Package ‘bayescopulareg’

November 30, 2020

Type Package
Title Bayesian Copula Regression
Version 0.1.3
Date 2020-11-30
Maintainer Ethan Alt <ethanalt@live.unc.edu>
Description Tools for Bayesian copula generalized linear models (GLMs).
   Regression parameters (including coefficients and dispersion parameters) are
   estimated via the adaptive random walk Metropolis approach developed by
Depends R (>= 3.6.0)
License GPL (>= 2)
Imports Rcpp (>= 1.0.3), stats
LinkingTo Rcpp, RcppArmadillo, RcppDist, mvtnorm
RoxygenNote 7.1.1
Encoding UTF-8
URL https://github.com/ethan-alt/bayescopulareg
BugReports https://github.com/ethan-alt/bayescopulareg/issues
NeedsCompilation yes
Author Ethan Alt [aut, cre],
   Yash Bhosale [aut]
Repository CRAN
Date/Publication 2020-11-30 17:10:06 UTC

R topics documented:

bayescopulaglm ....................................................... 2
predict.bayescopulaglm ........................................... 4

Index 6
bayescopulaglm

Sample from Bayesian copula GLM

Description

Sample from a GLM via Bayesian copula regression model. Uses random-walk Metropolis to update regression coefficients and dispersion parameters. Assumes Inverse Wishart prior on augmented data.

Usage

bayescopulaglm(
  formula.list,
  family.list,
  data,
  histdata = NULL,
  b0 = NULL,
  c0 = NULL,
  alpha0 = NULL,
  gamma0 = NULL,
  Gamma0 = NULL,
  S0beta = NULL,
  sigma0logphi = NULL,
  v0 = NULL,
  V0 = NULL,
  beta0 = NULL,
  phi0 = NULL,
  M = 10000,
  burnin = 2000,
  thin = 1,
  adaptive = TRUE
)

Arguments

formula.list A J-dimensional list of formulas giving how the endpoints are related to the covariates
family.list A J-dimensional list of families giving how each endpoint is distributed. See help(family)
data A data frame containing all response variables and covariates. Variables must be named.
histdata Optional historical data set for power prior on $\beta, \phi$
b0 Optional power prior hyperparameter. Ignored if is.null(histdata). Must be a number between (0,1] if histdata is not NULL
c0 A J-dimensional vector for $\beta | \phi$ prior covariance. If NULL, sets $c_0 = 10000$ for each endpoint
alpha0  A J-dimensional vector giving the shape hyperparameter for each dispersion parameter on the prior on φ. If NULL sets α0 = .01 for each dispersion parameter.

gamma0  A J-dimensional vector giving the rate hyperparameter for each dispersion parameter on the prior on φ. If NULL sets α0 = .01 for each dispersion parameter.

Gamma0  Initial value for correlation matrix. If NULL defaults to the correlation matrix from the responses.

S0beta  A J-dimensional list for the covariance matrix for random walk Metropolis on beta. Each matrix must have the same dimension as the corresponding regression coefficient. If NULL, uses solve(crossprod(X))

sigma0logphi  A J-dimensional vector giving the standard deviation on log(φ) for random walk Metropolis. If NULL defaults to 0.1

v0  An integer scalar giving degrees of freedom for Inverse Wishart prior. If NULL defaults to J + 2

V0  An integer giving inverse scale parameter for Inverse Wishart prior. If NULL defaults to diag(.001,J)

beta0  A J-dimensional list giving starting values for random walk Metropolis on the regression coefficients. If NULL, defaults to the GLM MLE

phi0  A J-dimensional vector giving initial values for dispersion parameters. If NULL, Dispersion parameters will always return 1 for binomial and Poisson models

M  Number of desired posterior samples after burn-in and thinning

burnin  burn-in parameter

thin  post burn-in thinning parameter

adaptive  logical indicating whether to use adaptive random walk MCMC to estimate parameters. This takes longer, but generally has a better acceptance rate

Value

A named list. ["betasample"] gives a J-dimensional list of sampled coefficients as matrices. ["phisample"] gives a M x J matrix of sampled dispersion parameters. ["Gammasample"] gives a J x J x M array of sampled correlation matrices. ["betasaccept"] gives a M x J matrix where each row indicates whether the proposal for the regression coefficient was accepted. ["phiaccept"] gives a M x J matrix where each row indicates whether the proposal for the dispersion parameter was accepted

Examples

set.seed(1234)
n <- 100
M <- 100

x <- runif(n, 1, 2)
y1 <- 0.25 * x + rnorm(100)
y2 <- rpois(n, exp(0.25 * x))

formula.list <- list(y1 ~ 0 + x, y2 ~ 0 + x)
family.list <- list(gaussian(), poisson())
data = data.frame(y1, y2, x)

## Perform copula regression sampling with default
## (noninformative) priors
sample <- bayescopulaglm(
    formula.list, family.list, data, M = M, burnin = 0, adaptive = F
)
## Regression coefficients
summary(do.call(cbind, sample$betasample))

## Dispersion parameters
summary(sample$phisample)

## Posterior mean correlation matrix
apply(sample$Gammasample, c(1,2), mean)

## Fraction of accepted betas
colMeans(sample$betaaccept)

## Fraction of accepted dispersion parameters
colMeans(sample$phiaccept)

---

predict.bayescopulaglm

**Predictive posterior sample from copula GLM**

**Description**

Sample from the predictive posterior density of a copula generalized linear model regression

**Usage**

```r
## S3 method for class 'bayescopulaglm'
predict(object, newdata, nsims = 1, ...)
```

**Arguments**

- `object` Result from calling `bayescopulaglm`
- `newdata` data.frame of new data
- `nsims` number of posterior draws to take. The default and minimum is 1. The maximum is the number of simulations in `object`
- `...` further arguments passed to or from other methods

**Value**

array of dimension c(n,J,nsims) of predicted values, where J is the number of endpoints
Examples

```r
set.seed(1234)
n <- 100
M <- 1000

x <- runif(n, 1, 2)
y1 <- 0.25 * x + rnorm(100)
y2 <- rpois(n, exp(0.25 * x))

formula.list <- list(y1 ~ 0 + x, y2 ~ 0 + x)
family.list <- list(gaussian(), poisson())
data = data.frame(y1, y2, x)

## Perform copula regression sampling with default
## (noninformative) priors
sample <- bayescopulaglm(
  formula.list, family.list, data, M = M
)
predict(sample, newdata = data)
```
Index

bayescopulaglm, 2
predict.bayescopulaglm, 4