Package ‘betaBayes’

June 8, 2021

Type Package
Title Bayesian Beta Regression
Version 1.0
Date 2021-06-07
Description Provides a class of Bayesian beta regression models for the analysis of continuous data with support restricted to an unknown finite support. The response variable is modeled using a four-parameter beta distribution with the mean or mode parameter depending linearly on covariates through a link function. When the response support is known to be (0,1), the above class of models reduce to traditional (0,1) supported beta regression models. Model choice is carried out via the logarithm of the pseudo marginal likelihood (LPML), the deviance information criterion (DIC), and the Watanabe-Akaike information criterion (WAIC). See Zhou and Huang (2021, “Bayesian beta regression for bounded responses with unknown supports”) <https://sites.google.com/view/haimingzhou/research>.

License GPL (>= 2)
Depends R (>= 3.5.0)
Imports Rcpp (>= 0.11.1), splines, methods, coda, betareg
LinkingTo Rcpp, RcppArmadillo (>= 0.4.300.0)
NeedsCompilation yes
Author Haiming Zhou [aut, cre, cph], Xianzheng Huang [aut]
Maintainer Haiming Zhou <zhouh@niu.edu>
Repository CRAN
Date/Publication 2021-06-08 07:40:05 UTC

R topics documented:

<table>
<thead>
<tr>
<th>beta4reg</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>covid</td>
<td>5</td>
</tr>
<tr>
<td>cox.snell.beta4reg</td>
<td>6</td>
</tr>
<tr>
<td>predict.beta4reg</td>
<td>7</td>
</tr>
</tbody>
</table>

Index

8
Description

This function fits Bayesian beta regression models. The response distribution can be either the beta with the support on (0,1) or the four-parameter beta with an unknown final support. The logarithm of the pseudo marginal likelihood (LPML), the deviance information criterion (DIC), and the Watanabe-Akaike information criterion (WAIC) are provided for model comparison.

Usage

```r
beta4reg(formula, data, na.action, link="logit", model = "mode",
        mcmc=list(nburn=3000, nsave=2000, nskip=0, ndisplay=500),
        prior=NULL, start=NULL, Xpred=NULL)
```

Arguments

- `formula`: a formula expression of the form `y ~ x`.
- `data`: a data frame in which to interpret the variables named in the `formula` argument.
- `na.action`: a missing-data filter function, applied to the `model.frame`.
- `link`: a character string for the link function. Choices include "logit", "probit", "loglog" and "cloglog".
- `model`: a character string for the regression type. The options include "mean" for a mean regression, "mode" for a mode regression.
- `mcmc`: a list giving the MCMC parameters. The list must include the following elements: `nburn` an integer giving the number of burn-in scans, `nskip` an integer giving the thinning interval, `nsave` an integer giving the total number of scans to be saved, `ndisplay` an integer giving the number of saved scans to be displayed on screen (the function reports on the screen when every `ndisplay` iterations have been carried out).
- `prior`: a list giving the prior information. The function itself provides all default priors. The following components can be specified here: `ma0` and `mb0` for the prior of marginal population mode or mean, `phia0` and `phib0` for the precision parameter, `beta0` and `S0` for the coefficients beta, `th1a0` and `th1b0` for the lower bound of the support, `th2a0` and `th2b0` for the upper bound of the support.
- `start`: a list giving the starting values of the parameters. The function itself provides all default choices. The following components can be specified here: `beta`, `theta`, `phi`.
- `Xpred`: A new design matrix at which estimates of the response model or mean are required. The default is the design matrix returned by the argument `formula`.
**Value**

This class of objects is returned by the `beta4reg` function to represent a fitted Bayesian beta regression model. Objects of this class have methods for the functions `print` and `summary`.

The `beta4reg` object is a list containing the following components:

- **modelname**: the name of the fitted model
- **terms**: the `terms` object used
- **link**: the link function used
- **model**: the model fitted: mean or mode
- **coefficients**: a named vector of coefficients. The last two elements are the estimates of theta1 and theta2 involved in the support of the four-parameter beta distribution.
- **call**: the matched call
- **prior**: the list of hyperparameters used in all priors.
- **start**: the list of starting values used for all parameters.
- **mcmc**: the list of MCMC parameters used
- **n**: the number of row observations used in fitting the model
- **p**: the number of columns in the model matrix
- **y**: the response observations
- **X**: the n by (p+1) original design matrix
- **beta**: the (p+1) by nsave matrix of posterior samples for the coefficients in the `linear.predictors`
- **theta**: the 2 by nsave matrix of posterior samples for theta1 and theta2 involved in the support.
- **phi**: the vector of posterior samples for the precision parameter.
- **cpo**: the length n vector of the stabilized estimate of CPO; used for calculating LPML
- **pD**: the effective number of parameters involved in DIC
- **DIC**: the deviance information criterion (DIC)
- **pW**: the effective number of parameters involved in WAIC
- **WAIC**: the Watanabe-Akaike information criterion (WAIC)
- **ratetheta**: the acceptance rate in the posterior sampling of theta vector involved in the support
- **ratebeta**: the acceptance rate in the posterior sampling of beta coefficient vector
- **ratephi**: the acceptance rate in the posterior sampling of precision parameter

The use of the `summary` function to the object will return new object with the following additional components:

- **coeff**: A table that presents the posterior summaries for the regression coefficients
- **bounds**: A table that presents the posterior summaries for the support boundaries theta1 and theta2
- **phivar**: A table that presents the posterior summaries for the precision phi.
Author(s)
Haiming Zhou <<zhouh@niu.edu>> and Xianzheng Huang <<huang@stat.sc.edu>>

