

Package ‘qbld’

August 25, 2020

Type Package

Title Quantile Regression for Binary Longitudinal Data

Version 1.0

Date 2020-08-17

Author Ayush Agarwal [aut, cre], Dootika Vats [ctb]

Maintainer Ayush Agarwal<ayush.agarwal50@gmail.com>

Description This project follows Rahman and Vossmeier (2019) as its motivating literature, and contributes to the three literatures by extending the various methodologies to a hierarchical Bayesian quantile regression model for binary longitudinal data (QBLD) and proposing a Markov chain Monte Carlo (MCMC) algorithm to estimate the model. The model handles both common (fixed) and individual-specific (random) parameters (commonly referred to as mixed effects in statistics). The algorithm implements a blocking procedure that is computationally efficient and the distributions involved allow for straightforward calculations of covariate effects.

License GPL-3

Imports Rcpp, stats, grDevices, graphics, mcmcse, stableGR, RcppDist, knitr, rmarkdown

LinkingTo Rcpp, RcppArmadillo, RcppDist

Depends R (>= 3.5)

Encoding UTF-8

VignetteBuilder knitr

RoxygenNote 7.1.1

NeedsCompilation yes

R topics documented:

qbld-package	2
airpollution	3
aldmix	4
gig	5
locust	6
model.qbld	7
plot.qbld	9
summary.qbld	10

Index	12
-------	----

Description

This project follows Rahman and Vossmeier (2019) as its motivating literature, and contributes to the three literatures by extending the various methodologies to a hierarchical Bayesian quantile regression model for binary longitudinal data (QBLD) and proposing a Markov chain Monte Carlo (MCMC) algorithm to estimate the model. The model handles both common (fixed) and individual-specific (random) parameters (commonly referred to as mixed effects in statistics). The algorithm implements a blocking procedure that is computationally efficient and the distributions involved allow for straightforward calculations of covariate effects.

Details

Package: qbld
 Type: Package
 Version: 1.0
 Date: 2020-08-17
 License: GPL (>= 3)

The package contains the following functions:

- `model.qbld` : Runs the QBLD sampler as in Rahman and Vossmeier(2019) and outputs a 'qbld' class object.
- `summary.qbld` : S3 method that summarizes the outputs of the model.qbld function.
- `plot.qbld` : S3 method that plots 'qbld' class object.
- `aldmix` : Cumulative density, Probability distribution function, Quantile function and Random generation for the Asymmetric Laplace distribution.
- `gig` : Probability distribution function, Quantile Random generation for the Generalised Inverse Gaussian.
- `airpollution`, `locust` : In-built datasets

Author(s)

Ayush Agarwal [aut, cre], Dootika Vats [ctb]

Maintainer: Ayush Agarwal<ayush.agarwal50@gmail.com>

References

Rahman, Mohammad and Vossmeier, Angela. (2018). Estimation and Applications of Quantile Regression for Binary Longitudinal Data. *Advances in Econometrics*. 40.

Vats, Dootika and Christina Knudson. "Revisiting the Gelman-Rubin Diagnostic." *arXiv: Computation* (2018): n. pag.

Keming Yu and Jin Zhang (2005) A Three-Parameter Asymmetric Laplace Distribution and Its Extension, *Communications in Statistics - Theory and Methods*.

Kobayashi, Genya. (2011). Gibbs Sampling Methods for Bayesian Quantile Regression. J Stat Comput Simul.

Devroye, L. Random variate generation for the generalized inverse Gaussian distribution. Stat Comput 24, 239–246 (2014).

Wolfgang Hörmann and Josef Leydold (2013). Generating generalized inverse Gaussian random variates, Statistics and Computing (to appear). *must have appeared by now*

J. S. Dagpunar (1989). An easily implemented generalised inverse Gaussian generator, Comm. Statist. B – Simulation Comput. 18, 703–710.

