Bayesian Multi-level Regression

Models for Socio-Environmental Data

Chris Che-Castaldo, Mary B. Collins, N. Thompson Hobbs

December 17, 2020

Lecture material

- Background
- Bayesian, multilevel models for grouped data
  - group level intercepts
  - group level intercepts with group level covariate
  - group level slopes and intercepts
  - an essential coding trick
  - prediction across groups
- Priors on group level variances: See lecture "More about priors 2."

Recall that

$$\underbrace{[\boldsymbol{\theta}, \sigma^2 | y_i]}_{\text{posterior}} \propto \overbrace{[y_i, \boldsymbol{\theta}, \sigma^2]}^{\text{joint}}$$

$$\underbrace{[\boldsymbol{\theta}, \sigma^2 | y_i]}_{\text{posterior}} = c\overbrace{[y_i, \boldsymbol{\theta}, \sigma^2]}^{\text{joint}}$$

MCMC allows us to discover the c.

The simple, Bayesian set-up

Deterministic model:

$$g(\boldsymbol{\theta}, x_i)$$

Stochastic model:

$$\underbrace{[\boldsymbol{\theta}, \sigma^2 | y_i]}_{\text{posterior}} \propto \overbrace{\underbrace{[y_i | g(\boldsymbol{\theta}, x_i), \sigma^2]}_{\text{likelihood}} \underbrace{[\boldsymbol{\theta}][\sigma^2]}_{\text{priors}}}^{\text{joint}}$$

The factored, joint distribution (aka joint conditional) provides a detailed blueprint for

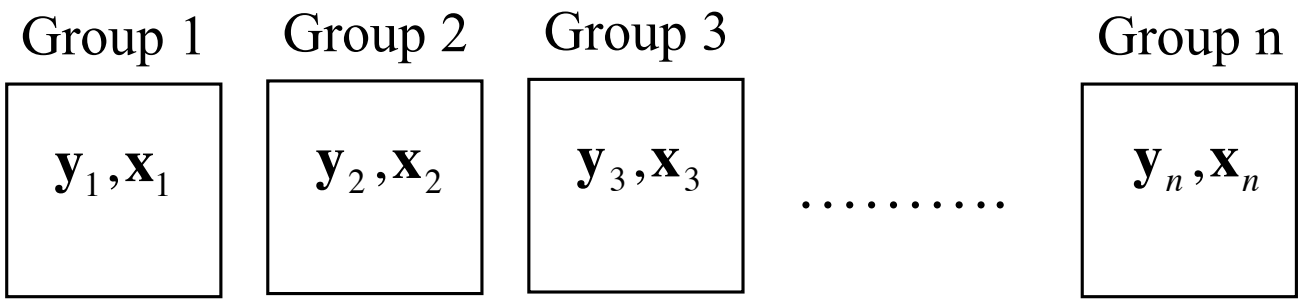
1. Writing the full conditionals as the basis for Gibbs (for conjugate full conditionals) or Metropolis-Hastings sampling (for any full conditional).
2. Writing JAGS code.

Hierarchical models: "modeling parameters"

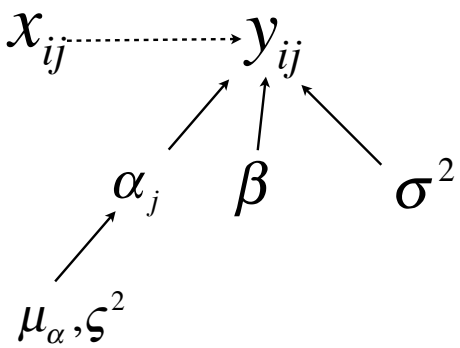
$$
\begin{aligned}
[\theta_1, \boldsymbol{\theta}_2, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2 \mid y_{ij}] &\propto [\theta_1, \boldsymbol{\theta}_2, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2, y_{ij}] \\
[\theta_1, \boldsymbol{\theta}_2, \boldsymbol{\alpha}, \boldsymbol{\sigma}^2 \mid y_{ij}] &\propto [y_{ij} | g(\theta_1, \theta_{2,j}, x_{ij}), \sigma_1^2] \\
&\times [\theta_{2,j} | h(\alpha_1, \alpha_2, u_j), \sigma_2^2] \\
&\times [\theta_1], [\alpha_1], [\alpha_2][\sigma_1^2][\sigma_2^2]
\end{aligned}
$$

Draw the DAG.

