Package ‘CoMiRe’

January 20, 2025

Type Package

Title Convex Mixture Regression

Version 0.8

Date 2023-08-22

Author Antonio Canale [aut, cre], Daniele Durante [ctb], Arianna Falcioni [aut], Luisa Galtarossa [aut], Tommaso Rigon [ctb]

Maintainer Antonio Canale <`antonio.canale@unipd.it`>

Description Posterior inference under the convex mixture regression (CoMiRe) models introduced by Canale, Durante, and Dunson (2018) <doi:10.1111/biom.12917>.

License GPL-2

NeedsCompilation yes

Imports Rcpp (>= 1.0.5), KernSmooth, ggplot2, gtools, mvtnorm, splines2 (>= 0.3.1), truncnorm, rlang

Depends R (>= 4.0)

LinkingTo RcppArmadillo, Rcpp (>= 1.0.5)

RoxygenNote 7.1.2

Encoding UTF-8

Repository CRAN

Date/Publication 2023-08-23 09:10:06 UTC

Contents

`CoMiRe-package` *Convex Mixture Regression*

Description

Posterior inference under the convex mixture regression (CoMiRe) models introduced by Canale, Durante, and Dunson (2018) <doi:10.1111/biom.12917>.

Details

The `CoMiRe` package implements the convex mixture regresion approach of Canale, Durante, and Dunson (2018) and some extensions to deal with binary response variables or to account for the presence of continuous and categorical confunders. Estimation is conducted via Gibbs sampler. Posterior plots for inference and goodness-of-fit tests are also available.

Author(s)

Antonio Canale [aut, cre], Daniele Durante [ctb], Arianna Falcioni [aut], Luisa Galtarossa [aut], Tommaso Rigon [ctb] Maintainer: Antonio Canale <antonio.canale@unipd.it>

References

Canale, A., Durante, D., and Dunson, D. (2018), Convex Mixture Regression for Quantitative Risk Assessment, Biometrics, 74, 1331-1340

`add.risk` *Additional risk function*

Description

Additional risk function estimated from the object `fit`

Usage

```
add.risk(y, x, fit, mcmc, a, alpha=0.05,
x.grid=NULL, y.grid=NULL)
```

Arguments

`y`	optional numeric vector for the response used in `comire.gibbs`. If `y` is missing, `y.grid` must be provided.
`x`	numeric vector for the covariate relative to the dose of exposure used in `comire.gibbs`.
`fit`	the output of `comire.gibbs`. an object of the class `classCoMiRe`.
`mcmc`	a list giving the MCMC parameters.
`a`	threshold of clinical interest for the response variable
`alpha`	level of the credible bands.
`x.grid`	optional numerical vector giving the actual values of the grid for x for plotting the additional risk function. If $x.gird$ is not provided, standard grids are automatically used.
`y.grid`	optional numerical vector giving the actual values of the grid for y for plotting the additional risk function. If $y.gird$ is not provided, standard grids are automatically used.

Value

A list of arguments for generating posterior output. It contains:

- `mcmc.risk` a matrix containing in the lines the MCMC chains, after thinning, of the additional risk function over `x.grid`, in the columns.
- `summary.risk` a data frame with four variables: the posterior means of the additional risk function over `x.grid`, the respective $\alpha/2$ and $1-\alpha/2$ quantiles, and `x.grid`.

Author(s)

Antonio Canale, Arianna Falcioni

Examples

```
{
data(CPP)
attach(CPP)

n <- NROW(CPP)
J <- H <- 10

premature <- as.numeric(gestage<=37)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## too few iterations to be meaningful. see below for safer and more comprehensive results

mcmc <- list(nrep=10, nb=2, thin=1, ndisplay=4)

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)
```

```
fit.dummy <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=1, max.x=180)

risk.data <- add.risk(y = gestage, x = dde, fit = fit.dummy, mcmc = mcmc,
    a = 37, x.grid = seq(0, max(dde), length = 100))
riskplot(risk.data$summary.risk, xlab="DDE", x = dde, xlim = c(0,150))


## safer procedure with more iterations (it may take some time)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## Fit the model for continuous y

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit1 <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=5, max.x=180)

risk.data <- add.risk(y = gestage, x = dde, fit = fit1, mcmc = mcmc,
a = 37, x.grid = seq(0, max(dde), length = 100))
riskplot(risk.data$summary.risk, xlab="DDE", x = dde, xlim = c(0,150))


}
```