Examples

```
# Dataset
data(airpollution)

# output will be a qbld class object, summarize is TRUE
output <- model.qbld(fixed_formula = wheeze~smoking+I(age^2)-1, data = airpollution, id="id",
  random_formula = ~1, p=0.25,
  nsim=1000, method="block", burn=0,
  summarize=TRUE, verbose=FALSE)

# plots
plot(output)

# GIG sampler
rgig(n = 1, lambda = 0.5, a = 1, b = 2)

# ALD sampler
raldmix(n = 10, mu = 5, sigma = 10, p = 0.5)
```

Add summary as well.

airpollution

Air Pollution

Description

This example is a subset of data from Six Cities study, a longitudinal study of the health effects of air pollution (Ware, J. H. et al., 1984).

Usage

```
data(airpollution)
```

Format

A data frame with 128 observations on the following 5 variables.

id identifies the number of the individual profile. This vector contains observations of 32 individual profiles.

wheeze a numeric vector that identifies the wheezing status (1="yes", 0="no") of a child at each occasion.

age a numeric vector corresponding to the age in years.

smoking a factor that identifies if the mother smokes (1="smoke", 0="no smoke").

counts a numeric vector corresponding to the replications of each individual profile.

Details

The data set presented by Fitzmaurice and Laird (1993) contains complete records on 32 children from Steubenville, Ohio, each woman was examined annually at ages 7 through 10. The repeated binary response is the wheezing status (1="yes", 0="no") of a child at each occasion. Although mother's smoking status could vary with time, it was determined in the first interview and was treated as a time-independent covariate. Maternal smoking was categorized as 1 if the mother smoked regularly and 0 otherwise.

Source

Fitzmaurice, G. M. and Laird, N. M. (1993). A Likelihood-Based Method for analyzing Longitudinal Binary Response. *Biometrika*, 80, 141-51.

References

Ware, J. H., Dockery, D. W., Spiro, A. III, Speizer, F. E. and Ferris, B. G., Jr. (1984). Passive smoking, gas cooking and respiratory health in children living in six cities. *Am. Rev. Respir. dis.*, 129, 366-74.

aldmix

Asymmetric Laplace distribution

Description

Cumulative density, Probability distribution function, Quantile function and Random generation for the Asymmetric Laplace distribution with quantile p , location parameter μ and scale parameter σ .

Usage

```
raldmix(n, mu, sigma, p)
```

```
daldmix(x, mu = 0, sigma = 1, p = 0.5)
```

```
paldmix(q, mu = 0, sigma = 1, p = 0.5, lower.tail = TRUE)
```

```
qaldmix(prob, mu = 0, sigma = 1, p = 0.5, lower.tail = TRUE)
```

Arguments

n : number of observations
 μ : location parameter
 σ : scale parameter
 p , prob : probability at which to calculate quantile
 x , q : vector of quantiles
 lower.tail : logical; decides b/w $P(X \leq p)$ or $P(X > p)$ for p/q

All math should be in math mode

Details

The Asymmetric Laplace distribution (ALD), which has the following pdf:

$$f(x; \mu, \sigma, p) = \frac{p(1-p)}{\sigma} \exp\left\{-\frac{(x-\mu)}{\sigma}(p - I(x \leq \mu))\right\}$$

If not specified, p=0.5, mu = 0, sigma = 1.

math mode

Value

- `raldmix` returns a vector of random numbers from ALD(mu,sigma,p).
- `daldmix` returns density of AL(mu,sigma,p) at point x.
- `paldmix` returns CDF prob of AL(mu,sigma,p) at quantile q.
- `qaldmix` returns inverse CDF quantile of AL(mu,sigma,p) at prob.

*include
ALD....*

References

Keming Yu & Jin Zhang (2005) A Three-Parameter Asymmetric Laplace Distribution and Its Extension, Communications in Statistics - Theory and Methods, 34:9-10, 1867-1879, DOI: 10.1080/03610920500199018

Kobayashi, Genya. (2011). Gibbs Sampling Methods for Bayesian Quantile Regression. J Stat Comput Simul. 81. 1565. 10.1080/00949655.2010.496117.

