

Bayesian Statistical Modeling Examples

Oliver Y. Hasegawa

Instruction

Thank you very much for tuning in.

Bellow are examples of Bayesian statistical modeling with mathematical equation and its Stan coding (There's also additional model using brms package in R).

Here we have variables called “**tightness**” and “**X1**” in our models.

1. Multivariate normal distribution model

1.1 Mathematical equation

$$\begin{pmatrix} \text{tightness}_i \\ X1_i \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu_{\text{tight}} \\ \mu_{X1} \end{pmatrix}, \begin{pmatrix} \sigma_{\text{tight}}^2 & \rho \sigma_{\text{tight}} \sigma_{X1} \\ \rho \sigma_{\text{tight}} \sigma_{X1} & \sigma_{X1}^2 \end{pmatrix}\right)$$

i represents the index of countries or states.

tightness and X_1 represent the variables which follow a bivariate normal distribution.

μ_{tight} and μ_{X1} respectively represent the means of tightness_i and X_{1i} .

σ_{tight}^2 and σ_{X1}^2 are the variances of tightness_i and X_{1i} .

ρ is the correlation coefficient between tightness_i and X_{1i} .

We assumed independence across i observations and estimated the parameters

μ_{tight} , μ_{X1} , σ_{tight}^2 , σ_{X1}^2 , and ρ .

1.2 Stan code

```
data {  
  int<lower=1> N;  
  array[N] vector[2] y;  
  
  vector[2] mu_prior_mean;
```

```

    vector<lower=0>[2] mu_prior_sd;
    vector<lower=0>[2] sigma_prior;
    real<lower=1> eta;
}

parameters {
    vector[2] mu;
    vector<lower=0>[2] sigma;
    cholesky_factor_corr[2] rho;
}

transformed parameters {
    matrix[2,2] L_Sigma = diag_pre_multiply(sigma, rho);
}

model {
    mu ~ normal(mu_prior_mean, mu_prior_sd);
    sigma ~ exponential(sigma_prior);
    rho ~ lkj_corr_cholesky(eta);

    y ~ multi_normal_cholesky(mu, L_Sigma);
}

generated quantities {
    array[N] vector[2] y_rep;
    for (n in 1:N) {
        y_rep[n] = multi_normal_cholesky_rng(mu, L_Sigma);
    }
}

```

2. Bayesian linear regression model

2.1 Mathematical equation

$$\begin{aligned}
 \text{tightness}_i &\sim \text{Normal}(\mu_i, \sigma) \\
 \mu_i &= \beta_0 + \beta_1 X1_i \\
 \beta_0 &\sim \text{Normal}(\mu_{\beta_0}, \tau_{\beta_0}) \\
 \beta_1 &\sim \text{Normal}(\mu_{\beta_1}, \tau_{\beta_1}) \\
 \sigma &\sim \text{Exponential}(\lambda)
 \end{aligned}$$

$tightness_i$ is predicted by X_{1i} , and μ_i is the expected value of $tightness_i$. While β_0 represents the intercept, β_1 represents the slope coefficient. σ is the residual standard deviation.

It is assumed that β_0 and β_1 are sampled from normal distributions whose means and standard deviations are μ_β and τ_β , respectively.

The values of X_1 , β_0 , β_1 , μ_β , τ_β , and σ were estimated as parameters.

2.2 Stan code

```
data {
  int<lower=1> N;
  vector[N] X1;
  vector[N] tightness;
  int<lower=0, upper=1> prior_only;

  real mu_alpha;
  real<lower=0> tau_alpha;
  real mu_beta;
  real<lower=0> tau_beta;
  real<lower=0> lambda;
}

parameters {
  real alpha;
  real beta;
  real<lower=0> sigma;
}

model {
  vector[N] mu;
  mu = alpha + beta*X1;

  alpha ~ normal(mu_alpha, tau_alpha);
  beta ~ normal(mu_beta, tau_beta);
  sigma ~ exponential(lambda);

  if (prior_only == 0) {
    tightness ~ normal(mu, sigma);
  }
}
```

```

generated quantities {
  vector[N] y_ppc;
  for (n in 1:N) {
    y_ppc[n] = normal_rng(alpha + beta * X1[n], sigma);
  }
}

```

3. Bayesian Hierarchical Random Intercept model

3.1 Mathematical equation

$$\begin{aligned}
\text{tightness}_i &\sim \text{Normal}(\mu_i, \sigma) \\
\mu_i &= \alpha_{\text{Region}[i]} + \beta_1 X1_i \\
\alpha_{\text{Region}} &\sim \text{Normal}(\mu_\alpha, \tau_\alpha) \\
\beta_1 &\sim \text{Normal}(\mu_{\beta_1}, \tau_{\beta_1}) \\
\mu_\alpha &\sim \text{Normal}(\mu_0, \tau_0) \\
\tau_\alpha &\sim \text{Exponential}(\lambda_\alpha) \\
\sigma &\sim \text{Exponential}(\lambda_\sigma)
\end{aligned}$$

tightness_i is predicted by $X1_i$, and μ_i is the expected value of tightness_i .