References

See Also
cox.snell.beta4reg

Examples
library(betaBayes)
library(coda)
library(betareg)

## Data from Ferrari and Cribari-Neto (2004)
data("GasolineYield", package = "betareg")
data("FoodExpenditure", package = "betareg")

## four-parameter beta mean regression
mcmc=list(nburn=2000, nsave=1000, nskip=4, ndisplay=1000);
# Note larger nburn, nsave and nskip should be used in practice.
prior = list(th1a0 = 0, th2b0 = 1)
# here the natural bound (0,1) is used to specify the prior
# GasolineYield
set.seed(100)
gy_res1 <- beta4reg(yield ~ batch + temp, data = GasolineYield,
                  link = "logit", model = "mean",
                  mcmc = mcmc, prior = prior)
(gy_sfit1 <- summary(gy_res1))
cox.snell.beta4reg(gy_res1) # Cox-Snell plot
# FoodExpenditure
set.seed(100)
fe_res1 <- beta4reg(I(food/income) ~ income + persons, data = FoodExpenditure,
                  link = "logit", model = "mean",
                  mcmc = mcmc, prior = prior)
(fe_sfit1 <- summary(fe_res1))
cox.snell.beta4reg(fe_res1) # Cox-Snell plot

## two-parameter beta mean regression with support (0,1)
mcmc=list(nburn=2000, nsave=1000, nskip=4, ndisplay=1000);
# Note larger nburn, nsave and nskip should be used in practice.
prior = list(th1a0 = 0, th1b0 = 0, th2a0 = 1, th2b0 = 1)
# this setting forces the support to be (0,1)
# GasolineYield
set.seed(100)
gy_res2 <- beta4reg(yield ~ batch + temp, data = GasolineYield,
                  link = "logit", model = "mean",
                  mcmc = mcmc, prior = prior)
(gy_sfit2 <- summary(gy_res2))
cox.snell.beta4reg(gy_res2) # Cox-Snell plot
# FoodExpenditure
set.seed(100)
fe_res2 <- beta4reg(I(food/income) ~ income + persons, data = FoodExpenditure,
               link = "logit", model = "mean",
               mcmc = mcmc, prior = prior)
(fesfit2 <- summary(fesres2))
cox.snell.beta4reg(fesres2) # Cox-Snell plot

---

**covid**  

**COVID-19 County Level Data**

**Description**

A county level COVID-19 dataset in US. It is of interest to examine the association between several county-level characteristics and the cumulative numbers of confirmed cases and deaths. County-level characteristics are based on the 2018 ACS 5-year estimates.

**Usage**

data(covid)

**Format**

- **FIPS**: FIPS county code
- **PopE**: total population
- **MaleP**: percentage of people who are male
- **WhiteP**: percentage of people who are white
- **BlackP**: percentage of people who are black or African American
- **Age65plusP**: percentage of people who are 65 years and over
- **PovertyP**: percentage of people whose income in the past 12 months is below poverty
- **RUCC_2013**: 2013 Rural Urban Continuum Code, with a higher value indicating a more rural county
- **State**: two-letter state abbreviation code
- **deaths**: cumulative number of deaths as of October 13, 2020
- **cases**: cumulative number of confirmed cases as of October 13, 2020

**Examples**

```r
data(covid)
head(covid)
```
cox.snell.beta4reg  Cox-Snell Diagnostic Plot

Description

This function provides the Cox-Snell diagnostic plot for fitting for Bayesian beta regression models.

Usage

```r
cox.snell.beta4reg(x, ncurves = 10, CI = 0.95, PLOT = TRUE)
```

Arguments

- `x`: an object obtained from the function `beta4reg`.
- `ncurves`: the number of posterior draws.
- `CI`: the level of confidence for point-wise credible intervals.
- `PLOT`: a logical value indicating whether the Cox-Snell residuals will be plotted.

Value

The function returns the plot (if `PLOT = TRUE`) and a list with the following components:

- `tgrid`: the x-axis values with length, say `ngrid`
- `Hhat`: the `ngrid` by 1 averaged cumulative hazard values across the `nsave` posterior samples
- `Hhatlow`: the `ngrid` by 1 lower bound cumulative hazard values
- `Hhatup`: the `ngrid` by 1 upper bound cumulative hazard values
- `H`: the `ngrid` by `nsave` cumulative hazard values

Author(s)

Haiming Zhou <<zhouh@niu.edu>> and Xianzheng Huang <<huang@stat.sc.edu>>

See Also

`beta4reg`
predict.beta4reg

Predict method for beta4 model fits

Description

Posterior predicted response values based on beta4 model object

Usage

## S3 method for class 'beta4reg'
predict(object, newx, ...)

Arguments

- **object**: an object obtained from the function `beta4reg`.
- **newx**: an m by p matrix at which predictions are required. If not specified, the original design matrix will be used.
- **...**: further arguments passed to or from other methods.

Value

The function returns an m by nsave matrix of posterior samples for response predictions at newx.

Author(s)

Haiming Zhou `<zhouh@niu.edu>` and Xianzheng Huang `<huang@stat.sc.edu>`

See Also

`beta4reg`
Index

beta4reg, 2, 6, 7

covid, 5
cox.snell.beta4reg, 4, 6

predict.beta4reg, 7
print.beta4reg (beta4reg), 2
print.summary.beta4reg (beta4reg), 2

summary.beta4reg (beta4reg), 2

terms, 3