Ứng dụng BayesVL v0.6.5 mô phỏng MCMC với bài toán burden ~ res + insured sử dụng dữ liệu thực 1042 quan sát

Vương Quân Hoàng
Lã Việt Phương

AISDL (Vuong & Associates)
SDAG, Centre for Interdisciplinary Social Research (Phenikaa University)

Note 2a

Hà Nội, 27-4-2019 (10:30 PM version 5)

Bài viết trình bày một mô hình và ứng dụng mô phỏng BayesVL v0.6.5 [**1**] trên môi trường R và Stan MCMC, cho bài toán thực tế [**2**] với dữ liệu thực tế, gồm 1042 điểm dữ liệu [**3**]. Nội dung của bài là một phần của tài liệu do AISDL biên soạn phục vụ hướng dẫn sử dụng "The BayesVL R Package", nhằm thúc đẩy việc sử dụng phương pháp thống kê Bayesian trong KHXH&NV.

Một phần mã chương trình [**1**] được viết dựa trên Statistical Rethinking của Richard McElreath [**4**] và BNLearn của Scuttari [**5**]. Công việc mô phỏng cơ bản sử dụng kỹ thuật Hamiltonian Markov chain Monte Carlo, dựa trên Stan và R.

1. Statement of the problem

Bài toán sử dụng cơ sở dữ liệu "Health Care, Medical Insurance, and Economic Destitution: A Dataset of 1042 Stories" [**3**].

Công việc: tiến hành lại việc đánh giá áp lực kinh kế lên bệnh nhân như thế nào khi có hoặc không có nhà ở và bảo hiểm y tế, dựa trên mô phỏng MCMC (thay vì ước lượng frequentist như trong [**2**]).

2. Dữ liệu và đánh giá mô hình (dataset and estimations)

Một trong các mô hình cơ sở, đơn giản nhất trong [**2**], trình bày dưới đây.

a. Dữ liệu và xây dựng mô hình

```
data1 <- read.csv("/Statistics/1042/1042data/1042data.csv", header = TRUE)
head(data1)
```

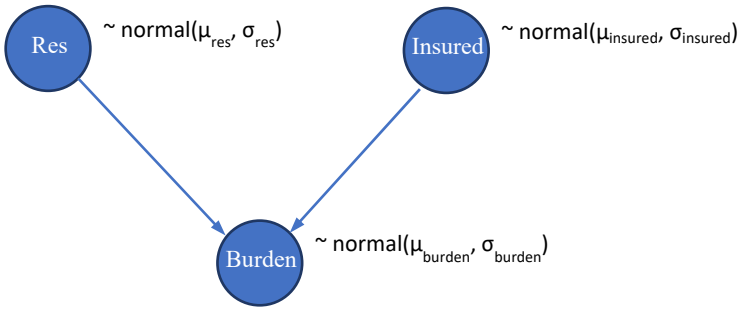
Các biến sử dụng trong mô hình:

- Res: bệnh nhân có phải là cư dân ở khu vực khám chữa bệnh không
- Insured: bệnh nhân có chế độ bảo hiểm không
- Burden: áp lực tài chính lên bệnh nhân và gia đình

Xác suất có điều kiện giữa áp lực kinh tế burden và điều kiện nhà ở (res) và bảo hiểm (insured) thể hiện như sau:

$$P(\text{burden} \mid \text{res}, \text{insured}) \propto P(\text{res} \mid \text{burden}, \text{insured})\, P(\text{burden} \mid \text{insured})$$

Nếu dựng mô hình quan hệ giữa nhà ở, bảo hiểm và áp lực kinh tế ta có thể thể hiện ở dạng lưới sau [**5**]:



Sử dụng package bayesvl trên R mô tả mô hình quan hệ ở trên:

```
# Add nodes to model
model <- bayesvl()
model <- bvl_addNode(model, "Burden", "norm")
model <- bvl_addNode(model, "Res", "cat")
model <- bvl_addNode(model, "Insured", "cat")

# Add arcs to model
model <- bvl_addArc(model, "Res", "Burden", "slope")
```

```
model <- bvl_addArc(model, “Insured”, “Burden”, “slope”)
```

b. Đánh giá mô hình

- Đánh giá mô hình bằng bnlearn

Tính xác suất có điều kiện các biến trong mô hình:

```
bvl_bnBayes(model, data1[c("Res","Burden","Insured")])
```

Kết quả:

```
Bayesian network parameters

  Parameters of node Burden (multinomial distribution)

Conditional probability table:

, , Insured = No

     Res
Burden         No        Yes
    A 0.058351178 0.308659218
    B 0.323875803 0.420391061
    C 0.580835118 0.252793296
    D 0.036937901 0.018156425

, , Insured = Yes

     Res
Burden         No        Yes
    A 0.147027601 0.746960486
    B 0.244692144 0.216058764
    C 0.588641189 0.035714286
    D 0.019639066 0.001266464

Parameters of node Res (multinomial distribution)

Conditional probability table:
       No       Yes
0.4458175 0.5541825

  Parameters of node Insured (multinomial distribution)

Conditional probability table:
       No       Yes
0.3070342 0.6929658
```

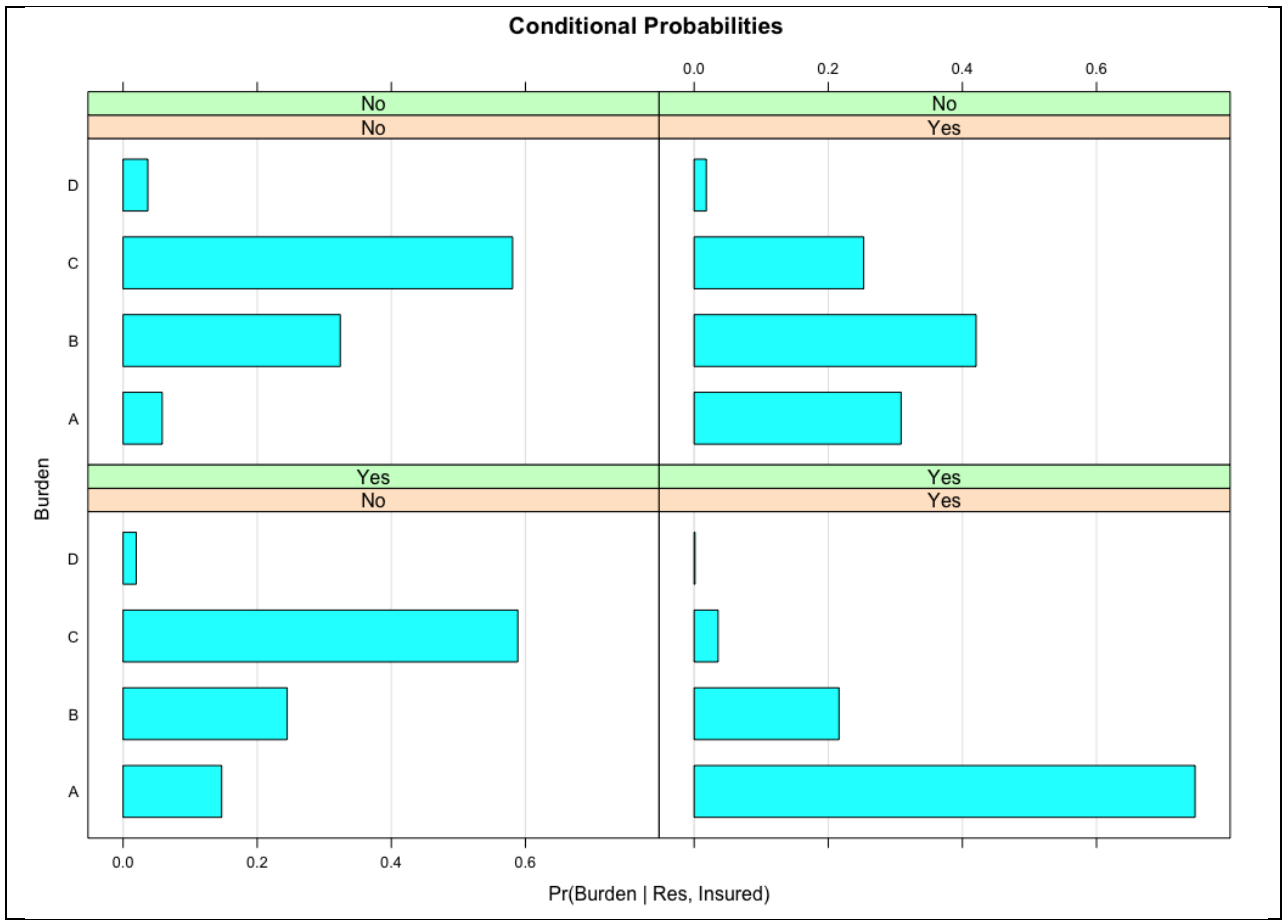
Hãy xem xét một bệnh nhân có nhà ở gần bệnh viện nhưng không có bảo hiểm, posterior distribution của áp lực tài chính của bệnh nhận này:

```
p(Burden|Res=”Yes”,Insured="No") = p(Res="Yes"|Burden,Insured="No")p(Burden|Insured="No")
```