σ is the residual standard deviation.

$\alpha_{\text{Region}[i]}$ is the intercept specific to each $\text{Region}[i]$, and β_1 is the fixed-effect slope coefficient.

μ_α and τ_α are the hyperparameters of α_{Region} .

The values of $X1$, α_{Region} , β_1 , μ_α , τ_α , μ_{β_1} , τ_{β_1} , μ_0 , τ_0 , and σ were estimated as parameters.

μ_α , μ_{β_1} , and μ_0 were assigned normal distributions, while τ_α and σ were assigned exponential distributions.

3.2 Stan code

```

data {
  int<lower=1> N;
  int<lower=1> R;
  array[N] int<lower=1, upper=R> region;
  vector[N] X1;
  vector[N] tightness;
  int<lower=0, upper=1> prior_only;

  real mu_alpha_bar;

```

```

    real tau_alpha_bar;
    real mu_beta;
    real tau_beta;
    real<lower=0> lambda_alpha;
    real<lower=0> lambda_sigma;
}

parameters {
    real alpha_bar;
    real<lower=0> alpha_sigma;
    vector[R] alpha_raw;
    real beta;
    real<lower=0> sigma;
}

transformed parameters {
    vector[R] alpha;
    alpha = alpha_bar + alpha_sigma * alpha_raw;
}

model {
    beta ~ normal(mu_beta, tau_beta);
    alpha_bar ~ normal(mu_alpha_bar, tau_alpha_bar);
    alpha_sigma ~ exponential(lambda_alpha);
    alpha_raw ~ normal(0, 1);
    sigma ~ exponential(lambda_sigma);

    if (prior_only == 0) {
        tightness ~ normal(alpha[region] + beta * X1, sigma);
    }
}

generated quantities {
    vector[N] y_ppc;
    for (i in 1:N) {
        y_ppc[i] = normal_rng(alpha[region[i]] + beta * X1[i], sigma);
    }
}

```

4. Bayesian Hierarchical Random Coefficient Model with Correlation Matrix

4.1 Mathematical equation

$$\begin{aligned}
 \text{tightness}_i &\sim \text{Normal}(\mu_i, \sigma) \\
 \mu_i &= \alpha_{\text{Region}[i]} + \beta_{\text{Region}[i]} X_{1i} \\
 \begin{bmatrix} \alpha_{\text{Region}} \\ \beta_{\text{Region}} \end{bmatrix} &\sim \text{MVNormal} \left(\begin{bmatrix} \alpha_0 \\ \beta_0 \end{bmatrix}, \mathbf{S} \right) \\
 \mathbf{S} &= \begin{pmatrix} \sigma_\alpha & 0 \\ 0 & \sigma_\beta \end{pmatrix} \mathbf{R} \begin{pmatrix} \sigma_\alpha & 0 \\ 0 & \sigma_\beta \end{pmatrix} \\
 \alpha_0 &\sim \text{Normal}(\mu_{\alpha_0}, \tau_{\alpha_0}) \\
 \beta_0 &\sim \text{Normal}(\mu_{\beta_0}, \tau_{\beta_0}) \\
 \sigma_\alpha &\sim \text{Exponential}(\lambda_\alpha) \\
 \sigma_\beta &\sim \text{Exponential}(\lambda_\beta) \\
 \mathbf{R} &\sim \text{LKJcorr}(\eta) \\
 \sigma &\sim \text{Exponential}(\lambda_\sigma)
 \end{aligned}$$

$\alpha_{\text{Region}[i]}$ is the region-specific intercept, and $\beta_{\text{Region}[i]}$ is the region-specific slope for predictor X_{1i} .

μ_i is the expected value of tightness_i , and σ is the residual standard deviation.

We modeled the random effects α_{Region} and β_{Region} jointly as draws from a bivariate normal distribution.