`as.classCoMiRe` *classCoMiRe class constructor*

Description

A constructor for the `classCoMiRe` class. The class `classCoMiRe` is a named list containing the output of the posterior estimation of CoMiRe model implemented in `comire.gibbs`

Usage

```
as.classCoMiRe(call = NULL, out = NULL, z = NULL, z.val = NULL, f0 = NULL, f1 = NULL,
nrep, nb, bin = FALSE, univariate = TRUE)
```

Arguments

`call`	a formula for `comire.gibbs`.
`out`	an output of `comire.gibbs`.
`z`	optional numeric vector or matrix for the confounding covariates.
`z.val`	optional numeric vector containing a fixed value of interest for each of the confounding covariates to be used for the plots. Default value is `mean(z)` for numeric covariates or the mode for factorial covariates.

`f0, f1`	optional matrices containing simulated values of the mixture densities at low and high dose exposure; default values are simulated with `comire.gibbs`. It is possible to change these for differente fixed values of `z`: see `predict_new_z` function.
`nrep`	integer giving the total number of iterations used in `comire.gibbs`.
`nb`	integer giving the number of burn-in iterations used in `comire.gibbs`.
`bin`	logical. It is `TRUE` if `y` is drawn for a binomial distribution.
`univariate`	logical. It is `TRUE` if the model is univariate.

Author(s)

Antonio Canale, Arianna Falcioni

`betaplot` *$\beta(x)$ plot*

Description

Posterior mean (continuous lines) and pointwise credible bands (shaded areas) for $\beta(x)$.

Usage

```
betaplot(x, fit, x.grid = NULL, xlim = c(0, max(x)), xlab = "x")
```

Arguments

`x`	numeric vector for the covariate relative to the dose of exposure used in `comire.gibbs`.
`fit`	the output of `comire.gibbs` opportunely trasformed in `classCoMiRe` class.
`x.grid`	optional numerical vector giving the actual values of the grid for x for plotting $\beta(x)$. If `x.gird` is not provided, standard grids are automatically used.
`xlim`	numeric vectors of length 2, giving the x coordinates ranges for the plot.
`xlab`	the title of the x axis.

Author(s)

Antonio Canale

Examples

```
{
data(CPP)
attach(CPP)

n <- NROW(CPP)
J <- H <- 10
```

```
premature <- as.numeric(gestage<=37)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## too few iterations to be meaningful. see below for safer and more comprehensive results

mcmc <- list(nrep=10, nb=2, thin=1, ndisplay=4)

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit.dummy <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=1, max.x=180)

betaplot(x=dde, fit=fit.dummy, x.grid=seq(0,180, length=100), xlim=c(0,150))


## safer procedure with more iterations (it may take some time)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## Fit the model for continuous y

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit1 <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=5, max.x=180)


betaplot(x=dde, fit=fit1, x.grid=seq(0,180, length=100), xlim=c(0,150))


}
```

`BMD` *Benchmark dose*

Description

Benchmark dose associated to a particular risk

Usage

```
BMD(level, risk, x, alpha=0.05)
```

Arguments

`level`	dose level of interest.
`risk`	`summary.risk$mcmc.risk` from the output of `add.risk` function.

`x`	numeric vector for the covariate relative to the dose of exposure used in `comire.gibbs`.
`alpha`	level of the credible bands.

Value

A dataframe containing as variables:

- `q` the dose level of interest.
- `BMD` the benchmark dose.
- `low` lower credible limit.
- `upp` upper credible limit.
- `BMDL` a more conservative benchmark dose.

Author(s)

Antonio Canale

Examples

```
{
data(CPP)
attach(CPP)

n <- NROW(CPP)
J <- H <- 10

premature <- as.numeric(gestage<=37)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## too few iterations to be meaningful. see below for safer and more comprehensive results

mcmc <- list(nrep=10, nb=2, thin=1, ndisplay=4)

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit.dummy <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=1, max.x=180)

risk.data <- add.risk(y = gestage, x = dde, fit = fit.dummy, mcmc = mcmc,
    a = 37, x.grid = seq(0, max(dde), length = 100))
bmd.data <- BMD(seq(0,.20, length=50), risk.data$mcmc.risk,
x=seq(0,max(dde), length=100), alpha=0.05)
bmd.plot(bmd.data)


## safer procedure with more iterations (it may take some time)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)
```

```
## Fit the model for continuous y

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit <- comire.gibbs(gestage, dde, family="continuous",
                    mcmc=mcmc, prior=prior, seed=5, max.x=180)


risk.data <- add.risk(y = gestage, x = dde, fit = fit, mcmc = mcmc,
a = 37, x.grid = seq(0, max(dde), length = 100))
bmd.data <- BMD(seq(0,.20, length=50), risk.data$mcmc.risk,
x=seq(0,max(dde), length=100), alpha=0.05)
bmd.plot(bmd.data)


}
```

