Bayesian Robust Generalized Mixed Models for Longitudinal Data

Kuo-Jung Lee & Ray-Bing Chen & Hsing-Ming Chang
National Cheng Kung University at Taiwan

Keunbaik Lee & Chanmin Kim
Sungkyunkwan University at Korea

Abstract

BayesRGMM has the functionality to deal with the incomplete longitudinal studies on binary and ordinal outcomes that are measured repeatedly on subjects over time with drop-outs. Here, we briefly describe the background of methodology and provide an overview of the contents in **BayesRGMM**.

Keywords: Bayesian, Longitudinal, Mixed-Effect, R, Robust.

1. Main Methodology

Denote the response vector for the ith subject by $\boldsymbol{y}_i = (y_{i1}, \cdots, y_{it}, \cdots, y_{in_i})'$ where y_{it} is a response at time period t ($i = 1, \cdots, N$; $t = 1, \cdots, n_i$). Note that the model and associated methodology can be applicable to the unequally spaced times and the distinct number of observations from subject to subject.We assume that the responses on different subjects are independent. Also, we assume that y_{it}'s are conditionally independent given a random vector b_i, and that y_{it}'s. For categorical responses, we assume that y_{it} has an exponential family distribution so that generalized linear models (GLMs) can be specified by

$$
\begin{aligned}
& g\left\{E(y_{it})\right\} = x_{it}^T\beta + z_{it}^T b_i, \\
& b_i = (b_{i1}, \ldots, b_{iq})^T \overset{\text{indep.}}{\sim} N(0, v_i^{-1}\Sigma), \\
& v_i \overset{\text{indep.}}{\sim} \Gamma\left(\nu/2, \nu/2\right),
\end{aligned}
\tag{1}
$$

where β is a $p \times 1$ unknown mean parameter vectors, x_{it} is a $p \times 1$ corresponding vector of covariates, z_{it} is a $q \times 1$ vector, 0 is a $n_i \times 1$ zero vector, Σ is a $q \times q$ covariance matrix reflecting the subject variations, and $\Gamma(a, b)$ denotes the gamma distribution with shape parameter a and scale parameter b. In this paper, we consider the normal and binary responses and the corresponding links are identity and probit, respectively.

To employ Markov Chain Monte Carlo algorithm techniques for Bayesian estimates and reduce the computational complexity, we introduce a latent variable latent variable y_{it}^* to associate with the binary or ordinal outcome y_{it} as follows, respectively.

Binary outcome: The unobservable latent variable y^*_{it} and the observed binary outcome y_{it} are connected by:

$$y_{it} = \mathbf{I}_{(y^*_{it}>0)}, \quad t = 1, \ldots, n_i,$$

where $\mathbf{I}_A$ is the indicator of event A. Note that y_{it} is 1 or 0 according to the sign of y^*_{it}. We assume that the latent variable is associated with explanatory variable x_{it} and random effect z_{it} with two different approaches to explaining the correlation of the repeated measures within a subject in next two sections.

Ordinal outcome The atent variable y^*_{it} is associated with each ordinal response y_{it}. Hence, the probaility of $y_{it} = k$ is modeled through the probability of y^*_{it} falling into the interval of $(\alpha_{k-1}, \alpha_k]$, that is, given the random effect b_i,

$$y_{it} = k \text{ if } \alpha_{k-1} < y^*_{it} \leq \alpha_k \text{ for } k = 1, \ldots, K, \tag{2}$$

where $-\infty = \alpha_0 < \alpha_1 < \cdots < \alpha_K = \infty$. As consequence, we have the following result:

$$p(y_{it} = k|b_i) = p(\alpha_{k-1} < y^*_{it} \leq \alpha_k|b_i),$$

for $k = 1, \ldots, K$.

We assume that the latent variable is associated with explanatory variable x_{it} and random effect z_{it} with two different approaches to explaining the correlation of the repeated measures within a subject in next two sections.

