

# Package ‘rmetallog’

September 19, 2018

**Type** Package

**Title** R Implementation of the Metalog Distribution

**Version** 1.0.0

**Description** Implementation of the metalog distribution in R.

The metalog distribution is a modern, highly flexible, data-driven distribution.

Metalogs are developed by Keelin (2016) <doi:10.1287/deca.2016.0338>.

This package provides functions to build these distributions from raw data.

Resulting metalog objects are then useful for exploratory and probabilistic analysis.

**Imports** lpSolve, ggplot2

**Suggests** devtools, knitr, rmarkdown

**Depends** R (>= 3.1.0)

**BugReports** <https://github.com/isaacfab/rmetallog/issues>

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.0

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Isaac Faber [aut, cre] (<<https://orcid.org/0000-0002-4478-9598>>),  
Jongbin Jung [aut]

**Maintainer** Isaac Faber <[isaacfab@gmail.com](mailto:isaacfab@gmail.com)>

**Repository** CRAN

**Date/Publication** 2018-09-18 23:00:03 UTC

## R topics documented:

dmetallog	2
fishSize	3
metalog	3
plot.metalog	5

pmetalog . . . . .	5
qmetalog . . . . .	6
rmetalog . . . . .	7
summary.metalog . . . . .	8

<b>Index</b>	<b>9</b>
--------------	----------

---

dmetalog	<i>Generate density values with quantiles from a metalog object. This is done through a newtons method approximation.</i>
----------	---

---

### Description

Generate density values with quantiles from a metalog object. This is done through a newtons method approximation.

### Usage

```
dmetalog(m, q, term = 3)
```

### Arguments

m	metalog object created from metalog()
q	y vector of quantiles
term	which metalog distribution to sample from

### Value

A numeric vector of probabilities corresponding to the q quantile vector

### Examples

```
# Load example data
data("fishSize")

# Create a bounded metalog object
myMetalog <- metalog(fishSize$FishSize,
                    bounds=c(0, 60),
                    boundedness = 'b',
                    term_limit = 9,
                    term_lower_bound = 9)

s <- dmetalog(myMetalog,q=c(3,10,25),term = 9)
```

---

fishSize	<i>Fish size measurements from the Pacific Northwest.</i>
----------	---

---

**Description**

Example data set of fish size measurements from the Pacific Northwest, used for illustrating the flexibility of the metalog distribution. This data set is bi-modal because the fish contain two different populations, one salt and two salt runs. The two salt, fish that have gone back to the ocean twice, are larger.

**Usage**

```
fishSize
```

**Format**

A single column data frame with 3474 rows:

**FishSize** Recorded sizes of individual steelhead trout

**Source**

TBD

---

metalog	<i>Fit the metalog distribution to data</i>
---------	---

---

**Description**

Fit the metalog distribution to data

**Usage**

```
metalog(x, bounds = c(0, 1), boundedness = "u", term_limit = 13,
        term_lower_bound = 2, step_len = 0.01, probs = NA,
        fit_method = "any")
```

**Arguments**

x	vector of numeric data
bounds	numeric vector specifying lower or upper bounds, none required if the distribution is unbounded
boundedness	character string specifying unbounded, semi-bounded upper, semi-bounded lower or bounded; accepts values u, su, sl and b (default: 'u')

<code>term_limit</code>	integer between 3 and 30, specifying the number of metalog distributions to generate. Larger term distributions have more flexibility (default: 13)
<code>term_lower_bound</code>	(Optional) the smallest term to generate, used to minimize computation of unwanted terms must be less than <code>term_limit</code> (default is 2)
<code>step_len</code>	(Optional) size of steps to summarize the distribution (between 0 and 0.01) this is only used if the data vector length is greater than 100. Use this if a specific fine grid fit is required. (default is 0.01)
<code>probs</code>	(Optional) probability quantiles, same length as <code>x</code>
<code>fit_method</code>	(Optional) preferred method of fitting distribution: accepts values OLS, LP or any (defaults to any)

### Value

A metalog object with elements

<code>params</code>	A list of the parameters used to create the metalog object
<code>dataValues</code>	a dataframe with the first column the raw data, second column the cumulative probabilities and the third the z vector
<code>Y</code>	The Y matrix values for each quantile and term
<code>A</code>	a dataframe of coefficients for each metalog distribution
<code>M</code>	a dataframe of quantiles (M) and probabilities (m) indexed for each term (i.e. M3,m3 for the third term)
<code>GridPlotCDF()</code>	a function that displays a grid plot of the CDF for each term
<code>VGridPlotPDF()</code>	a function that displays a grid plot of the PDF for each term
<code>Validation</code>	a vector of yes/no indicators of the valid distributions for each term

