# Package 'ltmix'

April 30, 2019

**Type** Package

**Title** Left-Truncated Mixtures of Gamma, Weibull, and Lognormal Distributions

**Version** 0.2.0

**Author** Martin Blostein <martin.blostein@gmail.com> and Tatjana Miljkovic <miljkot@miamioh.edu>

**Maintainer** Martin Blostein <martin.blostein@gmail.com>

**Description** Mixture modelling of one-dimensional data using combinations of left-truncated Gamma, Weibull, and Lognormal Distributions. Blostein, Martin & Miljkovic, Tatjana. (2019) <10.1016/j.insmatheco.2018.12.001>.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1

**Imports** gtools, pracma

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2019-04-30 16:30:10 UTC

## R topics documented:

`createLtmmObj` *Create an ltmm model object given data and parameters*

## Description

This function is useful for omparing models produced using the ltmix package to models fit using other, or for computing fit criteria and risk measures for a known set of parameters.

## Usage

```
createLtmmObj(x, distributions, trunc, Pars, Pi, npars = NULL)
```

## Arguments

| | |
|---|---|
| `x` | data vector |
| `distributions` | densities to combine |
| `trunc` | left truncation point (optional) |
| `Pars` | list of length G of parameter values |
| `Pi` | vector of length G of component proportions |
| `npars` | Can optionally be used to overwrite the number of free parameters (used in the calculation of AIC & BIC), if the model has additional constraints |

## Value

An ltmm model object

`ltmix` *ltmix: Left-Truncated Mixtures of Gamma, Weibull, and Lognormal Distributions*

## Description

Mixture modelling of one-dimensional data using combinations of left-truncated Gamma, Weibull, and Lognormal Distributions.

`ltmm` *Fit a Left-truncated mixture model (LTMM)*

## Description

This function generates a mixture model combining left-truncated lognormal, gamma, and weibull distributions

## Usage

```
ltmm(x, G, distributions, trunc = NULL, EM_init_method = "emEM",
  EM_starts = 5, init_pars = NULL, init_pi = NULL,
  init_classes = NULL, one_group_reps = 50, eps = 1e-06,
  max.it = 1000, verbose = FALSE)
```

## Arguments

| | |
|---|---|
| `x` | data vector |
| `G` | number of components |
| `distributions` | densities to combine |
| `trunc` | left truncation point (optional) |
| `EM_init_method` | initialization method for EM algorithm |
| `EM_starts` | number of random starts for initialization of EM algorithm. (only for G > 1) |
| `init_pars` | initial parameter values (list of length G) |
| `init_pi` | manually specified initial component proportions (for init_method=specified) |
| `init_classes` | manually specified initial classes. will overwrite init_pars and init_pi |
| `one_group_reps` | number of random starts for each numerical optimization in 1-component model |
| `eps` | stopping tolerance for EM algoithm |
| `max.it` | maximum number of iterations of EM algorithm |
| `verbose` | print information as fitting progresses? |

## Value

An ltmm model object, with the following properties:

**x** Copy of the input data

**distributions** The selected distributions

**trunc** The left truncation value, if specified

**fitted_pdf** The probability density function of the fitted model

**fitted_cfd** The cumulative density function of the fitted model

**VaR** The value-at-risk of the fitted model (function with p taken as onl yargument)

**ES** The expected shortfall of the fitted model (function with p taken as onl yargument)

**G** The number of components in the model

**Pi** The estimated probabilites of component membership

**Pars** The estimated model parameters

**ll** The log-likelihood of the fitted model

**bic** The BIC of the fitted model

**aic** The AIC of the fitted model

**id** The MAP component membership for each observation

**iter** The number of iterations until convergence for the EM algorithm

**npars** The total number of model parameters for the fitted model

**ll.history** The value of log-likelihood at each iteration of the EM algorithm

## Examples

```
x <- secura$Loss

fit <- ltmm(x, G = 2, distributions = c('gamma', 'gamma', 'weibull'), trunc = 1.2e6)

summary(fit)
plot(fit)
```

---

`ltmmCombo` *Fit a Left-truncated mixture model (LTMM)*

---

## Description

This function fits a family of finite mixture models using every combination of the left-truncated lognormal, gamma, and weibull distributions.

## Usage

```
ltmmCombo(x, G, distributions = c("lognormal", "gamma", "weibull"),
  trunc = NULL, EM_init_method = "emEM", EM_starts = 5,
  init_pars = NULL, init_pi = NULL, init_classes = NULL,
  one_group_reps = 50, eps = 1e-06, max.it = 1000, verbose = FALSE,
  parallel = FALSE, cores = NULL, save_each_fit = FALSE)
```

## Arguments

| | |
|---|---|
| `x` | data vector |
| `G` | number of components |
| `distributions` | densities to combine |

| | |
|---|---|
| trunc | left truncation point (optional) |
| EM_init_method | initialization method for EM algorithm |
| EM_starts | number of random starts for initialization of EM algorithm. (only for G > 1) |
| init_pars | initial parameter values (list of length G) |
| init_pi | manually specified initial component proportions (for init_method=specified) |
| init_classes | manually specified initial classes. will overwrite init_pars and init_pi |
| one_group_reps | number of random starts for each numerical optimization in 1-component model |
| eps | stopping tolerance for EM algoithm |
| max.it | maximum number of iterations of EM algorithm |
| verbose | print information as fitting progresses? |
| parallel | fit models in parallel? |
| cores | number of processes used for parallel computation. if NULL detect.cores() used |
| save_each_fit | save each model as it is produced, in a time-stamped directory (safer) |

## Value

An ltmmCombo model object, with the following properties:

**x** Copy of the input data

**distributions** The selected distributions

**combos** List of all combinations of distributions considered

**all.fits** List of all ltmm fit objects

**all.bic** Vector of BIC values for each model

**best.bic.fit** The best ltmm fit by BIC

**best.bic** The best BIC value of all fits

**best.bic.combo** The combination of distributions used for the best fit by BIC

**all.aic** Vector of AIC value for each model

**best.aic.fit** The best ltmm fit by AIC

**best.aic** The best AIC value of all fits

**best.aic.combo** The combination of distributions used for the best fit by AIC

**all.ll** Vector of log-likelihood value for each model

**summary_table** Table summarizing the AIC, BIC, LL, and risk measures for each fitted model

## References

Blostein, Martin & Miljkovic, Tatjana. (2019). On modeling left-truncated loss data using mixtures of distributions. Insurance Mathematics and Economics. 85. 35-46. 10.1016/j.insmatheco.2018.12.001.

## Examples

```
x <- secura$Loss

fits_GL <- ltmmCombo(x, G = 2, distributions = c('gamma', 'lognormal'), trunc = 1.2e6)
summary(fits_GL)
```

secura *The Secura Belgian Re Data*

## Description

"The Secura Belgian Re data set contains automobile claims from 1988 until 2001, which are at least as large as 1,200,000 Euros." (Beirlant, Goegebeur, Segers & Teugels, 2004).

## Usage

```
secura
```

## Format

An object of class `data.frame` with 370 rows and 2 columns.

## References

Beirlant, J., Goegebeur Y., Segers, J., & Teugels, J. Statistics of extremes : theory and applications. Hoboken, NJ: Wiley, 2004. Print.

https://lstat.kuleuven.be/Wiley/

# Index