The problem

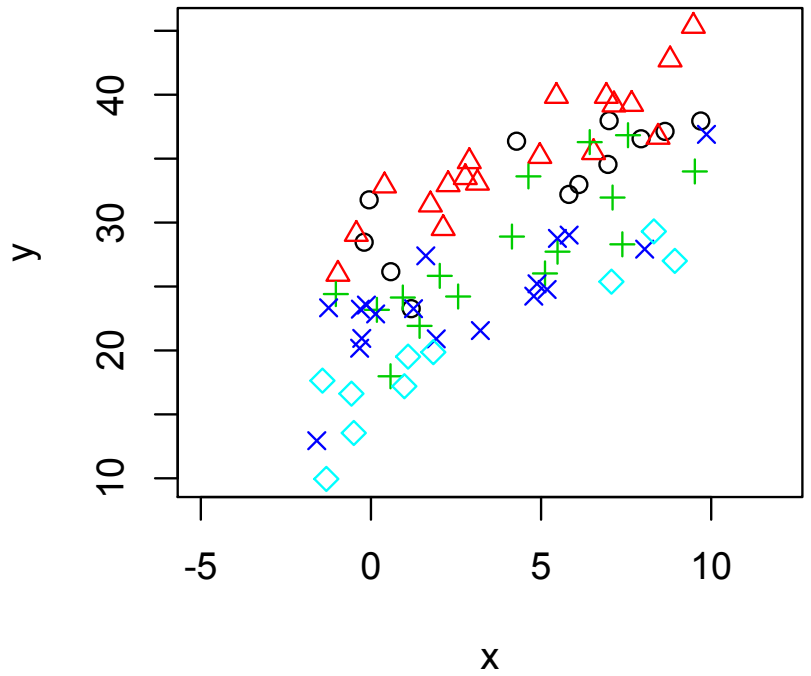

Group 1
Group 2
Group 3
Group n
$\mathbf{y}_1, \mathbf{x}_1$
$\mathbf{y}_2, \mathbf{x}_2$
$\mathbf{y}_3, \mathbf{x}_3$
.....
$\mathbf{y}_n, \mathbf{x}_n$

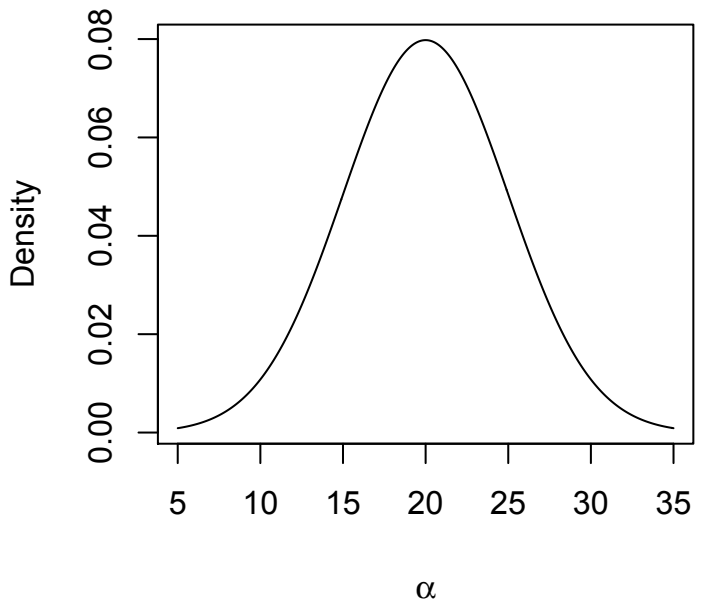
We can model the intercept (or slope):



$$[\beta, \boldsymbol{\alpha}, \sigma^2, \mu_\alpha, \varsigma^2, |\mathbf{y}] \propto \prod_{i=1}^{n_j} \prod_{j=1}^{J} \text{normal}\left(y_{ij} | \alpha_j + \beta x_{ij}, \sigma^2\right)$$
$$\times \text{normal}\left(\alpha_j | \mu_\alpha, \varsigma^2\right)$$
$$\times \text{normal}\left(\beta | 0, 10000\right) \text{normal}\left(\mu_\alpha | 0, 1000\right)$$
$$\times \text{inverse gamma}\left(\sigma^2 | .001, .001\right) \text{inverse gamma}\left(\varsigma^2 | .001, .001\right)$$

We seek to understand the distribution of intercepts.