1.1. Modified Cholesky Decomposition with Hypersphere Decomposition

We assume

$$y^*_{it} = x^T_{it}\beta + z^T_{it}b_i + \epsilon_{it},$$

where ϵ_{it}'s are prediction error and are assumed as

$$\boldsymbol{\epsilon}_i = (\epsilon_{i1}, \ldots, \epsilon_{in_i})^T \overset{indep.}{\sim} N(0, R_i)$$

with a correlation matrix R_i. Then the model (1) is equivalent to, for $i = 1, \ldots, N$ and $t = 1, \ldots, n_i$,

$$y_{it} = \begin{cases} 1, & y^*_{it} > 0; \\ 0 & \text{otherwhise.} \end{cases} \tag{3}$$

Let $\boldsymbol{y}^*_i = (y_{i1}, \ldots, y_{in_i})'$ and rewrite (3) in matrix form as

$$\boldsymbol{y}^*_i = X_i\beta + Z_ib_i + \boldsymbol{\epsilon}_i,$$

where X_i and Z_i are $n_i \times p$ and $n_i \times q$ matrices and defined as follows, respectively,

$$X_i = \begin{pmatrix} x^T_{i1} \\ \vdots \\ x^T_{in_i} \end{pmatrix}, Z_i = \begin{pmatrix} z^T_{i1} \\ \vdots \\ z^T_{in_i} \end{pmatrix}.$$

On account of identifiability, R_i is restricted as a correlation matrix. In addition to the diagonal elements equal to 1 and off-diagonal elements between -1 and 1, R_i is required to be a positive definite matrix. Moreover, the number of parameters to be estimated increases quadratically with the dimension of the matrix. In order to model R_i being positive definite, while alleviating the computational expensive, we propose a modeling of the correlation matrix using the hypersphere decomposition (HD) approach (Zhang, Leng, and Tang 2015). The correlation matrix R_i is reparameterized via hyperspherical coordinates (Zhang *et al.* 2015) by the following decomposition:

$$R_i = F_i F_i^T,$$

where F_i is a lower triangular matrix with the (t, j)th element f_{itj} given by

$$f_{itj} = \begin{cases} 1, & \text{for } t = j = 1; \\ \cos(\omega_{itj}), & \text{for } j = 1,\ t = 2, \cdots, n_i; \\ \cos(\omega_{itj}) \prod_{r=1}^{j-1} \sin(\omega_{itr}), & \text{for } 2 \leq j < t \leq n_i; \\ \prod_{r=1}^{j-1} \sin(\omega_{itr}), & \text{for } t = j;\ j = 2, \cdots, n_i. \end{cases}$$

Here ω_{itj}'s ($\in (0, \pi)$) are angle parameters for trigonometric functions, and the angle parameters are referred to hypersphere (HS) parameters.

As in Zhang *et al.* (2015), we consider the modeling of the angles ω_{itj}'s instead of the direct modeling of the correlation matrix, and the modeling can be directly interpreted for the correlation (Zhang *et al.* 2015). In order to obtain the unconstrained estimation of ω_{itj} and to reduce the number of parameters for facilitating the computation, we model the correlation structures of the responses in terms of the generalized linear models which are given by:

$$\log\left(\frac{\omega_{itj}}{\pi - \omega_{itj}}\right) = u_{itj}^T \delta, \tag{4}$$

where δ is $a \times 1$ vector of unknown parameter vector to model the HS parameters. Importantly, the proposed method reduces the model complexity and obtain fast-to-execute models without loss of accuracy. In addition, note that the design vector u_{itj} in (4) is used to model the HS parameters as functions of subject-specific covariates (Zhang *et al.* 2015; Pan and Mackenzie 2015). As a result, the design vector is specified in a manners similar to those used in heteroscedastic regression models. For example, time lag, $|t - j|$, in the design vector u_{itj} specifies higher lag models. We will introduce the priors of parameters in the model in Section 1.3.

1.2. Generalized Autoregressive and Moving-Averaging Model

In order to give the complete specification of the joint distribution, the latent random vectors $\boldsymbol{y}_i^* = (y_{i1}^*, \ldots, y_{in_i}^*)^T$ are jointly normally distributed given by:

$$\begin{aligned} y_{i1}^* &= x_{i1}^T \beta + \epsilon_{i1}, \\ y_{it}^* &= x_{it}^T \beta + z_{it}^T b_i + \sum_{j=1}^{u-1} \phi_{ij}(y_{i,t-j}^* - x_{i,t-j}^T \beta) + \sum_{s=1}^{v-1} \psi_{i,t-s} \epsilon_{i,t-s} + \epsilon_{it}, t = 1, \ldots, n_i, \end{aligned} \tag{5}$$