`bmd.plot` *Benchmark dose plot*

Description

Posterior mean (continuous lines) and pointwise credible bands (shaded areas) for the benchmark dose in function of the increase in risk.

Usage

```
bmd.plot(bmd.data)
```

Arguments

`bmd.data`	output of `BMD` function.

Author(s)

Antonio Canale

Examples

```
{
data(CPP)
attach(CPP)

n <- NROW(CPP)
J <- H <- 10

premature <- as.numeric(gestage<=37)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)
```

```
## too few iterations to be meaningful. see below for safer and more comprehensive results

mcmc <- list(nrep=10, nb=2, thin=1, ndisplay=4)

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit.dummy <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=1, max.x=180)

risk.data <- add.risk(y = gestage, x = dde, fit = fit.dummy, mcmc = mcmc,
    a = 37, x.grid = seq(0, max(dde), length = 100))
bmd.data <- BMD(seq(0,.20, length=50), risk.data$mcmc.risk,
x=seq(0,max(dde), length=100), alpha=0.05)
bmd.plot(bmd.data)


## safer procedure with more iterations (it may take some time)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## Fit the model for continuous y

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=5, max.x=180)

risk.data <- add.risk(y = gestage, x = dde, fit = fit, mcmc = mcmc,
a = 37, x.grid = seq(0, max(dde), length = 100))
bmd.data <- BMD(seq(0,.20, length=50), risk.data$mcmc.risk,
x=seq(0,max(dde), length=100), alpha=0.05)
bmd.plot(bmd.data)


}
```

`comire.gibbs` *Gibbs sampler for CoMiRe model*

Description

Posterior inference via Gibbs sampler for CoMiRe model

Usage

```
comire.gibbs(y, x, z = NULL, family = 'continuous',
       grid = NULL, mcmc, prior,
       state = NULL, seed, max.x = max(x), z.val = NULL, verbose = TRUE)
```

Arguments

`y`	numeric vector for the response: when `family="continuous"` `y` must be a numeric vector; if `family="binary"` `y` must assume values `0` or `1`.
`x`	numeric vector for the covariate relative to the dose of exposure.
`z`	numeric vector for the confunders; a vector if there is only one confounder or a matrix for two or more confunders
`family`	type of `y`. This can be `"continuous"` or `"binary"`. Default `"continuous"`.
`grid`	a list giving the parameters for plotting the posterior mean density and the posterior mean $\beta(x)$ over finite grids if `family="continuous"` and `z=NULL`. It must include the following values: • `grids`, logical value (if `TRUE` the provided grids are used, otherwise standard grids are automatically used); • `xgrid` and `ygrid`, numerical vectors with the actual values of the grid for y and x.
`mcmc`	a list giving the MCMC parameters. It must include the following integers: `nb` giving the number of burn-in iterations, `nrep` giving the total number of iterations, `thin` giving the thinning interval, `ndisplay` giving the multiple of iterations to be displayed on screen while the algorithm is running (a message will be printed every `ndisplay` iterations).
`prior`	a list containing the values of the hyperparameters. If `family = "continuous"`, it must include the following values: • `mu.theta`, the prior mean μ_θ for each location parameter θ_{0h} and θ_1, • `k.theta`, the prior variance k_θ for each location paramter θ_{0h} and θ_1, • `mu.gamma` (if `p` confounding covariates are included in the model) a `p`-dimentional vector of prior means μ_γ of the parameters γ corresponding to the confounders, • `k.gamma`, the prior variance k_γ for parameter corresponding to the confounding covariate (if `p=1`) or `sigma.gamma` (if `p>1`), that is the covariance matrix Σ_γ for the parameters corresponding to the `p` confounding covariates; this must be a symmetric positive definite matrix. • `eta`, numeric vector of size `J` for the Dirichlet prior on the beta basis weights, • `alpha`, prior for the mixture weights, • `a` and `b`, prior scale and shape parameter for the gamma distribution of each precision parameter, • `J`, parameter controlling the number of elements of the I-spline basis, • `H`, total number of components in the mixture at x_0. If `family="binary"` it must include the following values: • `eta`, numeric vector of size `J` for the Dirichlet prior on the beta basis weights, • `a.pi0` and `b.pi0`, the prior parameters of the prior beta distribution for π_0, • `J`, parameter controlling the number of elements of the Ispline basis.