Some notation

$$\mu_{ij} = \beta_0 + \beta_1 x_{ij} + \alpha_j$$
$$y_{ij} \sim \text{normal}(\mu_{ij}, \sigma^2)$$
$$\alpha_j \sim \text{normal}(0, \varsigma^2)$$

is identical to:

$$\mu_{ij} = \alpha_j + \beta_1 x_{ij}$$
$$y_{ij} \sim \text{normal}(\mu_{ij}, \sigma^2)$$
$$\alpha_j \sim (\mu_\alpha, \varsigma^2)$$

Some notation

σ variance sigma

$$\mu_{ij} = \beta_0 + \beta_1 x_{ij} + \epsilon_j$$

$$\epsilon_j \sim \text{normal}(0, \sigma^2)$$

$$y_{ij} \sim \text{normal}(\mu_{ij}, \sigma^2)$$

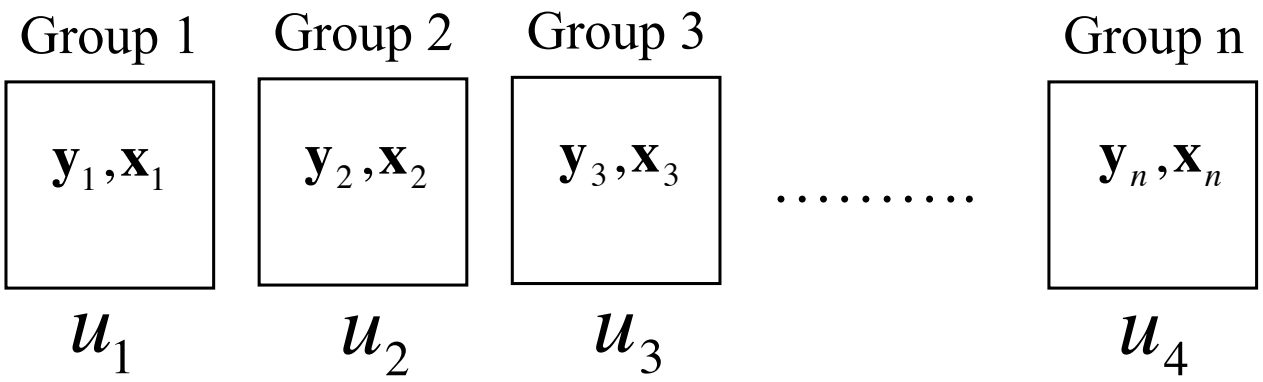
is the same as:

$$\mu_{ij} = \alpha_j + \beta_1 x_{ij}$$

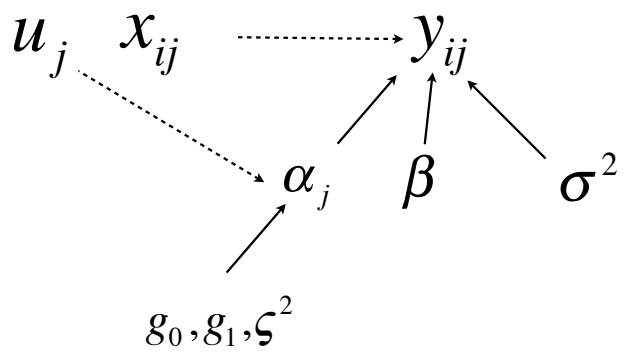
$$\alpha_j \sim \text{normal}(\mu_\alpha, \sigma^2)$$

$$y_{ij} \sim \text{normal}(\mu_{ij}, \sigma^2)$$

Include data on groups.