where ϕ_{ij}'s are generalized autoregressive parameters (GARPs) and ψ_{is}'s are generalized moving-average parameters (GMAPs). In addition, ϵ_{it}'s are prediction error and are assumed as

$$\boldsymbol{\epsilon}_i = (\epsilon_{i1}, \ldots, \epsilon_{in_i})^T \overset{indep.}{\sim} N(0, I_i),$$

where I_i is an $n_i \times n_i$ identity matrix. We can rewrite (5) in matrix form as

$$\Phi_i(\boldsymbol{y}_i^* - X_i\beta) = Z_i b_i + \Psi_i \boldsymbol{\epsilon}_i,$$

where X_i, $n_i \times p$, Z_i, $n_i \times q$, Φ_i, $n_i \times n_i$, Ψ_i, $n_i \times n_i$, are matrices and defined as follows, respectively,

$$X_i = \begin{pmatrix} x_{i1}^T \\ \vdots \\ x_{in_i}^T \end{pmatrix}, \quad Z_i = \begin{pmatrix} z_{i1}^T \\ \vdots \\ z_{in_i}^T \end{pmatrix}$$

$$\Phi_i = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ -\phi_{i1} & 1 & 0 & \dots & 0 & 0 \\ -\phi_{i2} & -\phi_{i1} & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & -\phi_{i,u-2} & \dots & 1 & 0 \\ 0 & \dots & -\phi_{i,u-1} & \dots & -\phi_{i1} & 1 \end{pmatrix}, \quad \Psi_i = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ \psi_{i1} & 1 & 0 & \dots & 0 & 0 \\ \psi_{i2} & \psi_{i1} & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & \psi_{i,v-2} & \dots & 1 & 0 \\ 0 & \dots & \psi_{i,v-1} & \dots & \psi_{i1} & 1 \end{pmatrix}$$

Note that Φ_i and Ψ_i uniquely exist and are respectively called the generalized autoregressive parameter matrix (GARPM) and moving-average parameter matrix (GMAPM).

The density of the latent variable $\boldsymbol{y}^*$ conditional on the random effect $b = (b_1, \dots, b_q)$ is given by

$$p(\boldsymbol{y}^*|\boldsymbol{b}, \theta) = \prod_{i=1}^{N}\prod_{t=1}^{n_i} f(y_{it}^*; \mu_{it}, I_i),$$

where $\theta = (\beta, \nu, \Sigma, \phi, \psi)$ denote the collection of model parameters, $\mu_{it} = x_{it}^T\beta + z_{it}^T b_i$ and $f(\cdot)$ is the multivariate normal density function.

1.3. Bayesian Methods

The density of the latent variable $\boldsymbol{y}^*$ conditional on the random effect $b = (b_1, \dots, b_q)$ is given by

$$p(\boldsymbol{y}^*|\boldsymbol{b}, \theta) = \prod_{i=1}^{N}\prod_{t=1}^{n_i} f(y_{it}^*; \mu_{it}, I_i),$$

where θ denote the collection of model parameters, $\mu_{it} = x_{it}^T\beta + z_{it}^T b_i$ and $f(\cdot)$ is the multivariate normal density function.

To complete the Bayesian specification of the model, we use proper prior distributions instead of noninformative priors, in order to guarantee the propriety of posterior distribution. The prior distributions for β, Σ, and δ in the model for binary outcome are given by:

$$\begin{aligned} \beta &\sim N_p(0, \sigma_\beta^2 \mathrm{I}), \\ \Sigma &\sim \mathcal{IW}(\nu_b, \Lambda^{-1}), \end{aligned}$$

where σ_β^2 and σ_δ^2 are large to be noninformative (Daniels and Zhao 2003), I is the identity matrix corresponding to the dimension of the parameter, and Λ is the positive definite scale matrix. Here $N_m(\mu, \Omega)$ denotes the m-variate multivariate normal distribution with a mean vector μ and a covariance matrix Ω, and $\mathcal{IW}(\nu, \Lambda^{-1})$ denotes the inverse Wishart distribtion with degrees of freedom ν and a symmetric positive definite $q \times q$ scale matrix.

The prior of the parameters in correlation matrix for two different correlation structures are

MCD In the the case of modified Cholesky decomposition with hypersphere decomposition, we assume $\delta \sim N_a(0, \sigma_\delta^2 \mathrm{I})$.