Nếu vẽ các xác suất có điều kiện này:

```
bvl_bnBarchart(model, data1[c("Res","Burden","Insured")])
```

Hình giúp so sánh các phân phối xác suất theo cách tính frequentist:



Nếu chỉ đánh giá sơ bộ trên dữ liệu như trên, có thể dễ nhận thấy trên dữ liệu thống kê đối với người có nhà và bảo hiểm, áp lực tài chính khám chữa bệnh thấp (burden tập trung cao mở mức A).

Đánh giá sức mạnh kết nối của mô hình quan hệ:

```
> bvl_bnStrength(model, data1[c("Res","Burden","Insured")])

   from    to    strength
```

```
1    Res Burden 1.682735e-81
2 Insured Burden 2.303670e-20
```

Chức năng này gọi package bnlearn để đo lường quan hệ giữa 2 biến trong lưới model thông qua xác suất tương ứng từng cung (arc). Giá trị strength được thể hiện chính là giá trị p-value.

Mô hình hiện tại có 2 arc, các giá trị p-values đều nhỏ hơn 0,05 và support rất tốt từ dữ liệu.

c. Mô hình hồi quy bayesian

Giả sử chúng ta chọn 1 mô hình hồi quy tuyến tính đơn giản của biến y với 1 biến covariate x, phương trình toán học sẽ có dạng:

$$y_i = \alpha + \beta x_i + \epsilon_i$$

Trong đó $\epsilon_i \sim N(0, \sigma)$.

Viết cách khác ta có thể có phân phối của y:

$$y_i \sim N(\alpha+\beta x_i, \sigma)$$

Sử dụng tập dữ liệu khảo sát $i \in I$.

Với mô hình quan hệ đã tạo ở trên, nếu viết ở dạng hồi quy tuyến tính, ta có phương trình mô phỏng như sau:

$$y_{burden}[i] = \alpha_{burden} + \beta_{res} x_{res}[i] + \beta_{insured} x_{insured}[i] + \epsilon_{burden}[i]$$

Trong phương trình, ta thấy có 2 hệ số β_{res}, và $\beta_{insured}$ là 2 hệ số hồi quy "slope" là hệ số ảnh hưởng đến hệ số góc của đường hồi quy. Khi tạo mô hình quan hệ các biến ở trên ta đã định nghĩa arc với loại quan hệ "slope":

```
model <- bvl_addArc(model, "Res", "Burden", "slope")
model <- bvl_addArc(model, "Insured", "Burden", "slope")
```

Vì vậy, để tạo mô hình hồi quy trên bayesvl chỉ cần thực hiện lệnh:

```
#  Generate the stan code for model
model_string <- bvl_model2Stan(model)
cat(model_string)
```

model_string chứa code Stan cho mô hình hồi quy tuyến tính.

Code Stan được tạo:

```
data{
   // Define variables in data
```

```
    int<lower=1> Nobs;  // Number of observations (an integer)
    real Burden[Nobs];   // outcome variable
    real Res[Nobs];
    real Insured[Nobs];
}
parameters{
    // Define parameters to estimate
    real<lower=0> sigma_Burden;
    real a_Burden;
    real b_Res_Burden;
    real b_Insured_Burden;

}
transformed parameters{
    // Transform parameters
    real mu_Burden[Nobs];
    for (i in 1:Nobs) {
      mu_Burden[i] = a_Burden + b_Res_Burden * Res[i] + b_Insured_Burden * Insured[i];
    }
}
model{
    // Priors
    a_Burden ~ normal(0,100);
    b_Res_Burden ~ normal(0,100);
    b_Insured_Burden ~ normal(0,100);

    // Likelihoods
    Burden ~ normal(mu_Burden, sigma_Burden);
}
```

Chạy mô phỏng mô hình:

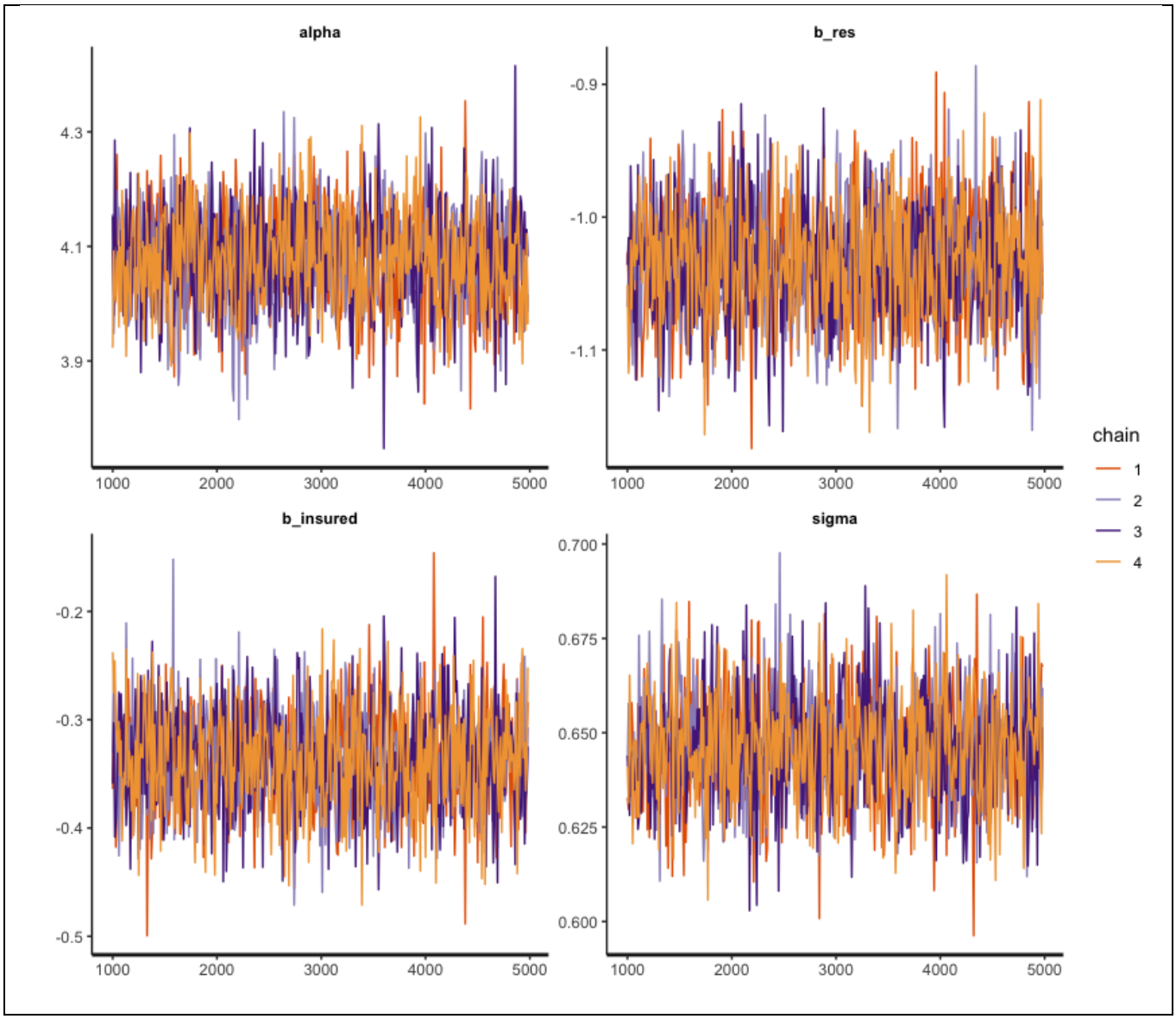
```
dat1042 <- with(data1,
          list(Nobs       = length(Res),
               Res        = as.numeric(Res),
               Insured    = as.numeric(Insured),
               Burden             = as.numeric(Burden)))

# Fit the model
model <- bvl_modelFit(model, dat1042, warmup = 2000, iter = 5000, chains = 4, cores = 4)
```

d. Đánh giá mô hình

Đánh giá sự hội tụ của mô hình có thể thực hiện thông qua trace plot:

```
traceplot(fit)
```

Hình ảnh các xích xác suất mô phỏng:



Ở mô hình này ta thấy trace plots tốt, không có các chuỗi dị thường divergent chains

Bước tiếp theo ta có thể xem tổng hợp kết quả mô phỏng bằng lệnh summary:

```
summary(fit)
```

Thu được kết quả cần đọc như sau:

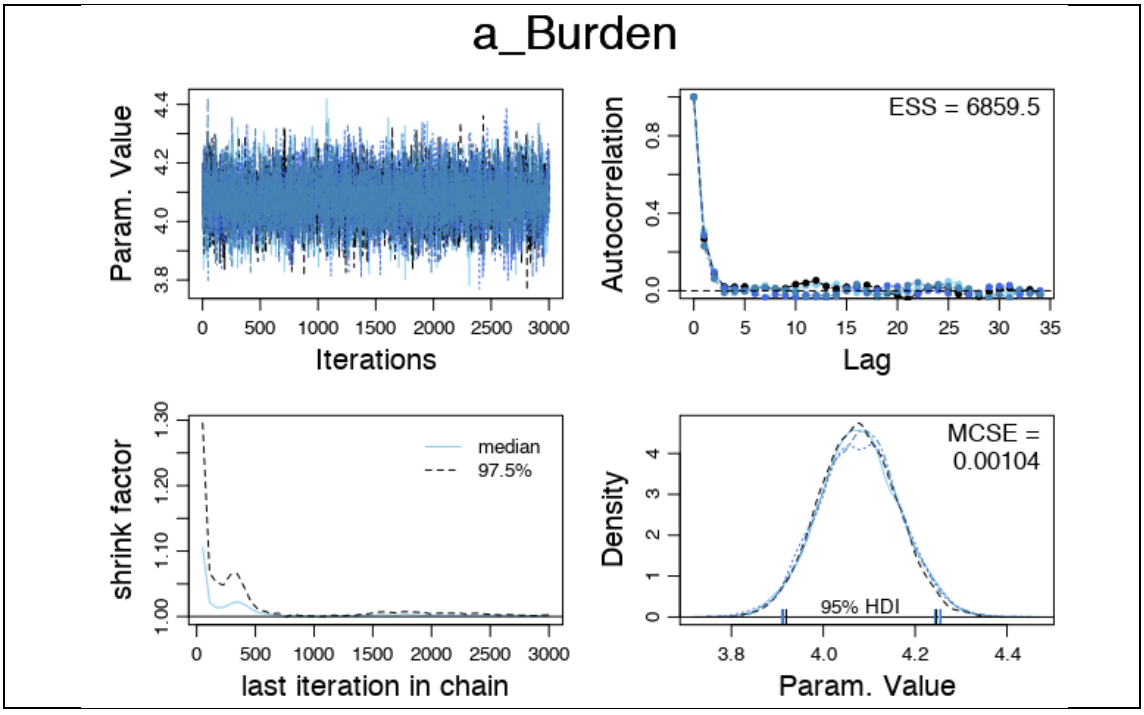
```
> summary(fit)

Model Info:
Nodes:          3
Arcs:           2
Scores:
```