See Also

[rgig](#) for random sampling from GIG distribution

Examples

```
raldmix(n = 10, mu = 5, sigma = 10, p = 0.5)
daldmix(c(4,5),mu = 0,sigma = 1,p = 0.5)
paldmix(c(1,4),mu = 0,sigma = 1,p = 0.5,lower.tail=TRUE)
qaldmix(0.5,mu = 0,sigma = 1,p = 0.5,lower.tail=TRUE)
```

gig

Generalised Inverse Gaussian

Description

Probability distribution function, Quantile Random generation for the Generalised Inverse Gaussian with three parameters a(chi),b(psi),p.

add space after,

Usage

`dgig(x, a, b, p, log_density)`

`rgig(n, lambda, a, b)`

Arguments

x	: Argument of pdf
a	: chi parameter. Must be nonnegative for positive lambda and positive else.
b	: psi parameter. Must be nonnegative for negative lambda and positive else.
log_density	: logical; returns log density if TRUE
n	: number of observations
lambda, p	: lambda parameter

Details

The Generalised Inverse Gaussian distribution (GIG), which has the following pdf

$$f(x) = x^{\lambda-1} \exp\left\{-\frac{\omega}{2}\left(x + \frac{1}{x}\right)\right\}$$

Handwritten note: exp { ... }

Value

- rgig returns a vector of random numbers from GIG(a,b,p).
- dgig returns density of a GIG(a,b,p) at point x.

References

- Devroye, L. Random variate generation for the generalized inverse Gaussian distribution. Stat Comput 24, 239–246 (2014).
- Wolfgang Hörmann and Josef Leydold (2013). Generating generalized inverse Gaussian random variates, Statistics and Computing (to appear), DOI: 10.1007/s11222-013-9387-3
- J. S. Dagpunar (1989). An easily implemented generalised inverse Gaussian generator, Comm. Statist. B – Simulation Comput. 18, 703–710.

See Also

[raldmix](#) for random sampling from Asymmetric Laplace distribution

Examples

```
rgig(n = 1, lambda = 0.5, a = 1, b = 2)
dgig(x = 1, a = 1, b = 2, p = 0.5, log_density = FALSE)
```

locust

Locust

Dataset

Description

This data set was presented by MacDonald and Raubenheimer (1995) and analyze the effect of hunger on locomotory behaviour of 24 locust (*Locusta migratoria*) observed at 161 time points. The subjects were divided in two treatment groups ("fed" and "not fed"), and within each of the two groups, the subjects were alternatively "male" and "female". For the purpose of this analysis the categories of the response variable were "moving" and "not moving". During the observation period, the behavior of each of the subjects was registered every thirty seconds.

Usage

```
data(locust)
```

Format

A data frame with 3864 observations on the following 7 variables.

`id` a numeric vector that identifies the number of the individual profile.

`move` a numeric vector representing the response variable.

`sex` a factor with levels 1 for "male" and 0 for "female".

`time` a numeric vector that identifies the number of the time points observed. The time vector considered was obtained dividing (1:161) by 120 (number of observed periods in 1 hour).

`feed` a factor with levels 0 "no" and 1 "yes".

Details

The response variable, `move` is the binary type coded as 1 for "moving" and 0 for "not moving". The sex covariate was coded as 1 for "male" and 0 for "female". The feed covariate indicating the treatment group, was coded as 1 for "fed" and 0 for "not fed". Azzalini and Chiogna (1997) also have analyzed this data set using their S-plus package `rm.tools`.

Source

MacDonald, I. and Raubenheimer, D. (1995). Hidden Markov models and animal behaviour. *Biometrical Journal*, 37, 701-712

References

Azzalini, A. and Chiogna, M. (1997). S-Plus Tools for the Analysis of Repeated Measures Data. *Computational Statistics*, 12, 53-66

model.qbld

QBLD Sampler

Description

Runs the QBLD sampler as in Rahman and Vossmeier(2019) and outputs a 'qbld' class object which consists of a list of markov chains for Beta(the fixed effects estimate), Alpha(the random effects estimate), and Varphi2 (as per the model), of which Beta and Varphi2 are of interest.