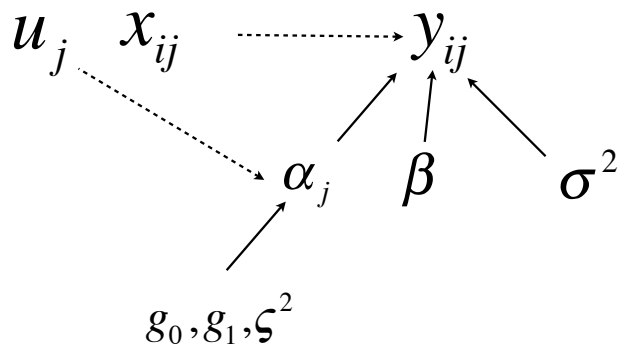


We can model the intercept (or slope) as a function of group level data:



$$[\boldsymbol{\alpha}, \beta, \sigma^2, \mathrm{g}, \varsigma^2, |\mathrm{y}] \propto \prod_{i=1}^{n_j} \prod_{j=1}^{J} \mathrm{normal}\left(y_{ij} | \alpha_j + \beta x_{ij}, \sigma^2\right)$$
$$\times \mathrm{normal}\left(\alpha_j | g_0 + g_1 u_j, \varsigma^2\right)$$
$$\times \mathrm{normal}\left(\beta | 0, .001\right) \mathrm{normal}\left(g_0 | 0, 1000\right) \mathrm{normal}\left(g_1 | 0, 1000\right)$$
$$\times \mathrm{inverse\,gamma}\left(\sigma^2 | .001, .001\right) \mathrm{inverse\,gamma}\left(\varsigma^2 | .001, .001\right)$$

An essential coding trick: Indexing groups



$$[\boldsymbol{\alpha}, \beta, , \sigma^2, \mathbf{g}, \varsigma^2, |\mathbf{y}] \propto \prod_{i=1}^{n_j} \prod_{j=1}^{J} \text{normal}\left(y_{ij} | \alpha_j + \beta x_{ij}, \sigma^2\right)$$
$$\times \text{normal}\left(\alpha_j | g_0 + g_1 u_j, \varsigma^2\right)$$
$$\times \text{normal}\left(\beta | 0, .001\right) \text{normal}\left(g_0 | 0, 1000\right) \times \text{normal}\left(g_1 | 0, 1000\right)$$
$$\times \text{inverse gamma}\left(\sigma^2 | .001, .001\right) \text{inverse gamma}\left(\varsigma^2 | .001, .001\right)$$

Indexing groups

```
> u
[1]  6.215579  8.716296 10.064460 11.292387 14.504154 14.734861
[7] 18.356877 18.910133
```

```
> head(y[,1:4])
     group i        x[i]     y[i]
[1,]     1 1 -0.00266051 13.48934
[2,]     1 2  4.54802848 22.29538
[3,]     1 3  9.86832462 29.03655
[4,]     1 4  0.99869789 18.61136
[5,]     1 5  1.27733200 20.59178
[6,]     1 6  4.32915675 25.37082
> tail(y[,1:4])
       group   i     x[i]     y[i]
[108,]     8 108 4.543959 38.93163
[109,]     8 109 1.287844 34.65796
[110,]     8 110 6.642313 40.62259
[111,]     8 111 7.404183 40.46518
[112,]     8 112 8.252571 41.47995
[113,]     8 113 9.558780 46.14771
```

Indexing groups

```
model{
beta ~ dnorm(0,.0001)
sigma ~ dunif(0,50)
tau.p <- 1/sigma^2
g0 ~ dnorm(0,.0001)
g1 ~ dnorm(0,.0001)
varsigma ~ dunif(0,50)
tau.g <- 1/varsigma^2
 for (i in 1:length(y)){
  mu[i] <- alpha[group[i]]+ beta*x[i]
  y[i] ~ dnorm(mu[i],tau.p)
  }
  for(j in 1:n.group){
  mu.g[j] <- g0 + g1*u[j]
  alpha[j]~dnorm(mu.g[j],tau.g)
    }
```