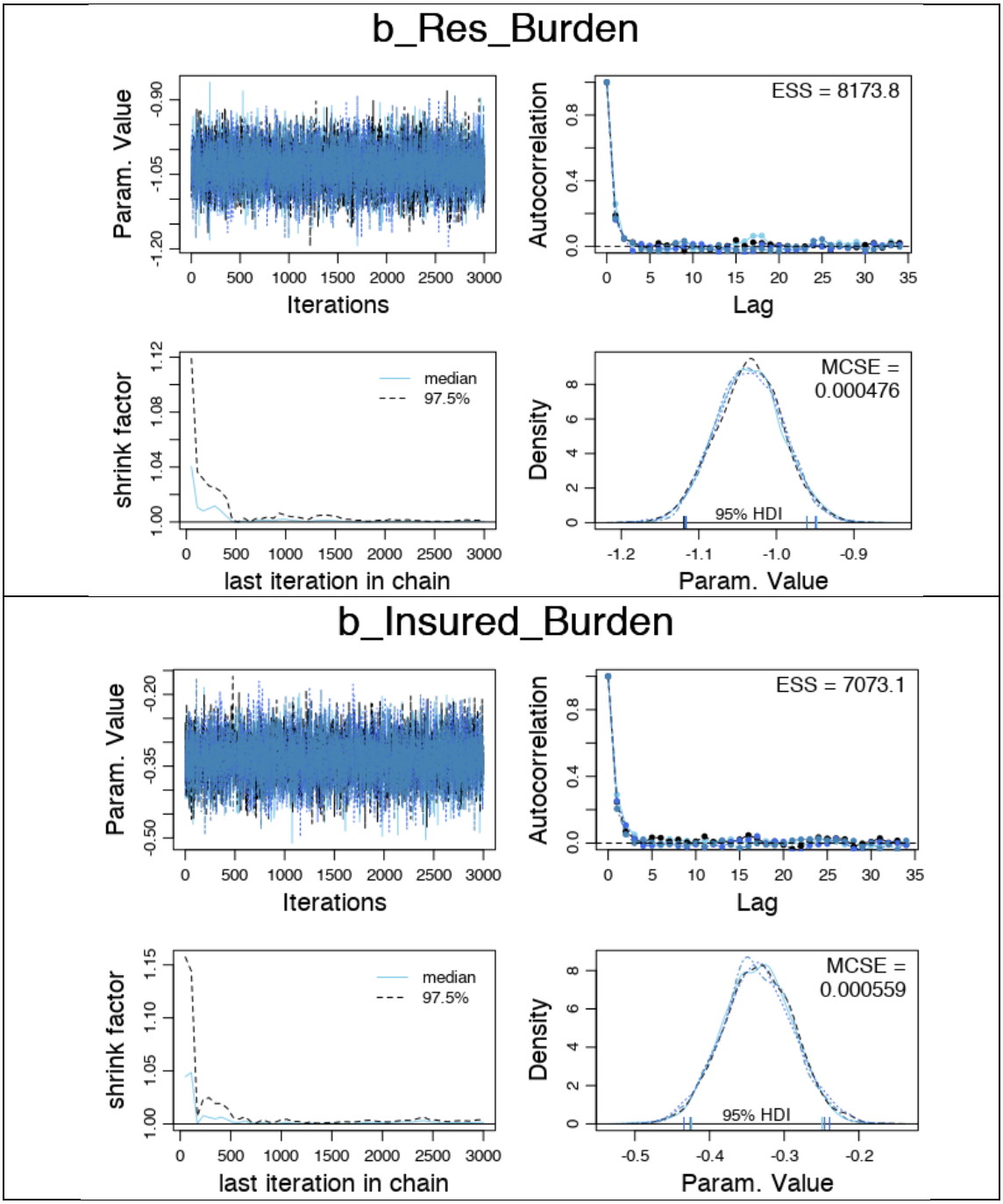
```
Formulas:       Burden ~ alpha + b_Res * Res + b_Insured * Insured

Estimates:
4 chains, each with iter=5000; warmup=1000; thin=10;

          mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
alpha     4.08       0 0.09  3.91  4.02  4.08  4.14  4.24  1533    1
b_res    -1.03       0 0.04 -1.12 -1.06 -1.03 -1.00 -0.95  1675    1
b_insured -0.34      0 0.05 -0.43 -0.37 -0.34 -0.31 -0.25  1592    1
sigma     0.65       0 0.01  0.62  0.64  0.65  0.66  0.68  1591    1
```

Chúng ta có thể thấy rằng effective sample size (n_eff) là ổn (trên 1000 thường là một dấu hiệu tốt), một số liệu chẩn đoán khác là giá trị Rhat cũng cho thấy sự hội tụ (Rhat bằng 1 là một dấu hiệu tốt; khi lớn hơn 1.1 có thể chỉ ra là có vấn đề).

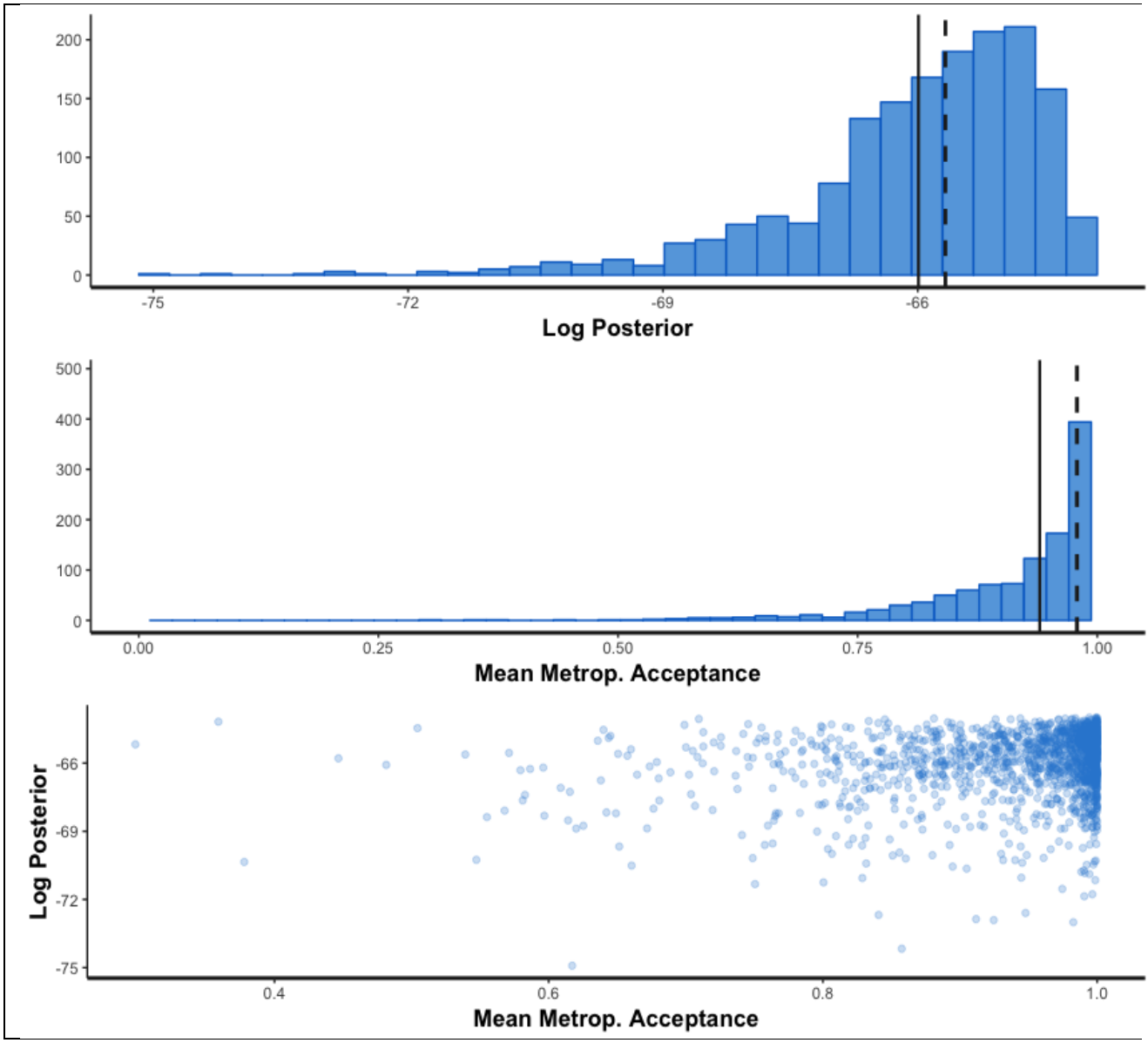
```
bvl_mcmcDiag(model, “a_Burden”)
bvl_mcmcDiag(model, “b_Res”)
bvl_mcmcDiag(model, “b_Insured”)
```

