

Package ‘HBglm’

July 14, 2015

Type Package

Title Hierarchical Bayesian Regression for GLMs

Version 0.1

Date 2015-7-13

Imports Formula, bayesm, sns, MfUSampler, stats

Author Asad Hasan, Alireza S. Mahani

Maintainer Asad Hasan <asad.hasan@sentrana.com>

Depends R (>= 2.10)

Description Convenient and efficient functions for performing 2-level hierarchical Bayesian regression analysis for multi-group data. The lowest level may belong to the generalized linear model (GLM) family while the prior level, which effects pooling, allows for linear regression on lower level covariates. Constraints on all or part of the parameter set maybe specified with ease. A rich set of methods is included to visualize and analyze results.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2015-07-14 15:59:53

R topics documented:

HBglm	2
linear_list	4

Index	6
--------------	----------

Description

Convenient and efficient functions for performing 2-level HB regressions analysis for multi-group data. The lowest level may belong to the GLM family while the prior level, which effects pooling, allows for linear regression on lower level covariates. Constraints on all or part of the parameter set maybe specified with ease. A rich set of methods is included to visualize and analyze results.

Usage

```
hbglm(formula, data, formula.fixed = NULL,
      family = "gaussian",
      sampler.control = hbglm.sampler.control(),
      model.control = hbglm.model.control(),
      ncores = 1, print.level = 0, ...)
hbglm.sampler.control(num.samples = -1, samp.factor = 50,
                      sampler = c("slice"))
hbglm.model.control(constraints = NULL,
                    initializer = c("user", "regression"), user.init.val = NULL)
## S3 method for class 'hbglm'
print(x, digits = max(3, getOption("digits") - 2),
      width = getOption("width"), nburnin = 0, ...)
## S3 method for class 'hbglm'
predict(object, newdata = NULL, grpID.col = NULL,
        type = c("link", "response", "ppp.mean", "ppp.dist"),
        nburnin = 0, times = 0, print.level = 0, ...)
## S3 method for class 'hbglm'
coef(object, nburnin = 0, ...)
## S3 method for class 'hbglm'
summary(object, nburnin = 0, ...)
## S3 method for class 'hbglm'
print.summary(x, digits = max(3, getOption("digits") - 2),
             width = getOption("width"), ...)
```

Arguments

formula	formula object or string specifying the model to be estimated (see Note).
data	A data.frame object with data organized in the 'long' format or as a list of lists (see Note).
formula.fixed	formula specifying the fixed effects included in the model.
family	Name of a GLM family member. Currently allowed "gaussian", "binomial", "poisson" with standard link functions.
sampler.control	List to control MCMC sampling.

<code>model.control</code>	List to specify initial values, constraints.
<code>ncores</code>	Number of cores to run on. Currently fixed at 1.
<code>print.level</code>	Positive integer to print stuff while running.
<code>num.samples</code>	Number of MCMC samples to draw. Default '-1' will cause internal default values to kick-in.
<code>samp.factor</code>	Used in setting internal default for number of MCMC samples to draw. Must be a positive integer
<code>sampler</code>	Name of MCMC sampler to use. Currently fixed at the slice sampler.
<code>constraints</code>	User specified constraints.
<code>initializer</code>	How to obtain initial values.
<code>user.init.val</code>	User specified initial values.
<code>x, object</code>	A <code>hbglm</code> class object.
<code>width</code>	Printing control options.
<code>nburnin</code>	Number of burn-in samples. These samples, starting from the initial, are discarded from print, summary and predict.
<code>digits</code>	Printing control options.
<code>newdata</code>	Same format as data argument. Used in prediction.
<code>grpID.col</code>	Name of column in data which has group information for each observation.
<code>type</code>	Used in predict. 'link' and 'response' are non-Bayesian predictions and return the avg response on the linear predictor and the response scale, respectively. They are computed by using model coefficients obtained by averaging MCMC samples. 'ppp.mean' return the distribution of means of the response posterior predictive probability (PPP) distribution. 'ppp.dist' returns samples drawn from the full response PPP distribution.
<code>times</code>	Used in predict when type = 'ppp.dist' to specify number of samples desired from the response PPP distribution. When 0, a default internal value is used.
<code>...</code>	Currently unused.

Value

An object of class `hbglm`, with elements:

<code>formula</code>	the model formula object
<code>formula.fixed</code>	the fixed effects formula object
<code>nsamples</code>	the number of samples drawn
<code>model.control</code>	list of model control args
<code>sampler.control</code>	list of sampler control args
<code>grpID.col</code>	Data column with grp ID information
<code>call</code>	the <code>hbglm</code> function call that user made,
<code>model</code>	List with number of parameters and model matrices
<code>family</code>	List with functions such as loglikelihood and inverse link for the GLM family.
<code>samples</code>	A list with the MCMC samples drawn; organized by coeff type.

Note

1. The data must be in the 'long' or the 'list' format. In 'list' format there must be 2 keys 'lower' and 'upper' specifying two data.frames for the lower and upper level models respectively. Lower level contains data related to the fixed and random effects, while upper level contains data about the groups. In 'long' the 'upper' and 'lower' level data must be consolidated in a single data.frame (the upper level data will have to suitably replicated).
2. The 'formula' argument should be specified in the format: responseVar ~ choice specific variables with generic coefficients | individual specific variables | choice specific variables with choice specific coefficients. These are the 3 available variable types.
3. Random effects are mandatory in the formula.

Author(s)

Asad Hasan, Alireza S. Mahani

Examples

```
library(HBglm)
data("linear_list", package = "HBglm")
fm <- formula(y ~ x1 + x2 + x3 - 1 | z1 + z2 + z3 - 1 | grpID)
fm.fixed <- formula(y ~ f1 + f2)
samp.control = hbglm.sampler.control(num.samples = 20)
## Not run:
  fit <- hbglm(fm, linear_list, formula.fixed = fm.fixed, family="gaussian",
              sampler.control = samp.control, print.level=0)
  print(fit)

## End(Not run)
```

linear_list

Simulated data for a Gaussian 2-level HB regression model.

Description

A list object containing data fixed and random effects as well group level data for prior regression.

Usage

```
data(linear_list)
```

Format

A list containing :

lower A data.frame with fixed and random effects data for the model.

upper A data.frame with group level data for prior regression.

Source

Simulated data, generated from R code.

Index

*Topic **Hierarchical, Bayesian,
regression, multilevel**

HBglm, 2

*Topic **datasets**

linear_list, 4

coef.hbglm (HBglm), 2

HBglm, 2

hbglm (HBglm), 2

linear_list, 4

predict.hbglm (HBglm), 2

print.hbglm (HBglm), 2

print.summary.hbglm (HBglm), 2

summary.hbglm (HBglm), 2