data

Usage

```
weibull4(x=seq(0, 1, length.out = 10), shape=2.5, scale=1, loc=0, area=1)
```

Arguments

x	Vector: data range for calculation of the Weibull distribution. If it is NULL, it will be set to seq(0,1,0.1)
shape	Weibull's shape parameter
scale	Weibull's scale parameter
loc	Weibull's location parameter
area	Weibull's area parameter: area under the PDF curve

Details

This package was specially built to fit COVID-19 data on the number of daily new cases and deaths in countries. So x must be integer. Alternatively, Date format is allowed

Value

Vector: $f(x)$ Weibull distribution's ordinate

Warning

This package is a secondary product of the referred science paper. Please, note that there is no warrants or professional support on its use

Note

Comments, suggestions and doubts must be sent to vitorhmc@ufba.br

Author(s)

Vitor Hugo Moreau, Ph.D.

References

MOREAU, V. H. (2020) Forecast projections of COVID-19 pandemic by Weibull distribution from daily new cases and deaths data, submitted.

Examples

```
## Build a 4-parameters Weibull distribution with given parameters
weibull4(seq(1,100,1), shape=2.5, scale=30, loc=10, area=1000)
## Build and plot 4-parameters Weibull distribution with given parameters
plot(seq(1,100,1), weibull4(seq(1,100,1), 2, 30, 10, 1), type="l")
## Build and plot 4-parameters Weibull distribution with a time series in the abscissa
Date <- seq(Sys.Date(), as.Date("2022-12-31"),1)
plot(Date, weibull4(Date, 1.6, 100, 100, 100), type="l")
```

weibull4.fit	<i>Weibull 4-parameters Metropolis-MCMC non-linear curve fitting function</i>
--------------	---

Description

Package to perform non-linear regression in data on the number of daily new cases and daily new deaths of COVID-19 and other epidemics to the 4-parameters Weibull distribution using Metropolis-Markov Chain-Monte Carlo Simulations (MCMC), as described in Moreau, 2020

Usage

```
weibull4.fit(x, y, shape=NA, scale=NA, loc=NA, area=NA, iter=1000, xmax=0)
```

Arguments

x	Vector: time data (may be date/time or numeric)
y	Vector: observed/measure event
shape	Starting value for Weibull's shape parameters. If it is NA, weibull4.fit will try to calculate it from x and y data.
scale	Starting value for Weibull's scale parameters. If it is NA, weibull4.fit will try to calculate it from x and y data.
loc	Starting value for Weibull's location parameters. If it is NA, weibull4.fit will try to calculate it from x and y data.
area	Starting value for Weibull's area parameters or the area under the PDF curve. If it is NA, weibull4.fit will try to calculate it from x and y data.
iter	Number of iterations to perform Metropolis-MCMC.
xmax	Forecast date to be calculated after x data. It must be in the same format than x.

Details

This package was specially built to fit COVID-19 data on the number of daily new cases and deaths in countries. So x must be integer. Alternatively, Date format is allowed

Value

LIST containning:

Fit data	Matrix with x and y fitted data
Estimates	Matrix containing shape, scale, location, area and SD of the METropolis-MCMC in the row 1; and standard deviation for each parameter in the row 2
Markov chains	Matrix containing the Markov chains for shape, scale, location, area and SD parameters

Warning

This package is a secondary product of the referred science paper. Please, note that there is no warrants or professional support on its use

Note

Comments, suggestions and doubts must be sent to vitorhmc@ufba.br

Author(s)

Vitor Hugo Moreau, Ph.D

References

MOREAU, V. H. (2020) Forecast projections of COVID-19 pandemic by Weibull distribution from daily new cases and deaths data, submitted. Part of the MCMC code came from the Florian Hartig blog: <https://theoreticalecology.wordpress.com/2010/09/17/metropolis-hastings-mcmc-in-r/>

Examples

```
## Perform non-linear curve fitting with World's data for daily new deaths of COVID-19
fit <- weibull4.fit(worldCOVID$date, worldCOVID$new_deaths)
plot(worldCOVID$date, worldCOVID$new_deaths, ylab="World's daily new deaths", xlab="Date")
lines(fit[[1]][,1], fit[[1]][,2], col="red")
```

worldCOVID

World's COVID-19 number of daily new deaths

Description

Dataset for weibull4 package. This dataset contain the number of the World's new daily deaths for COVID-19 to be fitted to the 4-parameters Weibull distribution in weibull4 package.