`state`	if `family="continuous"`, a list giving the current value of the parameters. This list is used if the current analysis is the continuation of a previous analysis or if we want to start the MCMC algorithm from some particular value of the parameters.
`seed`	seed for random initialization.
`max.x`	maximum value allowed for `x`.
`z.val`	optional numeric vector containing a fixed value of interest for each of the confounding covariates to be used for the plots. Default value is `mean(z)` for numeric covariates or the mode for factorial covariates.
`verbose`	logical, if `TRUE` a message on the status of the MCMC algorithm is printed to the console. Default is `TRUE`.

Details

The function fit a convex mixture regression (`CoMiRe`) model (Canale, Durante, Dunson, 2018) via Gibbs sampler. For continuous outcome $y \in \mathcal{Y}$, adverse esposure level $x \in \mathcal{X}$ and no confunding variables, one can set `family = 'continuous'` and `z = NULL` and fit model
$f_x(y) = \{1-\beta(x)\} \sum_{h=1}^{H} \nu_{0h} \phi(y; \theta_{0h}, \tau_{0h}^{-1}) + \beta(x)\phi(y; \theta_{\infty}, \tau_{\infty}^{-1})$;

where $\beta(x) = \sum_{j=1}^{J} \omega_j \psi_j(x), x \geq 0$, is a a monotone nondecreasing interpolation function, constrained between 0 and 1 and $\psi_1, ..., \psi_J$ are monotone nondecreasing I-splines basis.
If $p \geq 1$ confounding covariates $z \in \mathcal{Z}$ are available, passing the argument `z` the function fits model

$f(y; x, z) = \{1-\beta(x)\} f_0(y; z) + \beta(x) f_{\infty}(y; z)$;

where:
$f_0(y; z) = \sum_{h=1}^{H} \nu_{0h} \phi(y; \theta_{0h} + z^{\mathsf{T}}\gamma, \tau_{0h}^{-1})$, and $f_{\infty}(y; z) = \phi(y; \theta_{\infty} + z^{\mathsf{T}}\gamma, \tau_{\infty}^{-1})$.

Finally, if y is a binary response, one can set `family = 'binary'` and fit model

$p_x(y) = (\pi_x)^y (1-\pi_x)^{1-y}$;

where $\pi_x = P(Y = 1|x)$ is $\pi_x = \{1-\beta(x)\}\pi_0 + \beta(x)\pi_{\infty}$.

Value

An object of the class `classCoMiRe`, i.e. a list of arguments for generating posterior output. It contains:

- `call`the model formula
- `post.means` a list containing the posterior mean density beta over the grid, of all the mixture parameters and, if `family = "continuous"` and `z = NULL`, of f_0 and f_{inf} over the `y.grid`.
- `ci` a list containing the 95% credible intervals for all the quantities stored in `post.means`.
- `mcmc` a list containing all the MCMC chains.
- `z` the same of the input
- `z.val` the same of the input

- `f0,f1` MCMC replicates of the density in the two extremes (only if `family = 'continuous'`)
- `nrep,nb` the same values of the list `mcmc` in the input arguments
- `bin` logical, equal to `TRUE` if `family = 'binary'`
- `univariate` logical, equal to `TRUE` if `z` is null or a vector

Author(s)

Antonio Canale [aut, cre], Daniele Durante [ctb], Arianna Falcioni [aut], Luisa Galtarossa [aut], Tommaso Rigon [ctb]

References

Canale, A., Durante, D., and Dunson, D. (2018), Convex Mixture Regression for Quantitative Risk Assessment, Biometrics, 74, 1331-1340

Galtarossa, L., Canale, A., (2019), A Convex Mixture Model for Binomial Regression, Book of Short Papers SIS 2019