### Examples

```
# Load example data
data("fishSize")

# Create a bounded metalog object
myMetalog <- metalog(fishSize$FishSize,
                    bounds=c(0, 60),
                    boundedness = 'b',
                    term_limit = 13)
```

---

plot.metalog	<i>Plot of the metalog object</i>
--------------	-----------------------------------

---

**Description**

Plot of the metalog object

**Usage**

```
## S3 method for class 'metalog'  
plot(x, ...)
```

**Arguments**

x	metalog object created using metalog()
...	ignored; included for S3 generic/method consistency

**Value**

A summary plot of the CDF and PDF for each term

**Examples**

```
# Load example data  
data("fishSize")  
  
# Create a bounded metalog object  
myMetalog <- metalog(fishSize$FishSize,  
                    bounds=c(0, 60),  
                    boundedness = 'b',  
                    term_limit = 13)  
  
plot(myMetalog)
```

---

pmetalog	<i>Generate probabilities with quantiles from a metalog object. This is done through a newtons method approximation.</i>
----------	--

---

**Description**

Generate probabilities with quantiles from a metalog object. This is done through a newtons method approximation.

**Usage**

```
pmetalog(m, q, term = 3)
```

**Arguments**

m	metalog object created from metalog()
q	vector of quantiles
term	which metalog distribution to sample from

**Value**

A numeric vector of probabilities corresponding to the q quantile vector

**Examples**

```
# Load example data
data("fishSize")

# Create a bounded metalog object
myMetalog <- metalog(fishSize$FishSize,
                    bounds=c(0, 60),
                    boundedness = 'b',
                    term_limit = 9,
                    term_lower_bound = 9)

s <- pmetalog(myMetalog,q=c(3,10,25),term = 9)
```

---

qmetalog

*Generate quantiles with a probability from a metalog object*


---

**Description**

Generate quantiles with a probability from a metalog object

**Usage**

```
qmetalog(m, y, term = 3)
```

**Arguments**

m	metalog object created from metalog()
y	vector of probabilities
term	which metalog distribution to sample from

**Value**

A numeric vector of quantiles corresponding to the y probability vector

**Examples**

```
# Load example data
data("fishSize")

# Create a bounded metalog object
myMetalog <- metalog(fishSize$FishSize,
                    bounds=c(0, 60),
                    boundedness = 'b',
                    term_limit = 9,
                    term_lower_bound = 9)

s <- qmetalog(myMetalog,y=c(0.25,0.5,0.7),term = 9)
```

---

**rmetalog***Create random samples from an metalog distribution object*

---

**Description**

The rmetalog package implements the metalog distribution in R

**Usage**

```
rmetalog(m, n = 1, term = 3)
```

**Arguments**

m	metalog object created from metalog()
n	number of observations (default is 1)
term	which metalog distribution to sample from

**Value**

A numeric vector of n random samples from a selected distribution

**Examples**

```
# Load example data
data("fishSize")

# Create a bounded metalog object
myMetalog <- metalog(fishSize$FishSize,
                    bounds=c(0, 60),
                    boundedness = 'b',
                    term_limit = 9,
                    term_lower_bound = 9)

s <- rmetalog(myMetalog, n=1000, term = 9)
hist(s)
```

---

summary.metalog      *Summary of the metalog object*

---

**Description**

Summary of the metalog object

**Usage**

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

**Arguments**

object	metalog object created from metalog()
...	ignored; included for S3 generic/method consistency

**Value**

A summary of the object

**Examples**

```
# Load example data  
data("fishSize")  
  
# Create a bounded metalog object  
myMetalog <- metalog(fishSize$FishSize,  
                    bounds=c(0, 60),  
                    boundedness = 'b',  
                    term_limit = 13)  
  
summary(myMetalog)
```



# Index

\*Topic **datasets**

fishSize, [3](#)

dmetalog, [2](#)

fishSize, [3](#)

metalog, [3](#)

plot.metalog, [5](#)

pmetalog, [5](#)

qmetalog, [6](#)

rmetalog, [7](#)

rmetalog-package (rmetalog), [7](#)

summary.metalog, [8](#)