The covariance matrix \mathbf{S} was decomposed into standard deviations and a correlation matrix \mathbf{R} . $\mathbf{R} \sim \text{LKJcorr}(\eta)$ represents the correlation matrix between the region-specific intercepts and slopes,

while σ_α and σ_β are their respective standard deviations.

The values of X_1 , α_{Region} , β_{Region} , α_0 , β_0 , μ_a , τ_a , μ_β , τ_β , μ_0 , τ_0 , and σ were estimated as parameters.

α_0 and β_0 were assigned normal distributions.

σ_α , σ_β , and σ were assigned exponential distributions.

4.2 Stan code

```

data {
  int<lower=1> N;

```

```

int<lower=1> R;
array[N] int<lower=1, upper=R> region;
vector[N] X1;
vector[N] tightness;
int<lower=0,upper=1> prior_only;

real mu_alpha;
real tau_alpha;
real mu_beta;
real tau_beta;
real<lower=0> lambda_alpha;
real<lower=0> lambda_beta;
real<lower=0> lambda_sigma;
}

parameters {
  real a;
  real b;
  real<lower=0> sigma;

  vector<lower=0>[2] sigma_region;
  cholesky_factor_corr[2] L_Rho;

  matrix[2, R] z;
}

transformed parameters {
  matrix[2, R] dev = diag_pre_multiply(sigma_region, L_Rho) * z;

  vector[R] a_region = rep_vector(a, R) + dev[1]';
  vector[R] b_region = rep_vector(b, R) + dev[2]';
}

model {
  vector[N] mu;
  a ~ normal(mu_alpha, tau_alpha);
  b ~ normal(mu_beta, tau_beta);
  sigma ~ exponential(lambda_sigma);
  sigma_region[1] ~ exponential(lambda_alpha);
  sigma_region[2] ~ exponential(lambda_beta);
  L_Rho ~ lkj_corr_cholesky(4);
  to_vector(z) ~ normal(0, 1);
}

```

```

for (n in 1:N) {
  mu[n] = a_region[region[n]] + b_region[region[n]] * X1[n];
}
if (prior_only == 0) {
  tightness ~ normal(mu, sigma);
}
}

generated quantities {
  corr_matrix[2] Rho_out = multiply_lower_tri_self_transpose(L_Rho);

  vector[N] y_ppc;
  for (n in 1:N)
    y_ppc[n] = normal_rng(a_region[region[n]] + b_region[region[n]] * X1[n], sigma);
}

```

5. Gaussian Process Model with Phylogenetic distance RBF kernel

5.1 Mathematical equation

$$\begin{aligned}
\text{tightness}_i &\sim \text{Normal}(\mu_i, \sigma) \\
\mu_i &= \alpha + \gamma_{\text{Phylogeny}[i]} + \beta X_{1i} \\
\gamma &\sim \text{MVNormal}(\mathbf{0}, \mathbf{K}) \\
K_{ij} &= \eta^2 \exp(-\rho^2 D_{ij}^2) + \delta_{ij} \epsilon \\
\alpha &\sim \text{Normal}(\mu_\alpha, \tau_\alpha) \\
\beta &\sim \text{Normal}(\mu_\beta, \tau_\beta) \\
\eta &\sim \text{Exponential}(\lambda_\eta) \\
\rho &\sim \text{Exponential}(\lambda_\rho) \\
\sigma &\sim \text{Exponential}(\lambda_\sigma)
\end{aligned}$$

tightness_i is the observed outcome for the national unit, and X_{1i} is the predictor variable. α is the intercept, and β is the fixed-effect slope.

Importantly, $\gamma_{\text{Phylogeny}[i]}$ is a structured random effect drawn from a Gaussian Process, and σ is the residual standard deviation.

The vector of structured effects γ is modeled as a multivariate normal distribution.

The covariance matrix \mathbf{K} is constructed using a Radial Basis Function (RBF) kernel, which defines the pairwise similarity between units i and j based on their distance:

$$K_{ij} = \eta^2 \exp(-\rho^2 D_{ij}^2) + \delta_{ij} \epsilon$$

In this kernel, η is the marginal standard deviation of the Gaussian Process, and ρ is the length-scale parameter controlling how correlation decays with increasing or decreasing distance. D_{ij} represents the distance between units i and j , and ϵ is a small positive constant added to the diagonal through δ_{ij} , the Kronecker delta (jitter), to improve numerical stability.

The values of X_1 , α , β , η , ρ , and σ were estimated as parameters.