Hình kiểm tra:





Nếu vẽ log posterior:

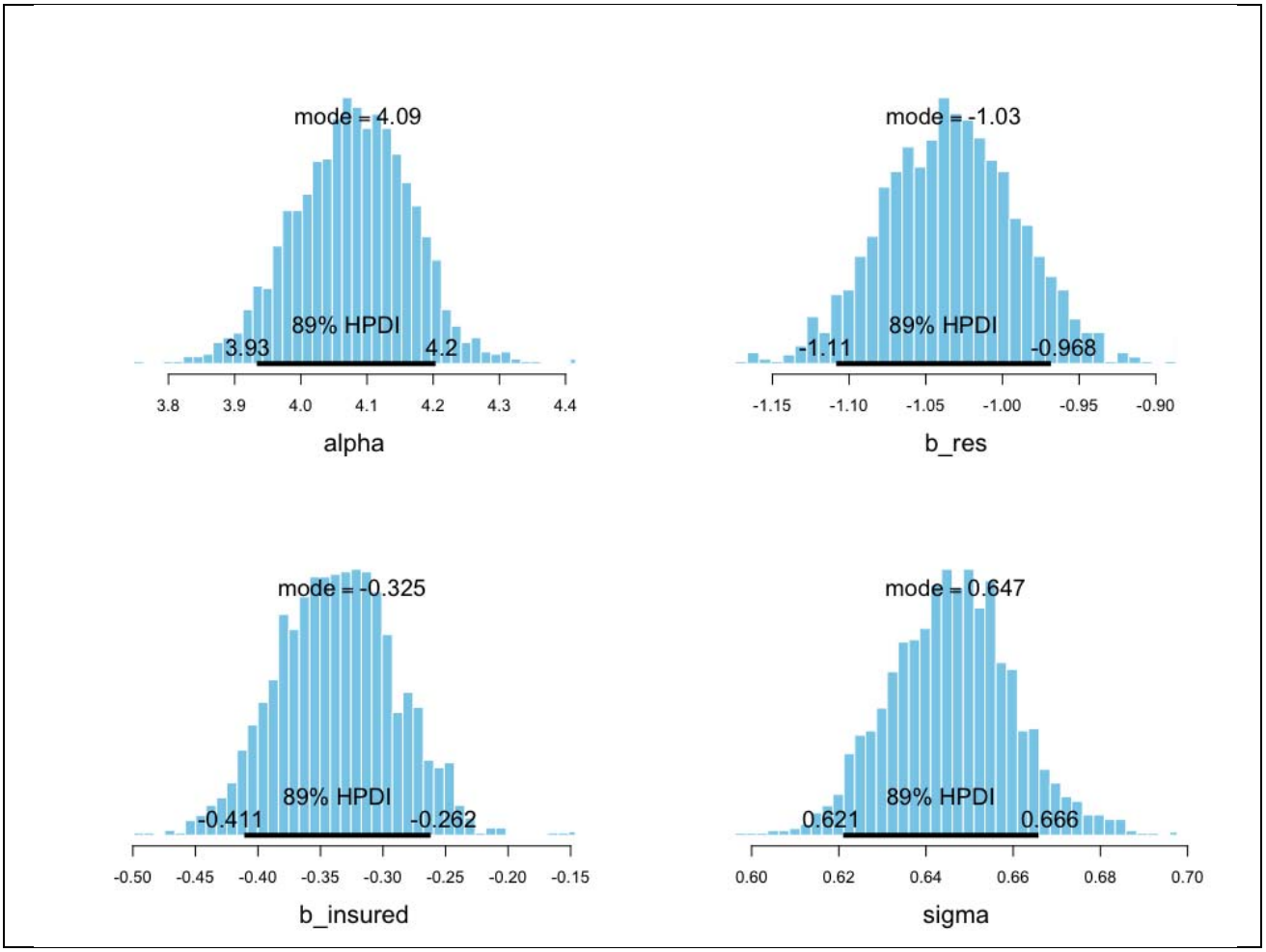


e. Kết quả hồi quy

Phân phối các hệ số:

```
bvl_plotParams(fit, 2, 2, 0.89)
```

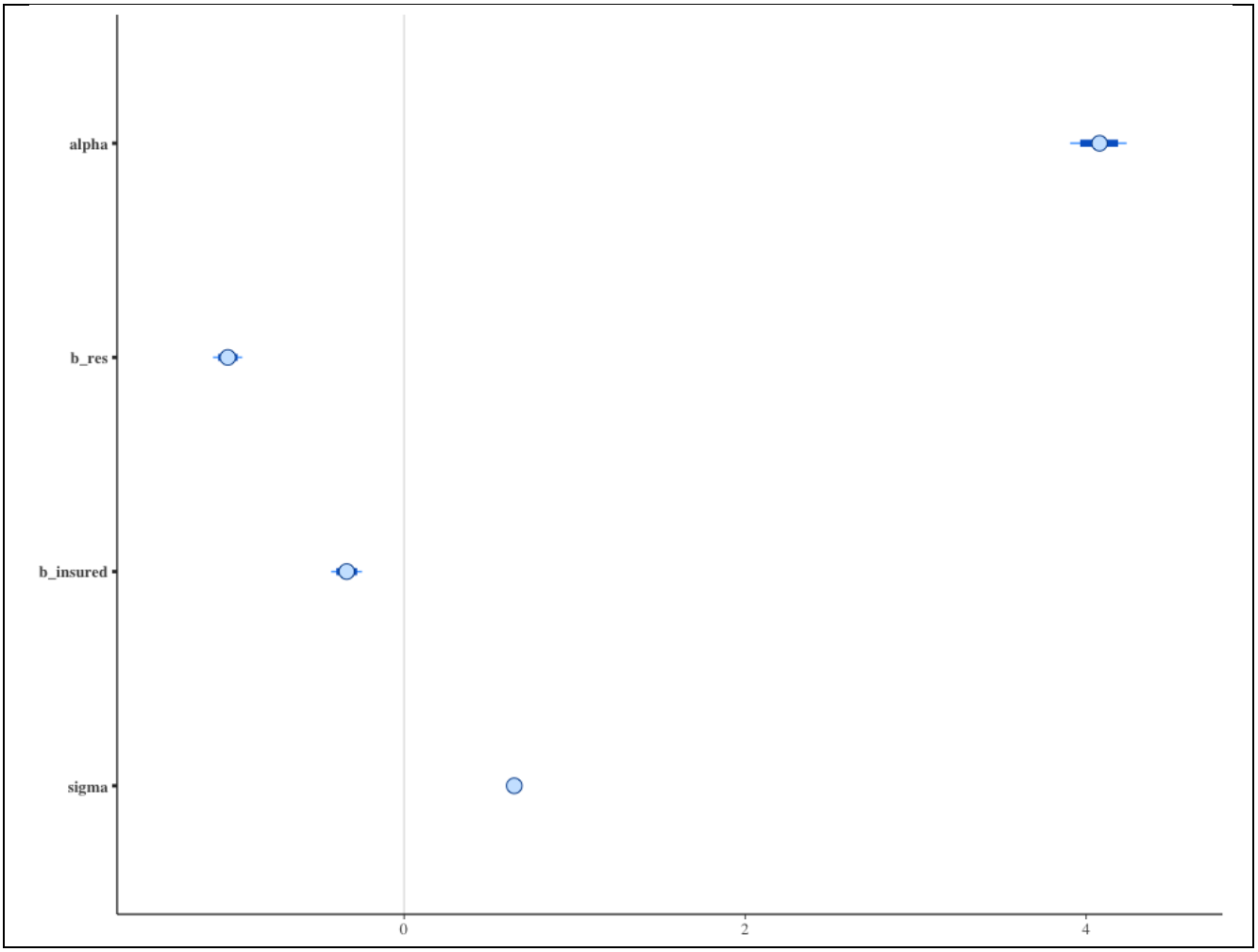
Histogram của các hệ số mô phỏng, kèm theo giá trị mode:



Nếu vẽ interval phân phối với mức đánh giá 80%:

```
bvl_plotIntervals(fit, prob = 0.8, prob_outer = 0.95, color_scheme = "brightblue")
```

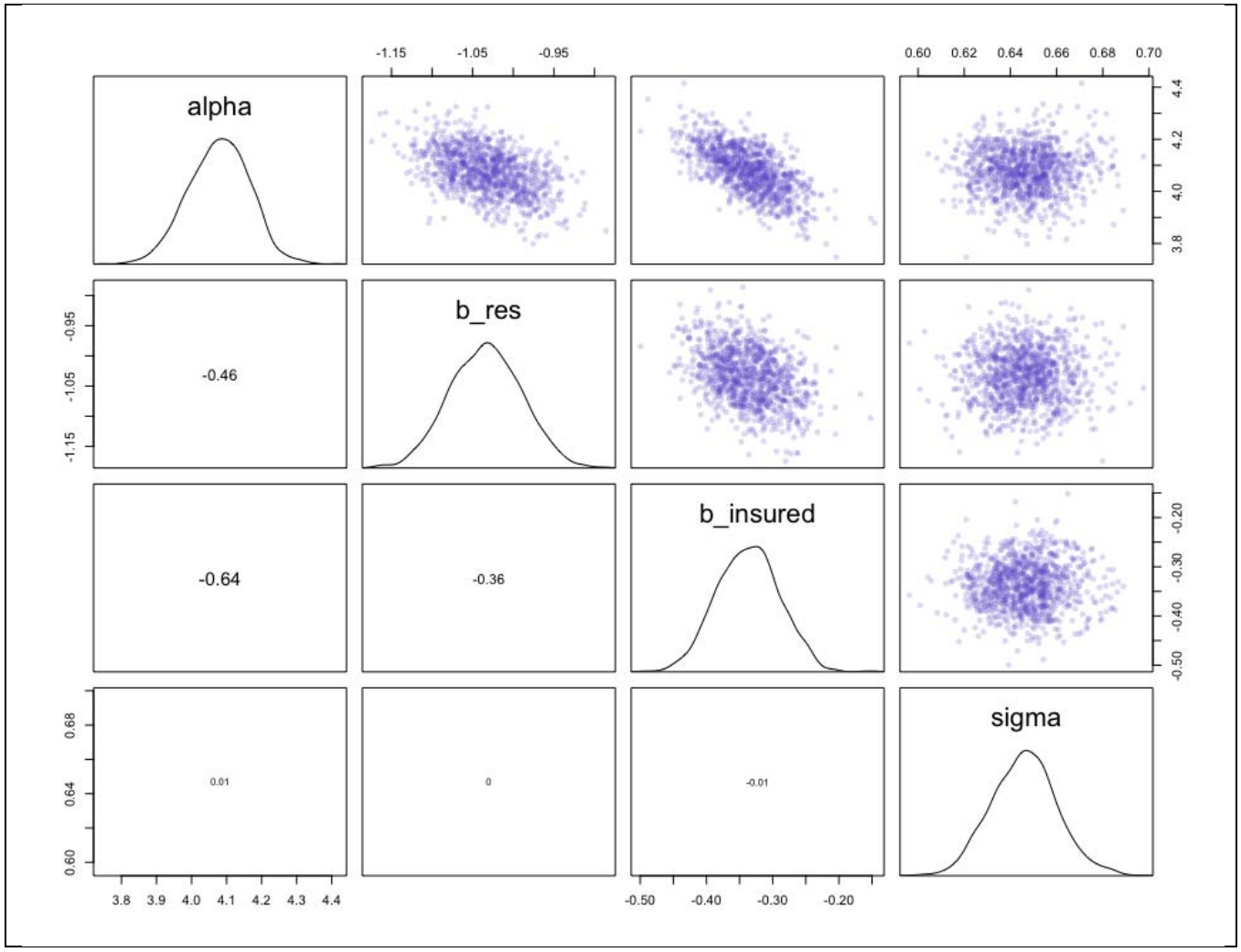
Các hệ số mô phỏng MCMC của mô hình tính toán nhờ Stan trong hình sau:



Tương ứng các cặp hệ số:

```
bvl_plotPair(fit, cex=1)
```

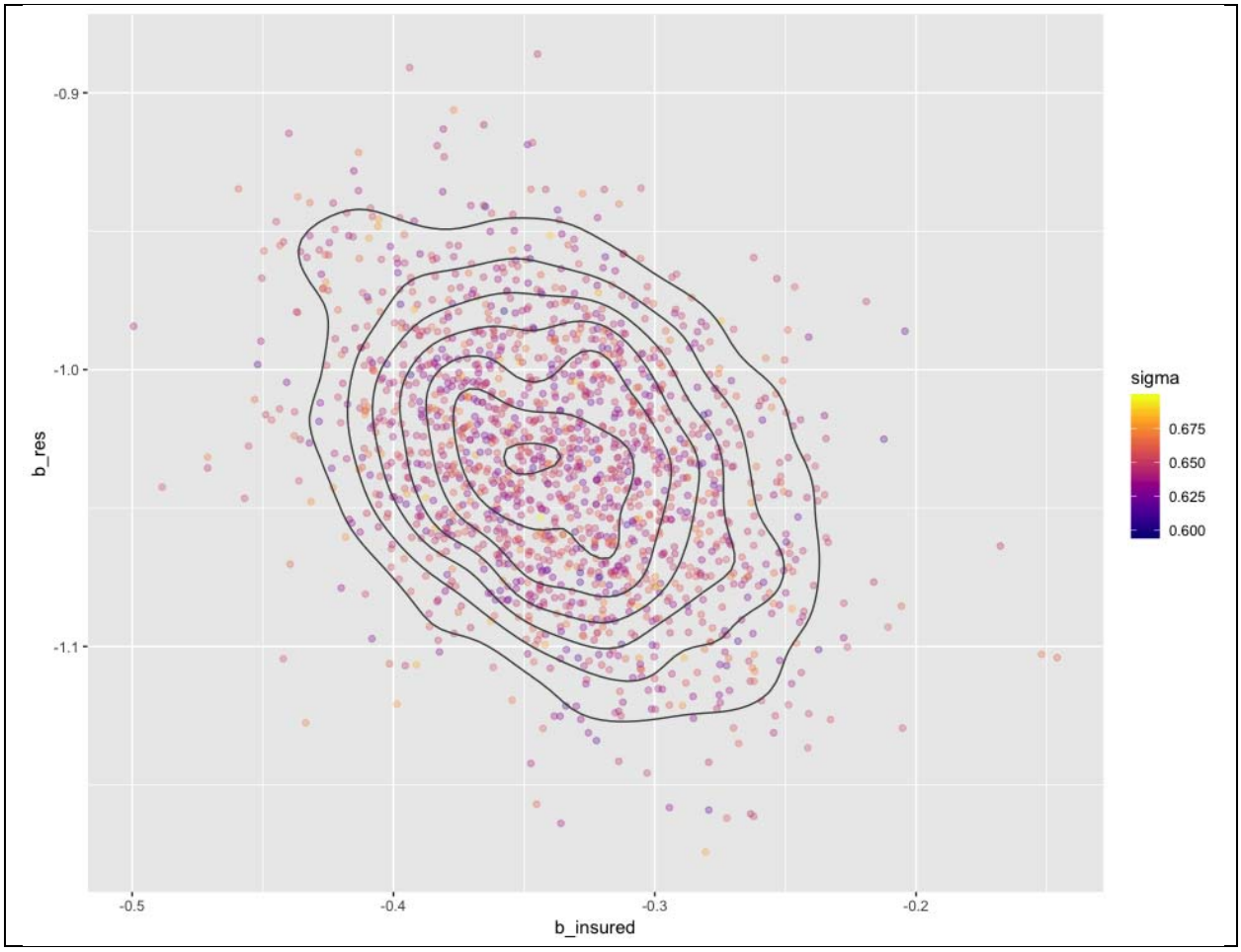
Hình ảnh thu được như sau:



Nếu so sánh riêng 2 tham số b_Res, b_Insured:

```
ggplot(model@posterior, aes(x=b_Insured_Burden, y=b_Res_Burden, color =
sigma_Burden))+
        geom_point(alpha = 0.3)+
        geom_density2d(color = "gray30")+
        scale_color_viridis(option = "C")+
        geom_abline(intercept=0,slope=1)
```

Hình vẽ thu được từ dữ liệu sau mô phỏng:



Ta thấy hệ số nghiêng nhiều về Res. Ảnh hưởng Res lên burden mạnh hơn nhiều so với insured. Tuy nhiên 2 hệ số này cùng dấu (-) ảnh hưởng cùng xu hướng. Cần quan sát đánh giá xung quanh các tọa độ tạo ra mật độ trù mật các điểm mô phỏng.

Khôi phục lại phương trình hồi quy ta sẽ có:

$$\text{Burden} = 4.08 - 1.03 * \text{Res} - 0.34 * \text{Insured}$$

Với bệnh nhân NonRes, UnInsured, mức áp lực kinh tế Financial Burden sẽ vào khoảng 2.71. Tức là xấp xỉ mức C.

Nếu visualize the hệ số hồi quy từ multiple estimates from the posterior cho từng hệ số b_Res và b_Insured như các hình Fig. 8a và Fig. 8b dưới đây.