Usage

```
model.qbld(fixed_formula, data, id = "id", random_formula = ~1, p = 0.25,
  nsim, b0 = 0, B0 = 1, c1 = 9, d1 = 10, method=c("block","unblock"),
  burn = 0, summarize = FALSE, verbose = FALSE)
```

Not a list of Markov chains

space

Arguments

- Handwritten: $\$X_i\$$*
- fixed_formula** : a description of the model to be fitted of the form response~fixed effects predictors i.e X_i in the model. *See details for more information*
- data** : data frame, NAs not allowed and should throw errors, factor variables are auto-converted, find airpollution.rda and locust.rda built into the package.
- id** : variable name in the dataset that specifies individual profile. By default, *'id="id"'* and data is expected to contain an id variable. This is omitted while modelling.
- random_formula** : a description of the model to be fitted of the form response~random effects predictors i.e S_i in the model. This defaults to S_i being only an intercept.
- p** : quantile for the AL distribution on the error term, $p=0.25$ by default. For very low (≤ 0.025) or very high (≥ 0.970) values of p, sampler forces to unblock version to avoid errors. *Handwritten: $\$ \leq \$$*
- nsim** : number of simulations to run the sampler.
- b0, B0** : Prior model parameters for Beta. These are defaulted to 0 vector, and Identity matrix.
- c1, d1** : Prior model parameters for Varphi2. These are defaulted to 9,10 (arbitrary) respectively.
- method** : Choose between the "Block" vs "Unblock" sampler, Block is slower but produces lower correlation.
- burn** : Burn in percentage, number between (0,1). Burn-in values are discarded and not used for summary calculations.
- summarize** : Outputs a summary table (same as summary(output)), in addition also prints Model fit AIC/BIC/Log-likelihood values. False by default.
- verbose** : False by default. Spits out progress reports while the sampler is running.

Details

For a detailed information on the sampler, please check the vignette. Data are contained in a data.frame. Each element of the data argument must be identifiable by a name. The simplest situation occurs when all subjects are observed at the same time points. The id variable represent the individual profiles of each subject, it is expected a variable in the data.frame that identifies the correspondence of each component of the response variable to the subject that it belongs, by default is named id variable. Hence NA values are not valid. For very low (≤ 0.025) or very high (≥ 0.970) values of p, sampler forces to unblock version to avoid errors. Block version in this case may lead to machine tolerance issues. *Handwritten: $\$ - \$$*

'qbld' object contains markov chains and sampler run information as attributes, and is compatible with S3 methods like summary, plot. make.qbld function can be used to convert a similar type-object to 'qbld' class.

Value

Returns 'qbld' class object. 'qbld' class contains the following :

- Beta: Matrix of MCMC samples of fixed-effects parameters.
 - Alpha: *Handwritten: 3D* Matrix of MCMC samples of random-effects parameters.
 - Varphi2: Matrix of MCMC samples for varphi2.
 - nsim: Attribute; No. of simulations of chain run.
 - burn: Attribute; Whether or not burn-in used.
 - which: Attribute; "block" or "unblock" sampler used.
- Handwritten: what does this mean?*

References

Rahman, Mohammad & Vossmeier, Angela. (2018). Estimation and Applications of Quantile Regression for Binary Longitudinal Data. *Advances in Econometrics*. 40.

See Also

A qbld object may be summarized by the summary function and visualized with the plot function.

[summary.qbld](#), [plot.qbld](#)

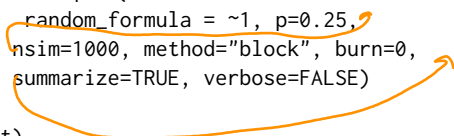
Datasets : [airpollution](#), [locust](#)

Examples

```
data(airpollution)

output <- model.qbld(fixed_formula = wheeze~smoking+I(age^2)+age+1, data = airpollution, id="id",
  random_formula = ~1, p=0.25,
  nsim=1000, method="block", burn=0,
  summarize=TRUE, verbose=FALSE)

plot(output)
```



plot.qbld

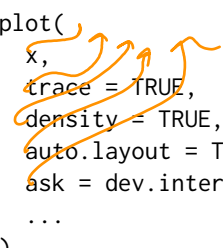
Plot QBLD

Description

Plots 'qbld' class object.