Indexing groups

```
model{
beta ~ dnorm(0,.0001)
sigma ~ dunif(0,50)
tau.p <- 1/sigma^2
g0 ~ dnorm(0,.0001)
g1 ~ dnorm(0,.0001)
varsigma ~ dunif(0,50)
tau.g <- 1/varsigma^2
 for (i in 1:length(y)){
  mu[i] <- alpha[group[i]]+ beta*x[i]
  y[i] ~ dnorm(mu[i],tau.p)
  }
  for(j in 1:n.group){
  mu.g[j] <- g0 + g1*u[j]
  alpha[j]~dnorm(mu.g[j],tau.g)
    }
```

Modeling intercepts and slopes

A correlation matrix:

Correlations

	Weight in kg	Hours of Sleep	Exposure while Sleeping	Life Span
Weight in kg	1	-.307	.338	.302
Hours of Sleep	-.307	1	-.642	-.410
Exposure while Sleeping	.338	-.642	1	.360
Life Span	.302	-.410	.360	1

[1]

Let i index rows and j index columns. Recall that the correlation between two random variables is simply their covariance divided by the standard deviation of both variables, $\rho_{ij} = \frac{\text{cov}_{ij}}{\sigma_i \sigma_j}$. It is the *standardized* covariance. Standardization means it can take on values between -1 and 1.

[1] http://www.theanalysisfactor.com/covariance-matrices/

Modeling intercepts and slopes

Let i index rows and j index columns. If we multiply this correlation matrix times $\sigma_i \sigma_j$ we obtain a *covariance* matrix:

Covariances

	body weight in kg	total sleep (hours/day	sleep exposure index (1-5)	maximum life span (years)
body weight in kg	808485.128	-1313.960	488.116	5113.271
total sleep (hours/day)	-1313.960	21.222	-4.544	-36.297
sleep exposure index (1-5)	488.116	-4.544	2.575	10.661
maximum life span (years)	5113.271	-36.297	10.661	331.468

Covariance ranges from $-\infty$ to $+\infty$.

Modeling intercepts and slopes

Imagine a vector of 3 random variables, $(z_i, z_2, z_3)'$ The covariance between any two of these random variables is simply an unstandardized version of the correlation between them: it is correlation measured in the units of the random variables. The covariance matrix (aka variance covariance matrix) of the random variable is:

$$\mathbf{\Sigma} = \begin{pmatrix} \sigma_1^2 & \text{Cov}_{1,2} & \text{Cov}_{1,3} \\ \text{Cov}_{2,1} & \sigma_2^2 & \text{Cov}_{2,3} \\ \text{Cov}_{3,1} & \text{Cov}_{3,2} & \sigma_3^2 \end{pmatrix} \tag{1}$$

Generalizing, a $m \times m$ covariance matrix has the variances of the random variable on the diagonal and the covariance on the off diagonal. The covariance between random variable i and j is $\text{Cov}_{ij} = \rho \sigma_i \sigma_j$ where ρ is the correlation coefficient, which takes on values between -1 and 1. Covariance can take on values between $-\infty$ and $+\infty$.

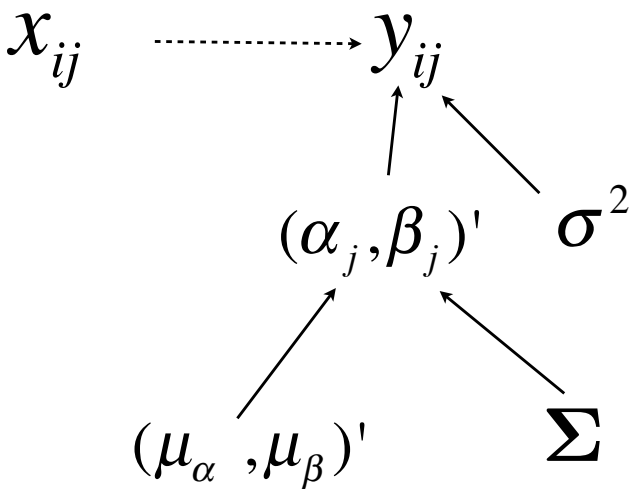
Covariance matrix for two parameter model