Usage

```
data("worldCOVID")
```

Format

A data frame with 157 observations on the following 2 variables.

```
date as.Date(c("2019-12-31", "2020-01-01", "2020-01-02", "2020-01-03", "2020-01-04", "2020-01-05", "2020-01-06", "2020-01-07", "2020-01-08", "2020-01-09", "2020-01-10", "2020-01-11", "2020-01-12", "2020-01-13", "2020-01-14", "2020-01-15", "2020-01-16", "2020-01-17", "2020-01-18", "2020-01-19", "2020-01-20", "2020-01-21", "2020-01-22", "2020-01-23", "2020-01-24", "2020-01-25", "2020-01-26", "2020-01-27", "2020-01-28", "2020-01-29", "2020-01-30", "2020-01-31", "2020-02-01", "2020-02-02", "2020-02-03", "2020-02-04", "2020-02-05",
```

```

"2020-02-06", "2020-02-07", "2020-02-08", "2020-02-09", "2020-02-10", "2020-02-11", "2020-
02-12", "2020-02-13", "2020-02-14", "2020-02-15", "2020-02-16", "2020-02-17", "2020-02-
18", "2020-02-19", "2020-02-20", "2020-02-21", "2020-02-22", "2020-02-23", "2020-02-24",
"2020-02-25", "2020-02-26", "2020-02-27", "2020-02-28", "2020-02-29", "2020-03-01", "2020-
03-02", "2020-03-03", "2020-03-04", "2020-03-05", "2020-03-06", "2020-03-07", "2020-03-
08", "2020-03-09", "2020-03-10", "2020-03-11", "2020-03-12", "2020-03-13", "2020-03-14",
"2020-03-15", "2020-03-16", "2020-03-17", "2020-03-18", "2020-03-19", "2020-03-20", "2020-
03-21", "2020-03-22", "2020-03-23", "2020-03-24", "2020-03-25", "2020-03-26", "2020-03-
27", "2020-03-28", "2020-03-29", "2020-03-30", "2020-03-31", "2020-04-01", "2020-04-02",
"2020-04-03", "2020-04-04", "2020-04-05", "2020-04-06", "2020-04-07", "2020-04-08", "2020-
04-09", "2020-04-10", "2020-04-11", "2020-04-12", "2020-04-13", "2020-04-14", "2020-04-
15", "2020-04-16", "2020-04-17", "2020-04-18", "2020-04-19", "2020-04-20", "2020-04-21",
"2020-04-22", "2020-04-23", "2020-04-24", "2020-04-25", "2020-04-26", "2020-04-27", "2020-
04-28", "2020-04-29", "2020-04-30", "2020-05-01", "2020-05-02", "2020-05-03", "2020-05-
04", "2020-05-05", "2020-05-06", "2020-05-07", "2020-05-08", "2020-05-09", "2020-05-10",
"2020-05-11", "2020-05-12", "2020-05-13", "2020-05-14", "2020-05-15", "2020-05-16", "2020-
05-17", "2020-05-18", "2020-05-19", "2020-05-20", "2020-05-21", "2020-05-22", "2020-05-
23", "2020-05-24", "2020-05-25", "2020-05-26", "2020-05-27", "2020-05-28", "2020-05-29",
"2020-05-30", "2020-05-31", "2020-06-01", "2020-06-02", "2020-06-03", "2020-06-04"))
new_deaths c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 0, 3, 11, 0, 9, 15, 15, 25, 25, 26,
38, 43, 46, 46, 57, 65, 66, 72, 73, 86, 89, 97, 108, 97, 255, 13, 144, 142, 106, 98, 139, 116,
119, 112, 104, 156, 79, 64, 38, 57, 64, 58, 67, 72, 84, 80, 102, 104, 97, 228, 208, 273, 326,
350, 442, 366, 751, 608, 813, 959, 1063, 1379, 1711, 1675, 1847, 2303, 2618, 2807, 3481,
3557, 3191, 3983, 4642, 5040, 4993, 6754, 6274, 4703, 5346, 7632, 6548, 7443, 7018, 5983,
5326, 5329, 7655, 10542, 8551, 8447, 6398, 4951, 5397, 7365, 6041, 7354, 5394, 6141, 3862,
4897, 6569, 6577, 5693, 5571, 4749, 3643, 3995, 5970, 6137, 5668, 5183, 4380, 3359, 3553,
5756, 5113, 5282, 5111, 4202, 2854, 3256, 5185, 4756, 5371, 4641, 3927, 1069, 3355, 3955,
5200, 4643, 4694, 4018, 2865, 3117, 4570, 5562)

```

Details

x values are in Date format. So, xmax must be as.Date() too.

Source

"<https://covid.ourworldindata.org/data/owid-covid-data.csv>"

References

Roser M, Ritchie H, Ortiz-Ospina E, Hasel J. Coronavirus Pandemic (COVID-19) [Internet]. 2020 [cited 2020 May 15]. Available from: <https://ourworldindata.org/coronavirus>

Examples

```

data(worldCOVID)
## maybe str(worldCOVID) ; plot(worldCOVID) ...

```

Index

- * **datasets**
 - worldCOVID, [9](#)
- * **likelihood**
 - likelihood, [2](#)
- * **markov**
 - proposalfunction, [5](#)
 - run_metropolis_MCMC, [5](#)
- * **metropolis**
 - posterior, [3](#)
 - run_metropolis_MCMC, [5](#)
- * **posterior**
 - posterior, [3](#)
- * **prior**
 - prior, [4](#)
- * **proposal**
 - proposalfunction, [5](#)
- * **weibull4**
 - likelihood, [2](#)
 - prior, [4](#)
- * **weibull**
 - weibull4, [6](#)
 - weibull4.fit, [8](#)

likelihood, [2](#)

posterior, [3](#)

prior, [4](#)

proposalfunction, [5](#)

run_metropolis_MCMC, [5](#)

weibull4, [6](#)

weibull4.fit, [8](#)

worldCOVID, [9](#)