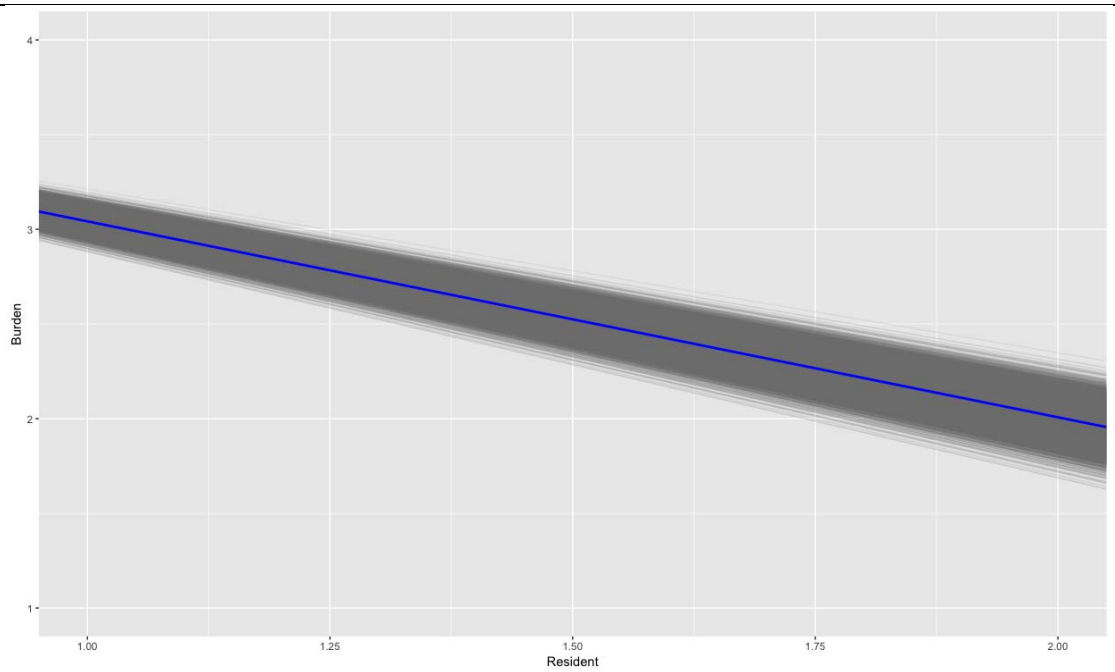


Fig. 8a

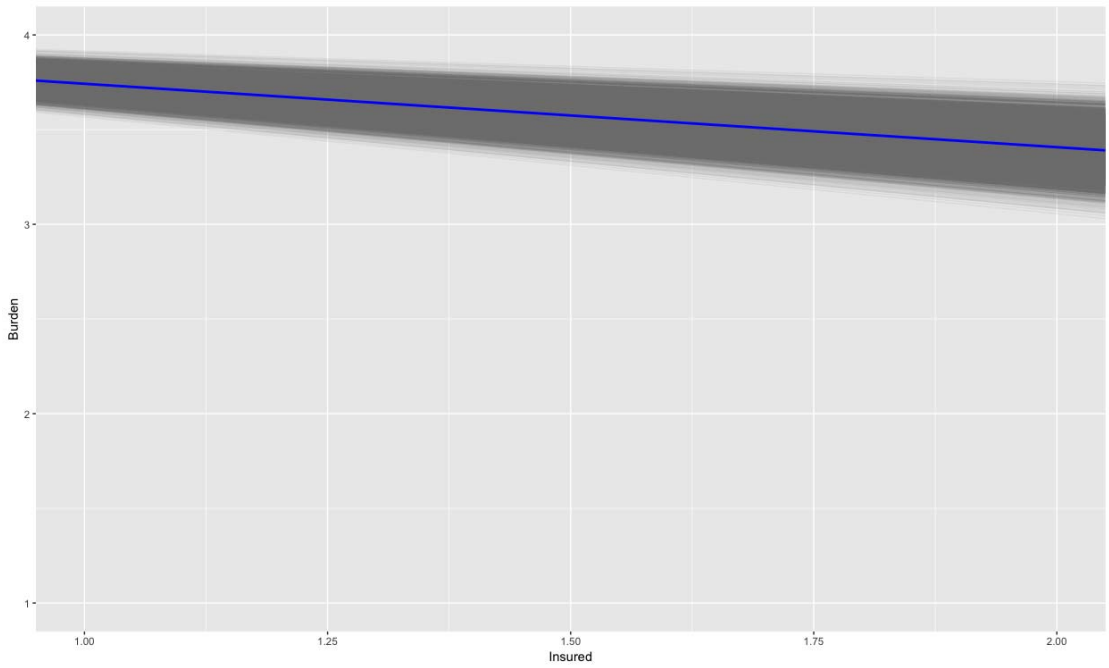


Fig. 8b

f. Posterior predictive checks

Để thực hiện Posterior predictive checks (PPC) ta cần cài sẵn code đánh giá mô hình khi thực hiện. Package bayesvl đã cài sẵn 1 đoạn code chuẩn để tính log posterior và y report, tuy nhiên cũng có thể cài thêm code đánh giá, ví dụ:

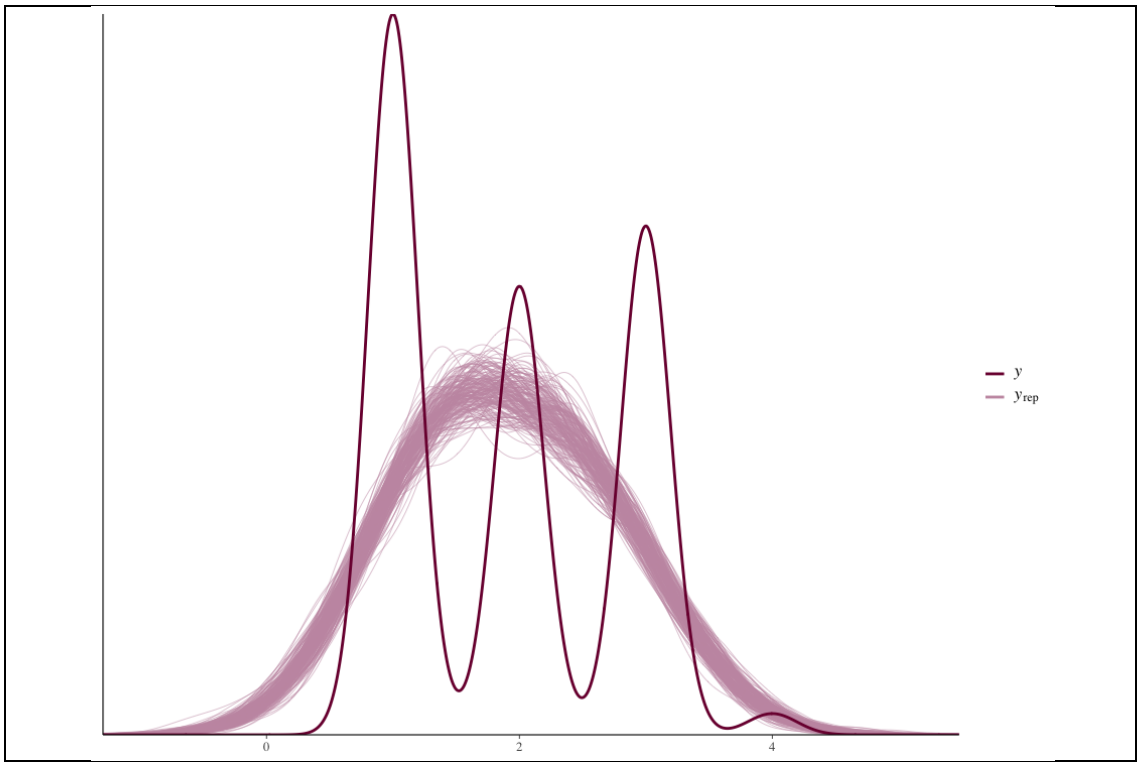
```
ppc_string <- “
  real log_lik_res1;

  vector[Nobs] y_rep_res1;
```

```
 vector[Nobs] y_rep_res2;

 log_lik_res1 = 0;

 for (i in 1:num_elements(y_rep_res1)) {
  y_rep_res1[i] = normal_rng(alpha + b_res * 1 + b_insured * insured[i], sigma);
 }
 for (i in 1:num_elements(y_rep_res2)) {
  y_rep_res2[i] = normal_rng(alpha + b_res * 2 + b_insured * insured[i], sigma);
 }

 for (i in 1: Nobs) {
        log_lik_res1 += normal_lpdf(y[i] | alpha + b_res * 1 + b_insured * insured[i], sigma);
 }"

# Fit the model

model <- bvl_modelFit(model, dat1042, warmup = 2000, iter = 5000, chains = 4, cores = 4,
ppc = ppc_string)
```

Kết quả y_rep sẽ cho ta thấy phân phối mật độ của dự báo mô hình dựa trên mô phỏng MCMC so với phân phối của dữ liệu đầu vào.

Trên histogram, ta thấy mean posterior của mẫu mô phỏng (đường kẻ xanh) được phân phối nằm khá sát với giá trị trung bình của dữ liệu chưa xử lý.

Nếu vẽ mật độ, ta có thể thấy phân phối có hình dạng như sau:



Phần **Note 2b** sẽ tiếp tục cùng phương pháp với mô hình phức tạp hơn, với yếu tố điều kiện kiểm soát "hierarchy".

References:

[**1**] Vuong QH, & La VP. (2019). BayesVL package for Bayesian statistical analyses in R. *Github*: BayesVL version 0.6.5; DOI: 10.31219/osf.io/ya9u6. Available from: <https://github.com/sshpa/bayesvl>.

[**2**] Vuong QH. (2015). Be rich or don't be sick: estimating Vietnamese patients' risk of falling into destitution. *SpringerPlus*, **4**(1), 529. DOI: 10.1186/s40064-015-1279-x.

[**3**] Ho MT, La VP, Nguyen MH, Vuong TT, Nghiem KCP, Tran T, Nguyen HKT, & Vuong QH. (2019). Health care, medical insurance, and economic destitution: A dataset of 1042 stories. *Data*, **4**, 57. DOI: 10.3390/data4020057.

[**4**] McElreath R. (2018). Statistical Rethinking: A Bayesian Course with Examples in R and Stan. London: Chapman and Hall/CRC.

[**5**] Scutari M. (2010). Learning Bayesian networks with the bnlearn R package. *Journal of Statistical Software*, **35**(3), 1-22.