Examples

```
{

data(CPP)
attach(CPP)

n <- NROW(CPP)
J <- H <- 10

premature <- as.numeric(gestage<=37)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## too few iterations to be meaningful. see below for safer and more comprehensive results

mcmc <- list(nrep=10, nb=2, thin=1, ndisplay=4)


prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit.dummy <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=1, max.x=180)

summary(fit.dummy)



## safer procedure with more iterations (it may take some time)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## 1. binary case ##
```

```
prior <- list(pi0=mean(gestage), eta=rep(1, J)/J,
             a.pi0=27, b.pi0=360, J=J)

fit_binary<- comire.gibbs(premature, dde, family="binary",
                          mcmc=mcmc, prior=prior, seed=5, max.x=180)


## 2. continuous case ##

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit1 <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=5, max.x=180)


## 2.2 One confunder ##

mage_std <- scale(mage, center = TRUE, scale = TRUE)

prior <- list(mu.theta=mean(gestage), k.theta=10, mu.gamma=0, k.gamma=10,
              eta=rep(1, J)/J, alpha=1/H, a=2, b=2, H=H, J=J)

fit2 <- comire.gibbs(gestage, dde, mage_std, family="continuous",
              mcmc=mcmc, prior=prior, seed=5, max.x=180)


## 2.3 More confunders ##

Z <- cbind(mage, mbmi, sei)
Z <- scale(Z, center = TRUE, scale = TRUE)
Z <- as.matrix(cbind(Z, CPP$smoke))
colnames(Z) <- c("age", "BMI", "sei", "smoke")

mod <- lm(gestage ~ dde + Z)
prior <- list(mu.theta = mod$coefficients[1], k.theta = 10,
              mu.gamma = mod$coefficients[-c(1, 2)], sigma.gamma = diag(rep(10, 4)),
              eta = rep(1, J)/J, alpha = 1/H, a = 2, b = 2, H = H, J = J)

fit3 <- comire.gibbs(y = gestage, x = dde, z = Z, family = "continuous", mcmc = mcmc,
                     prior = prior, seed = 5)


}
```

`fit.pdf.mcmc` *Posterior mean density plot for dose intervals*

Description

Pointwise posterior mean (continuous blue lines), and credible bands (shaded blue areas) for f (y | x, z) calculated in `x.val` under the the model fitted in `fit`.

Usage

```
fit.pdf.mcmc(y, x, fit, mcmc, J=10, H = 10, alpha = 0.05,
max.x = max(x), x.val, y.grid = NULL, xlim = c(0, max(x)),
ylim = c(0, 1), xlab = NULL)
```

Arguments

`y`	optional numeric vector for the response used in `comire.gibbs`. If `y` is missing, `y.grid` must be provided.
`x`	numeric vector for the covariate relative to the dose of exposure used in `comire.gibbs`.
`fit`	the output of `comire.gibbs` opportunely trasformed in `classCoMiRe` class.
`mcmc`	a list giving the MCMC parameters.
`J`	parameter controlling the number of elements of the I-spline basis
`H`	total number of components in the mixture at x_0.
`alpha`	level of the credible bands.
`max.x`	maximum value allowed for x.
`x.val`	central points of each dose interval to be used in the posterior estimation of the probability density function.
`y.grid`	optional numerical vector giving the actual values of the grid for `y` for plotting the posterior mean density. If `y.grid` is not provided, standard grids are automatically used.
`xlim`, `ylim`	numeric vectors of length 2, giving the x and y coordinates ranges for the plot.
`xlab`	the title of the x axis.

Author(s)

Antonio Canale, Arianna Falcioni

Examples

```
{
data(CPP)
attach(CPP)

n <- NROW(CPP)
J <- H <- 10

premature <- as.numeric(gestage<=37)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)
```

```
## too few iterations to be meaningful. see below for safer and more comprehensive results

mcmc <- list(nrep=10, nb=2, thin=1, ndisplay=4)

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit.dummy <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=1, max.x=180)

fit.pdf.mcmc(y = gestage, x = dde, fit = fit.dummy, mcmc = mcmc, J = 10, H = 10,
                         alpha = 0.05, max.x = max(dde), x.val = 125,
                         xlim = c(25,48), ylim = c(0,0.25),
                         xlab = "Gest. age. for DDE = 125")


## safer procedure with more iterations (it may take some time)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## Fit the model for continuous y

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit1 <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=5, max.x=180)

fit.pdf.mcmc(y = gestage, x = dde, fit = fit1, mcmc = mcmc, J = 10, H = 10,
                         alpha = 0.05, max.x = max(dde), x.val = 125,
                         xlim = c(25,48), ylim = c(0,0.25),
                         xlab = "Gest. age. for DDE = 125")


}
```

`plot.classCoMiRe` *CoMiRe plot*

Description

An S3 plot method for an object of `classCoMiRe` class.