ARMA In ARMA correlation structure, the non-informative priors is assumed for temporal parameters in GARPM ϕ's and GAMPM ψ's with constraints on them to ensure the stationary.

Furthermore, in the ordinal outcome a prior for α is provided

$$\alpha \sim N_{K-1}(0, \sigma_\alpha^2 I)\mathbf{I}_{(-\infty<\alpha_1<\cdots<\alpha_{K-1}<\infty)},$$

where σ_α^2 is prespecified.

2. Implementation

The aim of this section is to provide a detailed step-by-step R in simulation studies to highlight the most important features of package **BayesRGMM**, and to show how to extract the most important results. This section can also be considered as a user manual which allows the readers to run their own similar analyses involving only a mild modification of the example code.

The **BayesRGMM** package contains four core functions. The main function `BayesRobustProbit` carries out the entire MCMC procedure, and outputs the posterior samples and estimates for model parameters along with several useful estimated information criterion statistics. Internally, most of the calculation is provided by a compiled C++ code to reduce the computational time.User-friendly summary function `BayesRobustProbitSummary` that summarizes model estimation outcomes is equipped with `BayesRobustProbit` and `BayesCumulativeProbitHSD`. It provides basic posterior summary statistics such as the posterior point and confidence interval estimates of parameters and the values of information criterion statistics for model comparison. The function `SimulatedDataGenerator` and `SimulatedDataGenerator.CumulativeProbit` are used to generate simulated binary and ordinal data, respectively, for simulation studies. `CorrMat.HSD` is applied to calculate the correlation matrix in MCD model structure. In this section, we focus primarily on introducing the those functions, and demonstrate their usage with numerical experiments.

2.1. Simulation Studies

In the simulation, we demonstrate the use of functions in the **BayesRGMM** package.We consider a simple random intercept model $q = 1$ with the regression coefficient vector of size $p = 4$ given by

$$\beta = (-0.2, -0.3, 0.8, -0.4)',$$

where x_{it}'s are independently generated from $N(0, 1)$. In addition, v_i is independently simulated from $\Gamma(3, 3)$, b_i is from $N(0, v_i \times 0.5)$, and $z_{it} = 1$ for $i = 1, \ldots, n$. That is, b's correspond to a Student's t-distribution with the degrees of freedom equal to 6. We then generate the responses based on (1).

In addition to the different correlation structures, we also consider a data that is missing completely at random (MCAR). We set the missing machines as follows

$$\eta_{it} = -1.5 \times y_{t-1,i} + 1.2 \times y_{t-2,i};$$

Then the missing probability depends on η_{it}'s defined as

$$p_{it}^{\text{miss}} = \frac{e^{\eta_{it}}}{1 + e^{\eta_{it}}}.$$

The data for subject i at time point t is missing according to three observed responses for the subject.

