Package ‘LRMoE’

March 2, 2020

Type Package

Title LRMoE for actuarial loss modelling

Version 0.1.0

Date 2020-02-29

Author Spark Tseung [aut, cre],
Tsz Chai Fung [aut],
Andrei L. Badescu [aut],
Sheldon X. Lin [aut]

Maintainer Spark Tseung <spark.tseung@mail.utoronto.ca>

Description This package is based on the Logit-
Weighted Reduced Mixture of Experts (LRMoE) proposed by Fung et al. (2019),
which is a flexible framework actuarial loss modelling.

License GPL-3

URL https://github.com/sparktseung/LRMoE

BugReports https://github.com/sparktseung/LRMoE/issues

Encoding UTF-8

LazyData false

Depends R (>= 3.5.1)

Imports actuar,
copula,
expint,
EnvStats,
ggplot2,
matrixStats,
NMOF,
reshape2,
rmutil,
statmod,
stats

RoxygenNote 7.0.2

R topics documented:

`cluster.mm.frequency` *Initializes parameter for frequency distributions using CMM.*

Description

Initializes parameter for frequency distributions using CMM.

Usage

```
cluster.mm.frequency(Y, cluster)
```

Arguments

`Y`	A vector of response variables.
`cluster`	The `cluster` list vector returned by `kmeans`

Value

A list of parameter initialization.

`cluster.mm.severity` *Initializes parameter for severity distributions using CMM.*

Description

Initializes parameter for severity distributions using CMM.

Usage

```
cluster.mm.severity(Y, cluster)
```

Arguments

`Y`	A vector of response variables.
`cluster`	The `cluster` list vector returned by `kmeans`

Value

A list of parameter initialization.

`dataset.simulator` *Simulate y, given a fixed covariate matrix X and a model*

Description

Simulate y, given a fixed covariate matrix X and a model

Usage

```
dataset.simulator(X, alpha, comp.dist, zero.prob, params.list)
```

Arguments

`X`	A N*P matrix of of covariates. The first column must be 1. Each row may be different.
`alpha`	A g*P matrix of logit regression coefficients.
`comp.dist`	A d*g matrix of strings, describing component distributions by dimension and by component.
`zero.prob`	A d*g matrix of numbers between 0 and 1, describing zero probability masses by dimension and by component.
`paramas.list`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the initial parameter guess for the corresponding `comp.dist`.

Value

A matrix of simulated values, where each row represents a policyholder and each column a dimension of the response variable.

DemoData	*Demo data for* `LRMoE`.

Description

- `X`: A matrix of covariates required by the LRMoE fitting function (complete).
- `Y`: A matrix of response required by the LRMoE fitting function (complete).
- `X.obs`: A matrix of covariates required by the LRMoE fitting function (after truncation and censoring).
- `Y.obs`: A matrix of response required by the LRMoE fitting function (after truncation and censoring).

dgammacount.new	*Modified* `GammaCount` *pmf for better numerical accuracy.*

Description

Modified `GammaCount` pmf for better numerical accuracy.

Usage

```
dgammacount.new(y, m, s, log = FALSE)
```

Arguments

`y`	Vector of gamma count values.
`m, s`	Paramaters of Gamma count distribution.

See Also

`GammaCount`.

LRMoE.fit	*Fit an LRMoE model.*

Description

Fit an LRMoE model.

Usage

```
LRMoE.fit(
  Y,
  X,
  n.comp = 2,
  comp.dist = NULL,
  alpha.init = NULL,
  zero.init = NULL,
  params.init = NULL,
  penalty = TRUE,
  hyper.alpha = NULL,
  hyper.params = NULL,
  eps = 0.001,
  alpha.iter.max = 5,
  ecm.iter.max = 200,
  grad.jump = TRUE,
  grad.period = 5,
  grad.seq = 2^(seq(8) - 1) - 1,
  print = TRUE
)
```

Arguments

`Y`	A N*4d matrix of numerics, where N is sample size and d is the dimension of each obsevation. Each block of four columns should be organized as `(tl,yl,yu,tu)`, representing the truncation lower bound, censoring lower bound, censoring upper bound and truncation upper bound.
`X`	A N*P matrix of numerics, where P is the number of covariates. The first column of `X` should be 1, which is the intercept.
`n.comp`	A numeric which indicates the number of experts desired to fit the data. Default value is 2.
`comp.dist`	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of `Y`, while the columns represent the component distributions. See below for more details.
`alpha.init`	A g*P matrix of numerics, which contains initial guess of the logit regression coefficients. The last row should all be zero, representing the default latent class. If no initialization is provided, all coefficients are set to zero.
`zero.init`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions. If the corresponding entry in `comp.dist` is not zero-inflated, zero value must by supplied.
`params.init`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the initial parameter guess for the corresponding `comp.dist`.
`penalty`	TRUE/FALSE: whether the parameters are penalized for their magnitude. Default (and recommended) is TRUE.
`hyper.alpha`	A g*P matrix of numerics, which contains penalties for `alpha.init`. If `penalty=T` but no `hyper.alpha` is provided, a constant is used.
`hyper.params`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the corresponding penalty for `params.init`.