Usage

```
## S3 method for class 'classCoMiRe'
plot(
  x,
  y,
  xobs,
```

```
    mcmc,
    J = 10,
    H = 10,
    a = NULL,
    max.x = max(xobs),
    bandwidth = 20,
    x.grid = NULL,
    xlim = c(0, max(xobs)),
    ylim = c(0, 1),
    xlab = "x",
    alpha = 0.05,
    risk = TRUE,
    bmd = TRUE,
    level,
    oneevery = 20,
    ...
)
```

Arguments

`x`	the output of `comire.gibbs`, an object of the `classCoMiRe` class.
`y`	numeric vector for the response used in `comire.gibbs`.
`xobs`	numeric vector for the covariate relative to the dose of exposure used in `comire.gibbs`.
`mcmc`	a list giving the MCMC parameters.
`J`	parameter controlling the number of elements of the I-spline basis
`H`	total number of components in the mixture at x_0.
`a`	optional threshold of clinical interest for the response variable.
`max.x`	maximum value allowed for x.
`bandwidth`	the kernel bandwidth smoothing parameter for the `post.pred.check` plot.
`x.grid`	optional numerical vector giving the actual values of the grid for x for plotting the additional risk function. If $x.gird$ is not provided, standard grids are automatically used.
`xlim`, `ylim`	numeric vectors of length 2, giving the x and y coordinates ranges for the plot.
`xlab`	the title of the x axis.
`alpha`	level of the credible bands, default 0.05
`risk`	if `TRUE` the additional risk plot via `riskplot` is computed.
`bmd`	if `TRUE` the benchmark dose plot via `bmd.plot` is computed.
`level`	if `bmd=TRUE`, dose levels of interest for BMD plot.
`oneevery`	integer number representing how many MCMC draws to plot in the posterior predictive check. It draws one sample every `oneevery`.
`...`	additional arguments to be passed.

Details

The output is a list of `ggplot2` plots containing the result of the `betaplot` function and, if the threshold `a` is provided, of `post.pred.check`, `riskplot`, `bmd.plot`.

Value

If `a=NULL` returns only `betaplot` otherwise, if `risk=FALSE` and `bmd=FALSE` returns a list containing `betaplot` (which is automatically plotted) and `post.pred.check` plot. Finally, if `a` is provided, `risk=TRUE` and `bmd=TRUE` returns a list with `betaplot`, `post.pred.check`, `riskplot` and `bmd.plot`.

Author(s)

Antonio Canale, Arianna Falcioni

`post.pred.check` *Posterior predictive check plot*

Description

A plot for an object of `classCoMiRe` class. The plot is a goodness-of-fit assessment of CoMiRe model. Since Version 0.8 if `z` is provided into the `fit` object, an error message is returned. If `family = 'continuous'`, a smoothed empirical estimate of F(a|x) = pr(y < a | x) is computed from the observed data (black line) and from some of the data sets simulated from the posterior predictive distribution in the `fit` object (grey lines). If `family = 'binary'`, a smoothed empirical estimate of the proportion of events (black line) and of the smoothed empirical proportion of data simulated from the posterior predictive distribution in the `fit` object (grey lines). In the x axis are reported the observed exposures.

Usage

```
post.pred.check(y, x, fit, mcmc, J=10, H=10, a, max.x=max(x),
xlim=c(0, max(x)), bandwidth = 20, oneevery = 20)
```

Arguments

`y`	numeric vector for the response used in `comire.gibbs`
`x`	numeric vector for the covariate relative to the dose of exposure used in `comire.gibbs`
`fit`	the output of `comire.gibbs` opportunely trasformed in `classCoMiRe` class
`mcmc`	a list giving the MCMC parameters
`J`	parameter controlling the number of elements of the I-spline basis
`H`	total number of components in the mixture at x_0
`a`	threshold of clinical interest to compute the F(a|x,z)
`max.x`	maximum value allowed for x
`xlim`	numeric vectors of length 2, giving the x coordinates ranges for the plot

`bandwidth`	the kernel bandwidth smoothing parameter
`oneevery`	integer number representing how many MCMC draws to plot in the posterior predictive check. It draws one sample every `oneevery`.