Imagine that we have $j = 1, ..., J$ groups with multiple observations within groups and we fit a two parameter linear model to each group, finding J intercepts and slopes. We denote the vector of intercepts as $\boldsymbol{\alpha}$ and the vector of slopes as $\boldsymbol{\beta}$. We can calculate the variance for each vector $(\sigma^2_\alpha, \sigma^2_\beta)$ as well as the correlation between the vectors ρ. The variance covariance matrix is thus:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma^2_\alpha & \text{Cov}(\boldsymbol{\alpha}, \boldsymbol{\beta}) \\ \text{Cov}(\boldsymbol{\beta}, \boldsymbol{\alpha}) & \sigma^2_\beta \end{pmatrix} \tag{2}$$

where $\text{Cov}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \text{Cov}(\boldsymbol{\beta}, \boldsymbol{\alpha}) = \rho \sigma_\alpha \sigma_\beta$

Modeling intercepts *and* slopes



$$\begin{pmatrix} \alpha_j \\ \beta_j \end{pmatrix} \sim \text{MVN}\left(\begin{pmatrix} \mu_\alpha \\ \mu_\beta \end{pmatrix}, \boldsymbol{\Sigma}\right) \quad \text{MVN} = \text{multivariate normal}$$

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_\alpha^2 & \rho\sigma_\alpha\sigma_\beta \\ \rho\sigma_\alpha\sigma_\beta & \sigma_\beta^2 \end{pmatrix}$$

Modeling intercepts *and* slopes

$$\begin{aligned}\left[\boldsymbol{\alpha}, \boldsymbol{\beta}, \mu_\alpha, \mu_\beta, \sigma^2_{\text{reg}}, \sigma^2_\alpha, \sigma^2_\beta, \rho | \mathbf{y}\right] &\propto \prod_{j=1}^{J} \prod_{i=1}^{n_j} \text{normal}(y_{ij} | \alpha_j + \beta_j x_{ij}, \sigma^2_{\text{reg}}) \\ &\times \text{MVN}\left(\begin{pmatrix} \alpha_j \\ \beta_j \end{pmatrix} \middle| \begin{pmatrix} \mu_\alpha \\ \mu_\beta \end{pmatrix}, \boldsymbol{\Sigma} \right) \\ &\times \text{priors on } \mu_\alpha, \mu_\beta, \sigma^2_{\text{reg}}, \sigma^2_\alpha, \sigma^2_\beta, \rho\end{aligned}$$

Modeling intercepts *and* slopes for more than one slope

$$
\begin{aligned}
\left[\boldsymbol{\beta}, \boldsymbol{\mu}_\beta, \sigma^2_{\text{reg}}, |\mathbf{y}\right] \quad &\propto \quad \prod_{j=1}^{J}\prod_{i=1}^{n_j} \text{normal}(y_{ij}|\mathbf{x'}_{ij}\boldsymbol{\beta}_j, \sigma^2_{\text{reg}}) \\
&\times \quad \text{MVN}\left(\begin{pmatrix} \beta_{0j} \\ \beta_{1j} \\ \beta_{2j} \\ \vdots \\ \beta_{mj} \end{pmatrix} \middle| \begin{pmatrix} \mu_{\beta_{0_j}} \\ \mu_{\beta_1} \\ \mu_{\beta_2} \\ \vdots \\ \mu_{\beta_m} \end{pmatrix}, \boldsymbol{\Sigma}\right) \\
&\times \quad \text{priors on } \boldsymbol{\mu}_\beta, \sigma^2_{\text{reg}}, \boldsymbol{\Sigma}
\end{aligned}
$$

Modeling intercepts and slopes for > 1 slope

The Wishart distribution:

$$f_{\mathbf{X}}(\mathbf{x}) = \frac{|\mathbf{x}|^{(n-p-1)/2} e^{-\operatorname{tr}(\mathbf{V}^{-1}\mathbf{x})/2}}{2^{\frac{np}{2}} |\mathbf{V}|^{n/2} \Gamma_p(\frac{n}{2})}$$