`eps`	Stopping criteria for loglikelihood convergence. Default is `1e-03`.
`alpha.iter.max`	Maximum number of iterations for updating alpha. Defauls is 5. See also `alpha.m.recur`.
`ecm.iter.max`	Maximum number of iterations for ECM. Default is 200.
`grad.jump`	TRUE/FALSE: whether to use an approximated gradient jump to speed up convergence.
`grad.period`	How often should `grad.jump` occur. Default is every 5 iterations.
`grad.seq`	How are the gradient sequence selected. Default is `2^(seq(8)-1)-1`.
`print`	TRUE/FALSE: whether paramater updates are printed on screen. Default is TRUE.

`LRMoE.loglik` *Computes the loglikelihood of LRMoE, given X, Y and a fitted model.*

Description

Computes the loglikelihood of LRMoE, given X, Y and a fitted model.

Usage

```
LRMoE.loglik(X, Y, model, penalty = TRUE, hyper.alpha, hyper.params)
```

Arguments

`X` A N*P matrix of covariates.

`Y` A N*d matrix of response.

`model` A list of parameters specifying an LRMoE model, including

- `alpha`: A g*P matrix, where g is the number of components and P is the number of covariates.
- `comp.dist`: A d*g matrix of strings, describing component distributions by dimension and by component.
- `zero.prob`: A d*g matrix of numbers between 0 and 1, describing zero probability masses by dimension and by component.
- `params.list`: A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the initial parameter guess for the corresponding `comp.dist`.

`penalty` TRUE/FALSE, which indicates whether parameter penalty should be applied. Default (and recommended) is TRUE.

`hyper.alpha` A numeric, which penalizes the magnitude of `alpha`.

`hyper.params` A list of length d. Each element is a sublist of length g. Each element of a sublist is a vector of numerics, which penalizes expert parameters. See also `expert.loglik.pen.dim.comp`.

Value

Loglikelihood (with and without penalty), AIC and BIC.

mgammacount — *Calculates moments of GammaCount using finite approximation.*

Description

Calculates moments of GammaCount using finite approximation.

Usage

```
mgammacount(order, m, s, tol = 1e-10)
```

Arguments

order	A vector of positive power indices.
m, s	Paramaters of Gamma count distribution.
tol	Cut-off probability threshold. Values above (1-tol) are discarded.

Value

A vector of Gamma count distribution moments.

See Also

GammaCount.

pgammacount.new — *Modified GammaCount cdf for better numerical accuracy.*

Description

Modified GammaCount cdf for better numerical accuracy.

Usage

```
pgammacount.new(q, m, s, log = FALSE)
```

Arguments

q	Vector of quantiles.
m, s	Paramaters of Gamma count distribution.

See Also

GammaCount.

`plot.dataset.class.prob`
Plots a stacked bar chart of most likely latent class proportion, given a matrix of covariates.

Description

Plots a stacked bar chart of most likely latent class proportion, given a matrix of covariates.

Usage

```
## S3 method for class 'dataset.class.prob'
plot(X, alpha, title = "Proportion of Latent Classes")
```

Arguments

X	A matrix of covariates.
alpha	A matrix of logit regression coeffiecients.
title	A text string for plot title.

Value

A ggplot2 object.

See Also

LRMoE.fit, predict.class

`plot.dataset.prob.posterior`
Plots two stacked bar charts of most likely latent class proportions, which contrasts prior and posterior latent class proportions, given a covariate matrix.

Description

Plots two stacked bar charts of most likely latent class proportions, which contrasts prior and posterior latent class proportions, given a covariate matrix.

Usage

```
## S3 method for class 'dataset.prob.posterior'
plot(
  X,
  Y,
  alpha,
  comp.dist,
  zero.prob,
  params.list,
  title = "Proportion of Latent Classes"
)
```

Arguments

`X`	A matrix of covariates.
`Y`	A matrix of observed responses for `X`.
`alpha`	A matrix of logit regression coefficients.
`comp.dist`	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of `Y`, while the columns represent the component distributions.
`zero.prob`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
`title`	A text string for plot title.
`params.init`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding `comp.dist`.

Value

A `ggplot2` object.