2.2. Simulation 1: MCD Correlation Structure

The correlation matrix R_i is created based on the given values

$$\delta = (-0.5, -0.3)' \quad \text{and} \quad u_{itj} = (\mathbf{I}\{|t-s| = 1\}, \mathbf{I}\{|t-s| = 2\})' \tag{6}$$

```
R> # Simulation study for MCD correlation structure
R> library(BayesRGMM)
R> rm(list=ls(all=TRUE))
R> Fixed.Effs = c(-0.2, -0.3, 0.8, -0.4)
R> P = length(Fixed.Effs)
R> q = 1
R> T = 5
R> N = 100
R> num.of.iter = 100
R> HSD.para = c(-0.5,  -0.3)
R> a = length(HSD.para)
R> w = array(runif(T*T*a), c(T, T, a))
R> for(time.diff in 1:a)
+         w[, , time.diff]=1*(as.matrix(dist(1:T, 1:T, method="manhattan"))
+                             ==time.diff)
R> HSD.sim.data = SimulatedDataGenerator(Num.of.Obs = N, Num.of.TimePoints = T,
+       Fixed.Effs = Fixed.Effs, Random.Effs = list(Sigma = 0.5*diag(1), df=3),
+       Cor.in.DesignMat = 0., Missing = list(Missing.Mechanism = 2,
+       RegCoefs = c(-1.5, 1.2)), Cor.Str = "HSD",
+       HSD.DesignMat.para = list(HSD.para = HSD.para, DesignMat = w))
R> hyper.params = list(
+         sigma2.beta = 1,
+         sigma2.delta = 1,
+         v.gamma = 5,
+         InvWishart.df = 5,
+         InvWishart.Lambda = diag(q) )
R> HSD.output = BayesRobustProbit(fixed = as.formula(paste("y~-1+",
+             paste0("x", 1:P, collapse="+"))), data=HSD.sim.data$sim.data,
```

```
+              random = ~ 1, HS.model = ~IndTime1+IndTime2, Robustness=TRUE, subset = NULL
+              na.action='na.exclude', hyper.params = hyper.params,
+              num.of.iter = num.of.iter, Interactive = FALSE)

Start running MCMC procedure:

Finish MCMC Procedure.

Data Descriptives:
Longitudinal Data Information:
Number of Observations:  372        Number of Covariates:  3
Number of subjects: 100

R> original = options(digits = 4)
R> Model.Estimation = BayesRobustProbitSummary(HSD.output)
R> cat("\nCoefficients:\n")

Coefficients:

R> print(Model.Estimation$beta.est.CI)

   PostMean StErr 2.5% 97.5%
x1     -0.2  0.03 -0.5  0.02
x2     -0.3  0.04 -0.6  0.04
x3      0.8  0.05  0.3  1.00
x4     -0.4  0.02 -0.6 -0.21

R> cat("\nParameters in HSD model:\n")

Parameters in HSD model:

R> print(Model.Estimation$delta.est.CI)

         PostMean StErr  2.5% 97.5%
IndTime1     -0.2  0.04 -0.39  0.05
IndTime2      0.4  0.05  0.07  0.72

R> cat("\nRandom effect: \n")

Random effect:

R> print(Model.Estimation$random.cov)

            Variance
(Intercept)      0.7
```

```
R> cat("\nModel Information:\n")

Model Information:

R> print(Model.Estimation$model.info)

 logL AIC BIC CIC DIC MPL RJR ACC
 -128 467 744 0.6 354   5   1 0.9

R> cat("\nEstimate of Ri: \n")

Estimate of Ri:

R> print(Model.Estimation$Ri, quote = FALSE)

     [,1] [,2] [,3] [,4] [,5]
[1,] 1.0  0.2  -0.3 0.0  0.0
[2,] 0.2  1.0  0.1  -0.3 0.0
[3,] -0.3 0.1  1.0  0.1  -0.3
[4,] 0.0  -0.3 0.1  1.0  0.1
[5,] 0.0  0.0  -0.3 0.1  1.0

R> options(original)
R>
```

2.3. Simulation 2: ARMA Correlation Structure

To model the serial dependence for the repeated measurement, we consider an ARMA(1, 1) correlation structure with

$$\phi = 0.4, \qquad \text{and} \qquad \psi = 0.2.$$

```
R> library(BayesRGMM)
R> rm(list=ls(all=TRUE))
R> Fixed.Effs = c(-0.2,-0.8, 1.0, -1.2)
R> P = length(Fixed.Effs)
R> q = 1
R> T = 10
R> N = 100
R> num.of.iter = 100
R> ARMA.sim.data = SimulatedDataGenerator(Num.of.Obs = N, Num.of.TimePoints = T,
+   Fixed.Effs = Fixed.Effs, Random.Effs = list(Sigma = 0.5*diag(1), df=3),
+   Cor.in.DesignMat = 0., list(Missing.Mechanism = 2, RegCoefs = c(-1.5, 1.2)),
+   Cor.Str = "ARMA", ARMA.para=list(AR.para = 0.4, MA.para=0.2))
R> ARMA.output = BayesRobustProbit(fixed = as.formula(paste("y~-1+",
+   paste0("x", 1:P, collapse="+"))), data=ARMA.sim.data$sim.data, random = ~ 1,
+   Robustness=TRUE, subset = NULL, na.action='na.exclude', arma.order = c(1, 1),
+   num.of.iter = num.of.iter, Interactive = FALSE)
```

```
Data Descriptives:
Longitudinal Data Information:
Number of Observations:  775         Number of Covariates:  3
Number of subjects: 100

Start running MCMC procedure:

Finish MCMC Procedure.

R> original = options(digits = 4)
R> Model.Estimation = BayesRobustProbitSummary(ARMA.output)
R> cat("\nCoefficients:\n")

Coefficients:

R> print(Model.Estimation$beta.est.CI)

   PostMean StErr 2.5% 97.5%
x1    -0.09  0.05 -0.5   0.2
x2    -0.32  0.14 -0.8   0.8
x3     0.96  0.06  0.5   1.2
x4    -1.09  0.08 -1.4  -0.5

R> cat("\nAMRA parameters:\n\n")

AMRA parameters:

R> print(Model.Estimation$arma.est)

      PostMean StErr  2.5% 97.5%
phi 1     0.02 0.007 -0.02  0.05
psi 1     0.13 0.023 -0.05  0.28

R> cat("\nRandom effect: \n")

Random effect:

R> print(Model.Estimation$random.cov)

            Variance
(Intercept)      0.5

R> cat("\nModel Information:\n")

Model Information:
```

```
R> print(Model.Estimation$model.info)

 logL AIC BIC   CIC DIC MPL RJR ACC
 -171 554 830 5e-10 457  12   1 0.9

R> options(original)
```