Author(s)

Antonio Canale, Arianna Falcioni

Examples

```
{
data(CPP)
attach(CPP)

n <- NROW(CPP)
J <- H <- 10

premature <- as.numeric(gestage<=37)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## too few iterations to be meaningful. see below for safer and more comprehensive results

mcmc <- list(nrep=10, nb=2, thin=1, ndisplay=4)

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit.dummy <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=1, max.x=180)

post.pred.check(y = gestage, x = dde, fit = fit.dummy, mcmc = mcmc, J = 10, H = 10, a = 37,
                max.x = max(dde), xlim = c(0,150), oneevery = 4)


## safer procedure with more iterations (it may take some time)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## Fit the model for continuous y

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit1 <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=5, max.x=180)

post.pred.check(y = gestage, x = dde, fit = fit1, mcmc = mcmc, J = 10, H = 10, a = 37,
                max.x = max(dde), xlim = c(0,150))


}
```

`predict_new_z` *comire.gibbs for different fixed values of z*

Description

This function computes the predicted values of the density al low dose `f_0` and of the density at high dose `f_{inf}`, for fixed values of the confounders `z`.

Usage

```
predict_new_z(fit, y, z.val)
```

Arguments

`fit`	the output of `comire.gibbs` opportunely trasformed in `classCoMiRe` class
`y`	numeric vector for the response used in `comire.gibbs`
`z.val`	optional numeric vector containing a fixed value of interest for each of the confounding covariates to be used for the plots. Default value is `mean(z)` for numeric covariates or the mode for factorial covariates.

Value

An object of class `classCoMiRe`.

Author(s)

Antonio Canale, Arianna Falcioni

`print.classCoMiRe` *CoMiRe print*

Description

The `print.classCoMiRe` method prints the type of a `classCoMiRe` object.

Usage

```
## S3 method for class 'classCoMiRe'
print(x, ...)
```

Arguments

`x`	an object of class `classCoMiRe`;
`...`	additional arguments.

Author(s)

Antonio Canale, Arianna Falcioni

`riskplot` *Additional risk function plot*

Description

Posterior mean (continuous lines) and pointwise credible bands (shaded areas) for $Ra(x, a)$.

Usage

```
riskplot(risk.data, xlab = NULL, x = NULL, ylim=c(0,1), xlim=c(0, max(x)))
```

Arguments

`risk.data`	output of `add.risk` function.
`xlab`	the title of the x axis.
`x`	numeric vector for the covariate relative to the dose of exposure used in `comire.gibbs`.
`xlim`, `ylim`	numeric vectors of length 2, giving the x and y coordinates ranges for the plot.

Author(s)

Antonio Canale

Examples

```
{
data(CPP)
attach(CPP)

n <- NROW(CPP)
J <- H <- 10

premature <- as.numeric(gestage<=37)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## too few iterations to be meaningful. see below for safer and more comprehensive results

mcmc <- list(nrep=10, nb=2, thin=1, ndisplay=4)

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit.dummy <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=1, max.x=180)
```

```
risk.data <- add.risk(y = gestage, x = dde, fit = fit.dummy, mcmc = mcmc,
    a = 37, x.grid = seq(0, max(dde), length = 100))
riskplot(risk.data$summary.risk, xlab="DDE", x = dde, xlim = c(0,150))


## safer procedure with more iterations (it may take some time)

mcmc <- list(nrep=5000, nb=2000, thin=5, ndisplay=4)

## Fit the model for continuous y

prior <- list(mu.theta=mean(gestage), k.theta=10, eta=rep(1, J)/J,
              alpha=rep(1,H)/H, a=2, b=2, J=J, H=H)

fit <- comire.gibbs(gestage, dde, family="continuous",
                     mcmc=mcmc, prior=prior, seed=5, max.x=180)

risk.data <- add.risk(y = gestage, x = dde, fit = fit, mcmc = mcmc,
a = 37, x.grid = seq(0, max(dde), length = 100))
riskplot(risk.data$summary.risk, xlab="DDE",
x = dde, xlim = c(0,150))


}
```

`summary.classCoMiRe` *CoMiRe summary*

Description

The `summary.classCoMiRe` method provides summary information on `classCoMiRe` objects.

Usage

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

Arguments

`object`	an object of class `classCoMiRe`;
`...`	additional arguments

Author(s)

Antonio Canale Arianna Falcioni

Index