See Also

`LRMoE.fit`, `predict.class.posterior`

`plot.ind.class.prob` *Plots a stacked bar chart of latent class probabilities, given a covariate vector.*

Description

Plots a stacked bar chart of latent class probabilities, given a covariate vector.

Usage

```
## S3 method for class 'ind.class.prob'
plot(X, alpha, title = "Prediction of Latent Classes")
```

Arguments

`X`	A vector of covariates for one policyholder.
`alpha`	A matrix of logit regression coeffiecients.
`title`	A text string for plot title.

Value

A `ggplot2` object.

See Also

`LRMoE.fit`, `predict.class.prob`

`plot.ind.class.prob.posterior` *Plots two stacked bar charts of latent class probabilities, which contrasts prior and posterior latent class probabilities, given a covariate vector.*

Description

Plots two stacked bar charts of latent class probabilities, which contrasts prior and posterior latent class probabilities, given a covariate vector.

Usage

```
## S3 method for class 'ind.class.prob.posterior'
plot(
  X,
  Y,
  alpha,
  comp.dist,
  zero.prob,
  params.list,
  title = "Prediction of Latent Classes"
)
```

Arguments

`X`	A matrix of covariates.
`Y`	A matrix of observed responses for `X`.
`alpha`	A matrix of logit regression coefficients.
`comp.dist`	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of `Y`, while the columns represent the component distributions.
`zero.prob`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
`params.list`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding `comp.dist`.
`title`	A text string for plot title.

Value

A `ggplot2` object.

See Also

`LRMoE.fit`, `predict.class.prob.posterior`

`plot.ind.fitted.dist` *Plot the fitted density, given a vector of covariates and a model*

Description

Plot the fitted density, given a vector of covariates and a model

Usage

```
## S3 method for class 'ind.fitted.dist'
plot(
  X,
  alpha,
  comp.dist,
  zero.prob,
  params.list,
  plot.dim = 1,
  plot.lim = NULL
)
```

Arguments

`X`	A vector of covariates.
`alpha`	A g*P matrix of numerics, which contains initial guess of the logit regression coefficients. The last row should all be zero, representing the default latent class.
`comp.dist`	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of `Y`, while the columns represent the component distributions.
`zero.prob`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
`params.list`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding `comp.dist`.
`plot.dim`	A numeric indicating which dimention of y to plot.
`plot.lim`	Upper bound of y for plotting. Default is 50, if no value is provided.

Value

A `ggplot2` object.

See Also

`LRMoE.fit`

`predict.class` *Predict the most likely latent class, given a fixed covariate matrix X and a model.*

Description

Predict the most likely latent class, given a fixed covariate matrix X and a model.

Usage

```
## S3 method for class 'class'
predict(X, alpha)
```

Arguments

X	A matrix of covariates.
alpha	A matrix of logit regression coefficients.

Value

A vector of the most likely latent class by observation.

See Also

LRMoE.fit.

`predict.class.posterior` *Predict the most likely latent class, given a fixed covariate matrix X and a model.*

Description

Predict the most likely latent class, given a fixed covariate matrix X and a model.

Usage

```
## S3 method for class 'class.posterior'
predict(X, Y, alpha, comp.dist, zero.prob, params.list)
```

Arguments

X	A matrix of covariates.
Y	A matrix of observed responses for X.
alpha	A matrix of logit regression coefficients.
comp.dist	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of Y, while the columns represent the component distributions.

`zero.prob`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
`params.init`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding `comp.dist`.

Value

A vector of the most likely latent class by observation.

See Also

`LRMoE.fit`.

`predict.class.prob` *Predict the latent class probabilities, given a fixed covariate matrix X and a model.*

Description

Predict the latent class probabilities, given a fixed covariate matrix X and a model.

Usage

```
## S3 method for class 'class.prob'
predict(X, alpha)
```

Arguments

`X`	A matrix of covariates.
`alpha`	A matrix of logit regression coefficients.

Value

A matrix of latent class probabilities by observation and by component.

See Also

`LRMoE.fit`.

`predict.class.prob.posterior` *Predict the posterior latent class probabilities, given a fixed covariate matrix X, Y and a model.*

Description

Predict the posterior latent class probabilities, given a fixed covariate matrix X, Y and a model.

Usage

```
## S3 method for class 'class.prob.posterior'
predict(X, Y, alpha, comp.dist, zero.prob, params.list)
```

Arguments

`X`	A matrix of covariates.
`Y`	A matrix of observed responses for `X`.
`alpha`	A matrix of logit regression coefficients.
`comp.dist`	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of `Y`, while the columns represent the component distributions.
`zero.prob`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
`params.list`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding `comp.dist`.