2.4. Ordinal Outcome

We consider a simple random intercept model ($q = 1$). For $k = 1, 2, 3$ and $t = 1, \ldots, n_i$, the model is given by:

$$y^*_{it} = \beta_1 Time_{it} + \beta_2 Group_i + \beta_3 Time_{it} \times Group_i + b_{i0} + \epsilon_{it}, \tag{7}$$

$$b_{i0} \sim N(0, \sigma^2_b), \tag{8}$$

$$\epsilon_i = (\epsilon_{i1}, \ldots, \epsilon_{in_i})^T \sim N(0, R_i), \tag{9}$$

where $Time_{it} \sim N(0, 1)$ and $Group_i$ equals 0 or 1 with approximately the same sample size for each group. The true parameters in the simulations are as below:

$$(\beta_{01}, \beta_{02}) = (-0.5, 0.5); \quad (\beta_1, \beta_2, \beta_3) = (-0.1, 0.1, -0.1); \quad \sigma^2_b = 0.2.$$

The model for correlation matrix R_i is given by

$$\log\left(\frac{\omega_{itj}}{\pi - \omega_{itj}}\right) = \delta_1 1_{(|t-j|=1)} + \delta_2 1_{(|t-j|=2)}, \tag{10}$$

where $(\delta_1, \delta_2) = (-0.9, -0.6)$.

We consider a data that is missing completely at random (MCAR) with a machine defined by

$$\eta_{it} = -0.7 \times y_{t-1,i} - 0.2 \times y_{t-2,i} - 0.1 \times y_{t-3,i};$$