A vague prior on $\mathbf{\Sigma}$:

$$\mathbf{\Sigma} \sim \text{Wishart}(\mathbf{S}, m+1) \tag{3}$$

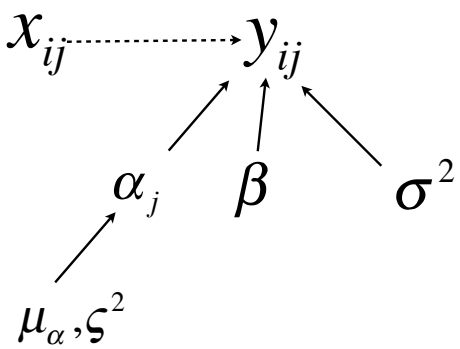
where m is the number of coefficients including the intercept and $\mathbf{S}$ is an $m \times m$ matrix with ones on the diagonal and zeros on the off diagonals.
Example code: `Sigma ~ dwish(S,y.Nvar + 1)`

Compute $\sigma's$ and ρ as derived quantities of the elements of $\mathbf{\Sigma}$. Remember, the `Sigma` in JAGS uses precisions not variances. For informed priors, see the `eivtools` package in R.

Guidance

- The Wishart distribution is an easy, useful way to impose reasonably vague priors on covariance matrices. See Gelman and Hill 2009, pages 376-380.
  - My experience with simulated data is that these priors are vague for the means but somewhat informative for the variances and for the correlation.
  - STAN has a distribution for priors on covariance matrices that appears to be superior to the Wishart, although the Wishart is widely used and recommended.
- It is also entirely feasible, if somewhat tedious, to simply expand the two parameter case (done in lab) to include more than one slope.
- We assume that there is a single variance for all random variables such that $\mathbf{\Sigma} = \sigma^2\mathbf{I}$, where $\mathbf{I}$ is the identity matrix with ones on the diagonal and zeros elsewhere.
- We assume that each random variable has its own variance σ_i^2 and the random variables are uncorrelated such that $\mathbf{\Sigma} = \mathbf{I}\boldsymbol{\sigma}^{2'}$.

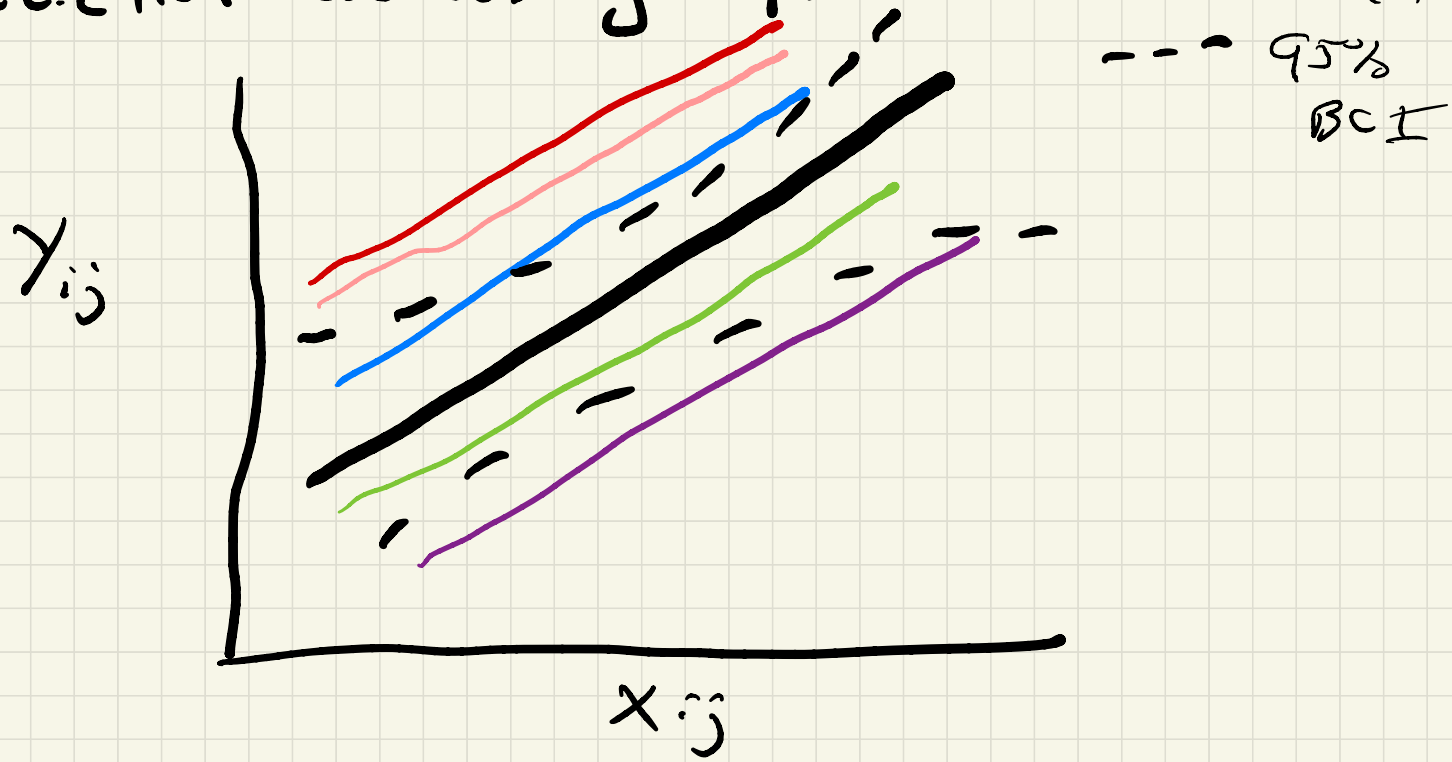
We can model the intercept (or slope):