Value

A matrix of latent class probabilities by observation and by component.

See Also

`LRMoE.fit`.

`predict.cte` *Predict the CTE of y, given a fixed covariate matrix X and a model.*

Description

Predict the CTE of y, given a fixed covariate matrix X and a model.

Usage

```
## S3 method for class 'cte'
predict(X, alpha, comp.dist, zero.prob, params.list, prob = NULL)
```

Arguments

`X`	A matrix of covariates.
`alpha`	A matrix of logit regression coefficients.
`comp.dist`	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of `Y`, while the columns represent the component distributions.
`zero.prob`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
`params.list`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding `comp.dist`.
`probs`	A vector of probabilities. Default is a vector of length d of 0.95, if no value is provided.

Value

A matrix of CTE.

See Also

`LRMoE.fit`, `predict.quantile`.

`predict.excess` *Predict the excess mean of y, given a fixed covariate matrix X, a model and a vector of limits by dimension.*

Description

Predict the excess mean of y, given a fixed covariate matrix X, a model and a vector of limits by dimension.

Usage

```
## S3 method for class 'excess'
predict(X, alpha, comp.dist, zero.prob, params.list, limit)
```

Arguments

`X`	A matrix of covariates.
`alpha`	A matrix of logit regression coefficients.
`comp.dist`	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of `Y`, while the columns represent the component distributions.
`zero.prob`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
`params.list`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding `comp.dist`.
`limit`	A vector of limit to apply to each dimension of y.

Value

A matrix of excess mean values by observation and by dimension. This is equan to `predict.mean -predict.limit`.

See Also

`LRMoE.fit`, `predict.limit`.

`predict.limit` *Predict the limited mean of y, given a fixed covariate matrix X, a model and a vector of limits by dimension.*

Description

Predict the limited mean of y, given a fixed covariate matrix X, a model and a vector of limits by dimension.

Usage

```
## S3 method for class 'limit'
predict(X, alpha, comp.dist, zero.prob, params.list, limit)
```

Arguments

`X`	A matrix of covariates.
`alpha`	A matrix of logit regression coefficients.
`comp.dist`	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of `Y`, while the columns represent the component distributions.
`zero.prob`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
`params.list`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding `comp.dist`.
`limit`	A vector of limit to apply to each dimension of y.

Value

A matrix of limited mean values by observation and by dimension. Calculation is done for severity distributions only. `NA` values are returned for frequency distributions.

See Also

`LRMoE.fit`, `GammaSupp`.

predict.mean *Predict the mean of y, given a fixed covariate matrix X and a model.*

Description

Predict the mean of y, given a fixed covariate matrix X and a model.

Usage

```
## S3 method for class 'mean'
predict(X, alpha, comp.dist, zero.prob, params.list)
```

Arguments

X	A matrix of covariates.
alpha	A matrix of logit regression coefficients.
comp.dist	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of Y, while the columns represent the component distributions.
zero.prob	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
params.list	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding comp.dist.

Value

A matrix of mean values by observation and by dimension.

See Also

LRMoE.fit.

predict.quantile *Predict the VaR of y, given a fixed covariate matrix X and a model.*

Description

Predict the VaR of y, given a fixed covariate matrix X and a model.

Usage

```
## S3 method for class 'quantile'
predict(X, alpha, comp.dist, zero.prob, params.list, prob = NULL)
```

Arguments

`X`	A matrix of covariates.
`alpha`	A matrix of logit regression coefficients.
`comp.dist`	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of `Y`, while the columns represent the component distributions.
`zero.prob`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
`params.list`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding `comp.dist`.
`probs`	A vector of probabilities. Default is a vector of length d of 0.95, if no value is provided.

Value

A matrix of VaR.

See Also

`LRMoE.fit`, `predict.cte`.

`predict.var` *Predict the variance of y, given a fixed covariate matrix X and a model.*

Description

Predict the variance of y, given a fixed covariate matrix X and a model.

Usage

```
## S3 method for class 'var'
predict(X, alpha, comp.dist, zero.prob, params.list)
```

Arguments

`X`	A matrix of covariates.
`alpha`	A matrix of logit regression coefficients.
`comp.dist`	A d*g matrix of strings, which specify the component distributions to fit. The rows represent the dimensions of `Y`, while the columns represent the component distributions.
`zero.prob`	A d*g matrix of numerics in (0,1), which specify the probability mass at zero for component distributions.
`params.list`	A list of length d, where each element is a sublist of length g. Each sublist contains one numeric vector, which is the parameter value for the corresponding `comp.dist`.

Value

A matrix of variance by observation and by dimension.

See Also

`LRMoE.fit`.

Index