Then the missing probability depends on η_{it}'s defined as

$$p^{\text{miss}}_{it} = \frac{e^{\eta_{it}}}{1 + e^{\eta_{it}}}.$$

The data for subject i at time point t is missing according to three observed responses for the subject.

```
R> library(BayesRGMM)
R> rm(list=ls(all=TRUE))
R> Fixed.Effs = c(-0.1, 0.1, -0.1) #c(-0.8, -0.3, 1.8, -0.4) #c(-0.2,-0.8, 1.0, -1.2)
R> P = length(Fixed.Effs)
R> q = 1 #number of random effects
R> T = 7 #time points
R> N = 100 #number of subjects
R> Num.of.Cats = 3 #in KBLEE simulation studies, please fix it.
```

```
R> num.of.iter = 1000 #number of iterations
R> HSD.para = c(-0.9, -0.6) #the parameters in HSD model
R> a = length(HSD.para)
R> w = array(runif(T*T*a), c(T, T, a)) #design matrix in HSD model
R> for(time.diff in 1:a)
+ w[, , time.diff] = 1*(as.matrix(dist(1:T, 1:T, method="manhattan")) ==time.diff)
R> x = array(0, c(T, P, N))
R> for(i in 1:N){
+     #x[,, i] = t(rmvnorm(P, rep(0, T), AR1.cor(T, Cor.in.DesignMat)))
+     x[, 1, i] = 1:T
+     x[, 2, i] = rbinom(1, 1, 0.5)
+     x[, 3, i] = x[, 1, i]*x[, 2, i]
+ }
R> DesignMat = x
R> #Generate a data with HSD model
R>
R>
R> #MAR
R> CPREM.sim.data = SimulatedDataGenerator.CumulativeProbit(Num.of.Obs = N,
+     Num.of.TimePoints = T, Num.of.Cats = Num.of.Cats, Fixed.Effs = Fixed.Effs,
+     Random.Effs = list(Sigma = 0.5*diag(1), df=3), DesignMat = DesignMat,
+     Missing = list(Missing.Mechanism = 2, MissingRegCoefs=c(-0.7, -0.2, -0.1)),
+     HSD.DesignMat.para = list(HSD.para = HSD.para, DesignMat = w))
R> print(table(CPREM.sim.data$sim.data$y))

  1   2   3
147 174 186

R> print(CPREM.sim.data$classes)

      0%   33.33%   66.67%     100%
-10.6644  -1.4533  -0.1887   4.1869

R> BCP.output = BayesCumulativeProbitHSD(
+     fixed = as.formula(paste("y~", paste0("x", 1:P, collapse="+"))),
+     data=CPREM.sim.data$sim.data, random = ~ 1, Robustness = TRUE,
+     subset = NULL, na.action='na.exclude', HS.model = ~IndTime1+IndTime2,
+     hyper.params=NULL, num.of.iter=num.of.iter, Interactive = FALSE)

Start reading Data
End reading Data
Read Hyperparameters.
Start running MCMC procedure:

Finish MCMC Procedure.
```

```
Finish Parameter Estimation.
AIC = 1848.64        BIC = 2122.19        CIC = 31.6191        logL = -819.322
DIC = inf        RJR = 182.395        MPL = 0        ACC = 0.627219

Data Descriptives:
Longitudinal Data Information:
Number of Observations:  507        Number of Covariates:  2
Number of subjects: 100

R> BCP.Est.output = BayesRobustProbitSummary(BCP.output)
R> BCP.Est.output

$beta.est.CI
   PostMean StErr 2.5% 97.5%
x1      0.1 0.007 -0.2   0.9
x2      1.1 0.028  0.0   4.6
x3     -0.1 0.006 -0.8   0.1

$delta.est.CI
         PostMean StErr 2.5% 97.5%
IndTime1    -0.02 0.001 -0.2  0.02
IndTime2    -0.01 0.002 -0.2  0.05

$model.info
 logL  AIC  BIC CIC DIC MPL RJR ACC
 -819 1849 2122  32 Inf   0 182 0.6

$random.cov
            Variance
(Intercept)       10

$Ri
     [,1]  [,2]  [,3]  [,4]  [,5]  [,6]
[1,] "1.0" "0.0" "0.0" "0.0" "0.0" "0.0"
[2,] "0.0" "1.0" "0.0" "0.0" "0.0" "0.0"
[3,] "0.0" "0.0" "1.0" "0.0" "0.0" "0.0"
[4,] "0.0" "0.0" "0.0" "1.0" "0.0" "0.0"
[5,] "0.0" "0.0" "0.0" "0.0" "1.0" "0.0"
[6,] "0.0" "0.0" "0.0" "0.0" "0.0" "1.0"
```

References

Daniels MJ, Zhao YD (2003). "Modelling the random effects covariance matrix in longitudinal data." *Statistics in Medicine*, **22**, 1631–1647.

Pan J, Mackenzie G (2015). "On modelling mean-covariance structures in longitudinal studies." *Biometrika*, **90**, 239–244.

Zhang W, Leng C, Tang CY (2015). "A joint modelling approach for longitudinal studies." *Journal of Royal Statistical Society, Series B*, **77**, 219–238.

Affiliation:

Kuo-Jung Lee
Deparmtne of Statistis and Institute of Data Science
National Cheng Kung University
E-mail: kuojunglee@ncku.edu.tw
URL: https://sites.google.com/view/kuojunglee

Ray-Bing Chen
Deparmtne of Statistis and Institute of Data Science
National Cheng Kung University, Taiwan
E-mail: rbchen@ncku.edu.tw
URL: https://sites.google.com/view/ray-bingchenswebsite/home

Keunbaik Lee
Department of Statistics
Sungkyunkwan University, Korea
E-mail: keunbaik@skku.edu
URL: https://ecostat.skku.edu/eng_ecostat/intro/faculty_stat.do

Chanmin Kim
Department of Statistics
Sungkyunkwan University, Korea
E-mail: ckim.bayes@gmail.com
URL: https://ecostat.skku.edu/eng_ecostat/intro/faculty_stat.do