The parameters α and β were assigned normal distributions, while the hyperparameters η , ρ , and the residual standard deviation σ were assigned exponential distributions.

5.2 Stan code

```
data { int<lower=1> N;
vector[N] tightness;
vector[N] X1;
matrix[N, N] D;

real mu_alpha;
real tau_alpha;
  real mu_beta;
real tau_beta;
  real<lower=0> s_eta;
real<lower=0> s_rho;
real<lower=0> lambda_sigma;
int<lower=0, upper=1> prior_only;
}

transformed data {
  real delta = 1e-8;
}

parameters {
  real alpha;
  real beta;
  real<lower=0> eta;
  real<lower=0> rho;
  real<lower=0> sigma;
  vector[N] gamma;
}
```

```

model {
  matrix[N, N] K;
  for (i in 1:N) {
    K[i, i] = square(eta) + delta;
    for (j in (i+1):N) {
      real d = D[i, j];
      K[i, j] = square(eta) * exp( - square(rho) * square(d) );
      K[j, i] = K[i, j]; }
    }

  gamma ~ multi_normal(rep_vector(0, N), K);
  alpha ~ normal(mu_alpha, tau_alpha);
  beta ~ normal(mu_beta, tau_beta);
  eta ~ normal(0, s_eta);
  rho ~ normal(0, s_rho);
  sigma ~ exponential(lambda_sigma);
  vector[N] mu;
  for (i in 1:N) {
    mu[i] = alpha + beta * X1[i] + gamma[i]; }

    if (prior_only == 0) {
      tightness ~ normal(mu, sigma);
    }
  }

generated quantities {
  vector[N] y_rep; for (i in 1:N) {
    y_rep[i] = normal_rng(alpha + gamma[i] + beta * X1[i], sigma);
  }
}

```

Additional: Hierarchical Bayesian Linear Model with Random-Effects Covariance Structure

In completely different analysis I have used similar but different model with brms package in R instead of the Stan code.

Here we have contact (interpersonal contact with other individual with lesbian, gay, and bisexual person; LGB) and law enforcement (of same-sex marriage).

Mathematical equation

$$\begin{aligned}y_i &\sim \text{Normal}(\mu_i, \sigma) \\ \mu_i &= \alpha + \beta_1 \text{Contact}_i + \beta_2 \text{Law}_i + u_{j[i]}^{\text{geo}} + u_{k[i]}^{\text{phy}} \\ \mathbf{u}^{\text{geo}} &\sim \text{MVN}(\mathbf{0}, \mathbf{G}) \\ \mathbf{u}^{\text{phy}} &\sim \text{MVN}(\mathbf{0}, \mathbf{S}) \\ \alpha &\sim \text{Normal}(0, \delta_\alpha) \\ \beta_1 &\sim \text{Normal}(0, \delta_{\beta_1}) \\ \beta_2 &\sim \text{Normal}(0, \delta_{\beta_2}) \\ \delta_{\beta_1}, \delta_{\beta_2} &\sim \text{Exponential}(\lambda_\beta) \\ \sigma &\sim \text{Exponential}(\lambda_\sigma)\end{aligned}$$

y_i represents the outcome variable (e.g., justifiability of LGB) for individual i , and Contact_i and Law_i are the individual- and country-level predictors, respectively. α is the overall intercept, while β_1 and β_2 represent the fixed-effect slopes for Contact_i and Law_i .

Importantly, $u_{j[i]}^{\text{geo}}$ and $u_{k[i]}^{\text{phy}}$ are structured random effects capturing country-level dependencies based on geographic and phylogenetic proximity, respectively.

Each random-effect vector, \mathbf{u}^{geo} and \mathbf{u}^{phy} , is modeled as a multivariate normal distribution with zero mean and covariance matrices Σ_G and Σ_S .

These covariance matrices represent the spatial correlation structure across countries, derived from geographic distance and cultural/phylogenetic similarity matrices.

brms code

```
model <- brm(
  formula = outcome ~ contact + law +
    (1 | gr(country_geo, cov = G, id = "geo")) +
    (1 | gr(country_phy, cov = S, id = "phy")) ,
  data = data,
  data2 = list(G = G_pd, S = S_pd),
  family = gaussian(),
  prior = c(
    prior(normal(0,1), class = Intercept),
    prior(normal(0,1), class = b),
```

```
    prior(exponential(2), class = sd),  
    prior(exponential(1), class = sigma)),  
chains = 4, cores = 4, iter = 5000, warmup = 1000  
)
```