Return to single slope, multiple intercepts

$$[\beta, \boldsymbol{\alpha}, \sigma^2, \mu_\alpha, \varsigma^2, |\mathbf{y}] \propto \prod_{i=1}^{n_j} \prod_{j=1}^{J} \text{normal}\left(y_{ij} | \alpha_j + \beta x_{ij}, \sigma^2\right)$$
$$\times \text{normal}\left(\alpha_j | \mu_\alpha, \varsigma^2\right)$$
$$\times \text{normal}\left(\beta | 0, 10000\right) \text{normal}\left(\mu_\alpha | 0, 1000\right)$$
$$\times \text{inverse gamma}\left(\sigma^2 | .001, .001\right) \text{inverse gamma}\left(\varsigma^2 | .001, .001\right)$$

Prediction across groups



Tempting, but wrong

$$M_i^{all} = \frac{1}{J} \sum_{j=1}^{J} M_{ij}$$

Simply compute a derived quantity by averaging over groups.

Inference to all possible groups:

1. Make a draw

$$\tilde{\alpha}_j \sim \text{normal}(\mu_\alpha, \sigma_\alpha^2)$$

2. Compute

$$\tilde{\mu} = \tilde{\alpha}_j + \beta x_i$$

The x_i are specified as a vector of data

3. Store $\tilde{\mu}$ at each MCMC iteration.

Inference to the specific groups studied:

1. Create a vector $p = (\frac{1}{J}, \ldots \frac{1}{J})$ of length J

2. Make a draw

$$\tilde{j} \sim \text{Categorical}(p)$$

(see dcat in JAGS)

3. Compute

$$\tilde{\mu} = \alpha_{\tilde{j}} + \beta_1 x_i$$

4. Store $\tilde{\mu}$ at each iteration

Finite population inference

1) You have a sample of J groups from a frame of N possible groups.

2) J is a reasonable large fraction of N ($> .05$ ish)

Prediction for a finite population of groups:

1. Create a vector $p = (\frac{1}{N}, \ldots \frac{1}{N})$ of length N

2. Make a draw
$$\tilde{j} \sim \text{Categorical}(p)$$
(see dcat in JAGS)

3. If $\tilde{j} \leq J$ compute:
$$\tilde{\mu} = \alpha_{\tilde{j}} + \beta_1 x_i$$

4. If $\tilde{j} > J$:

Make a draw
$$\tilde{\alpha}_i \sim \text{normal}(\mu_\alpha, \sigma_\alpha^2)$$

Compute
$$\tilde{\mu} = \tilde{\alpha}_i + \beta x_i$$

5. Store $\tilde{\mu}$ at each mcmc iteration.