Usage

```
## S3 method for class 'qbld'
plot(
  x,
  trace = TRUE,
  density = TRUE,
  auto.layout = TRUE,
  ask = dev.interactive(),
  ...
)
```



Arguments

x	: 'qbld' class object to plot.
trace	: Whether or not to plot trace plots for covariates, TRUE by default
density	: Whether or not to plot density for covariates, TRUE by default.
auto.layout	: Auto set layout or not, TRUE as default. Plots according to the local settings if false.

ask : For Interactive plots
 ... : Other plot arguments

Value

Plots as specified.

See Also

[summary.qbld](#), [model.qbld](#)

summary.qbld	<i>QBLD Summary Class</i>
--------------	---------------------------

Description

Outputs a 'summary.qbld' class object, and prints as described.

Usage

```
## S3 method for class 'qbld'
summary(
  object,
  quantiles = c(0.025, 0.25, 0.5, 0.75, 0.975),
  epsilon = 0.05,
  ...
)

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

Arguments

object : 'qbld' class object
 quantiles : Vector of quantiles for summary of the covariates, defaulted to c(0.025, 0.25, 0.5, 0.75, 0.975) *describe this*
 epsilon : epsilon value for calculating target.psr and significance stars, 0.05 by default.
 ... : Other summary arguments
 x : (for print.summary.qbld) 'qbld.summary' class object

Details

'qbld.summary' class summarizes the outputs of the model.qbld function. Markov Std Error (MCSE), Effective sample size (ESS) are calculated using mcmcse package. Gelman-Rubin diagnostic, (R hat) is calculated using ESS values for each covariate. The diagnostic is then compared to a target value to check whether enough samples are generated, significance stars are added to indicate the same. MultiESS and multi-Gelman repeats the same process by treating all the parameter chains as one multi-variate chain.

↑ space 'mcmcse' Rubin space
→ This detail is not needed. Just say significance stars are indicated using Vats and Koudson and Vats et. al.

Value

summary.qbld produces following sets of summary statistics for each variable:

- `statistics`: Contains the mean, sd, markov std error, ess and Gelman-Ruben diagnostic
- `quantiles`: Contains quantile estimates for each variable
- `nsim`: No. of simulations run
- `burn`: Burn-in used or not
- `which`: Block, or Unblock version of sampler
- `p`: quantile for the AL distribution on the error term
- `multiess`: multiess value for the sample
- `multigelman`: multivariate version of Gelman-Ruben

References

Vats, Dootika and Christina Knudson. “Revisiting the Gelman-Rubin Diagnostic.” arXiv: ~~Compu-~~
~~tation (2018): n. pag.~~

James M. Flegal, John Hughes, Dootika Vats, and Ning Dai. (2020). `mcmcse`: Monte Carlo Standard Errors for MCMC. R package version 1.4-1. Riverside, CA, Denver, CO, Coventry, UK, and Minneapolis, MN.

Christina Knudson and Dootika Vats (2020). `stableGR`: A Stable Gelman-Rubin Diagnostic for Markov Chain Monte Carlo. R package version 1.0.

See Also

[plot.qbld](#), [model.qbld](#)

Additional functions : [mcse.mat](#), [ess](#), [multiESS](#), [stable.GR](#), [target.psrfr](#)

Index

- * **package**
 - qbld-package, [2](#)
- * **qbld**
 - qbld-package, [2](#)
- airpollution, [2](#), [3](#), [9](#)
- aldmix, [2](#), [4](#)
- daldmix (aldmix), [4](#)
- dgig (gig), [5](#)
- ess, [11](#)
- gig, [2](#), [5](#)
- locust, [2](#), [6](#), [9](#)
- mcse.mat, [11](#)
- model.qbld, [2](#), [7](#), [10](#), [11](#)
- multiESS, [11](#)
- paldmix (aldmix), [4](#)
- plot.qbld, [2](#), [9](#), [9](#), [11](#)
- print.summary.qbld (summary.qbld), [10](#)
- qaldmix (aldmix), [4](#)
- qbld (qbld-package), [2](#)
- qbld-package, [2](#)
- raldmix, [6](#)
- raldmix (aldmix), [4](#)
- rgig, [5](#)
- rgig (gig), [5](#)
- stable.GR, [11](#)
- summary.qbld, [2](#), [9](#), [10](#), [10](#)
- target.psr, [11](#)