

## Section 3

### Random Coefficients

#### Diagramming and writing the random carbon fertilizer model

Now we are interested in the effect of soil carbon and fertilizer type on N<sub>2</sub>O emissions. Model the effect of carbon as above, but include a group level effect of fertilizer type on the slope of the emission vs fertilizer addition model. This is to say that the slopes of the regressions are drawn from a distribution of fertilizer types. Index plot with  $i$ , site with  $j$ , and fertilizer type with  $k$ . Thus, there will be  $K$  slopes, one for each fertilizer type, drawn from a distribution with mean  $\mu_\beta$  and variance  $\zeta_\beta^2$ . Modify the carbon model you built in the previous step to incorporate effect of fertilizer type.

Be careful here because the group level effects are formed for two separate groups, site and fertilizer type. You might be tempted (or perhaps terrified) to think that you need to model the covariance in this problem, which is not the case. This is required only if you are modeling slope and intercept as group level effects for the same grouping variable, for example, site. You will see how this is done in the last problem. Think about it. Covariance between the slope and intercept is only important if they are being estimated from data within the same group. There is only a single fertilizer type with each group, so it cannot covary with the intercept.

1. Draw a Bayesian network for a linear regression model of N<sub>2</sub>O emission ( $y_{ijk}$ ) on fertilizer addition ( $x_{ijk}$ ) and soil carbon content ( $w_j$ ).
2. Write out the posterior and joint distributions for a linear regression model of N<sub>2</sub>O emission ( $y_{ijk}$ ) on fertilizer addition ( $x_{ijk}$ ) and soil carbon content ( $w_j$ ). Choose appropriate distributions for each random variable.

$$\begin{aligned}
 [\alpha, \beta, \sigma, \kappa_\alpha, \eta_\alpha, \zeta_\alpha, \mu_\beta, \zeta_\beta | y] \propto & \prod_{i=1}^n \prod_{j=1}^k \text{normal}(\log(y_j) | g(\alpha_j, \beta, \log(m_{ijk})), \sigma^2) \\
 & \times \text{normal}(\alpha_j | h(\kappa, \eta, w_j), \zeta_\alpha^2) \\
 & \times \text{normal}(\kappa_\alpha | 0, 1000) \\
 & \times \text{normal}(\eta_\alpha | 0, 1000) \\
 & \times \text{uniform}(\zeta_\alpha^2 | 0, 100) \\
 & \times \text{normal}(\beta_{ijk} | o(\mu_\beta, w_{ijk}), \zeta_\beta^2) \\
 & \times \text{normal}(\mu_\beta | 0, 10000) \\
 & \times \text{uniform}(\zeta_\beta^2 | 0, 100) \\
 & \times \text{uniform}(\sigma^2 | 0, 100)
 \end{aligned}$$

#### Fitting the random carbon fertilizer model with JAGS

Modify your random intercepts model to implement the model that include soil carbon content and fertilizer type as covariates at the site level. Use the data and initial values for JAGS provided below to allow you to concentrate on writing JAGS code for the model.

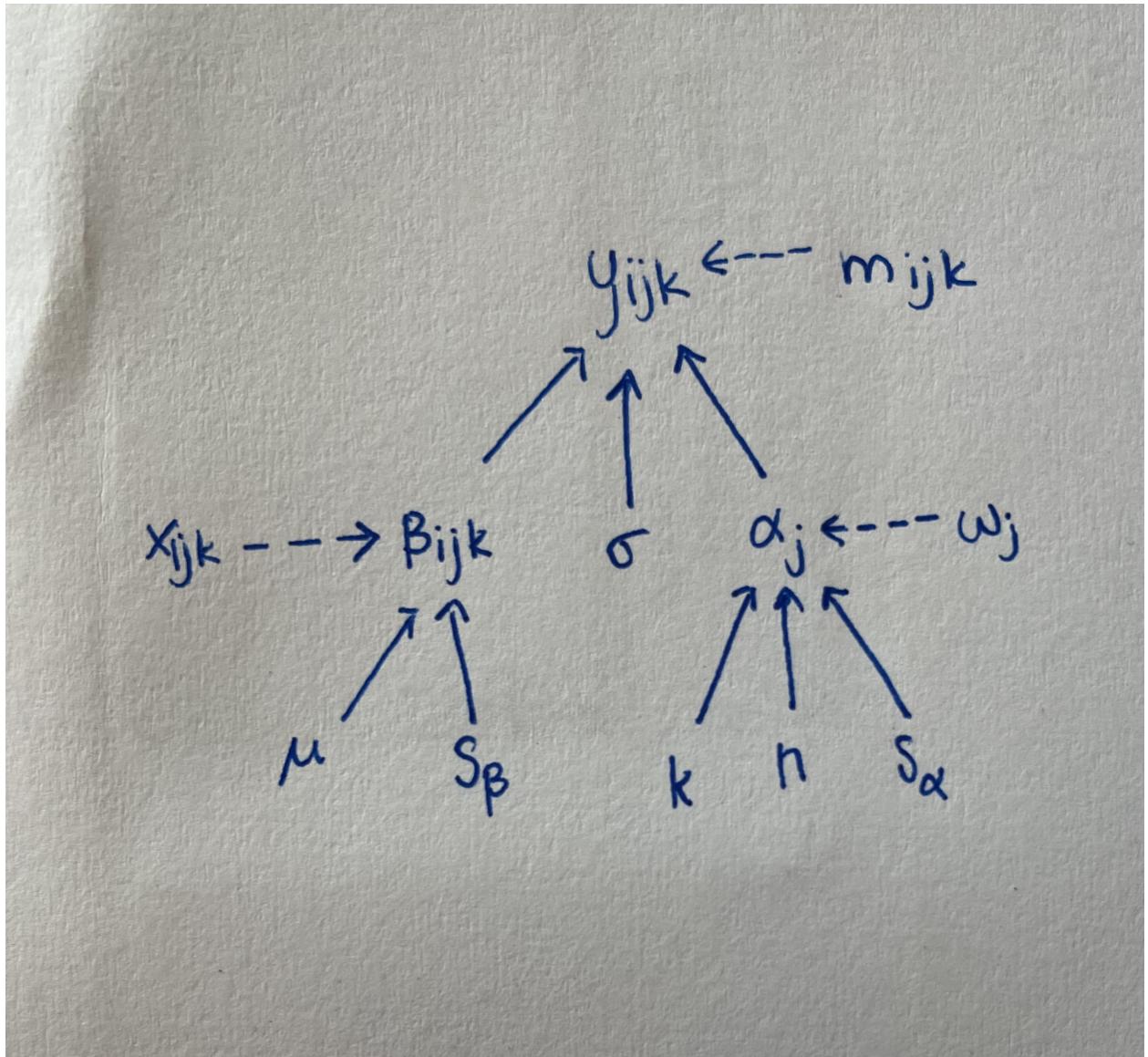


Figure 1: Directed acyclic graph for the joint distribution.

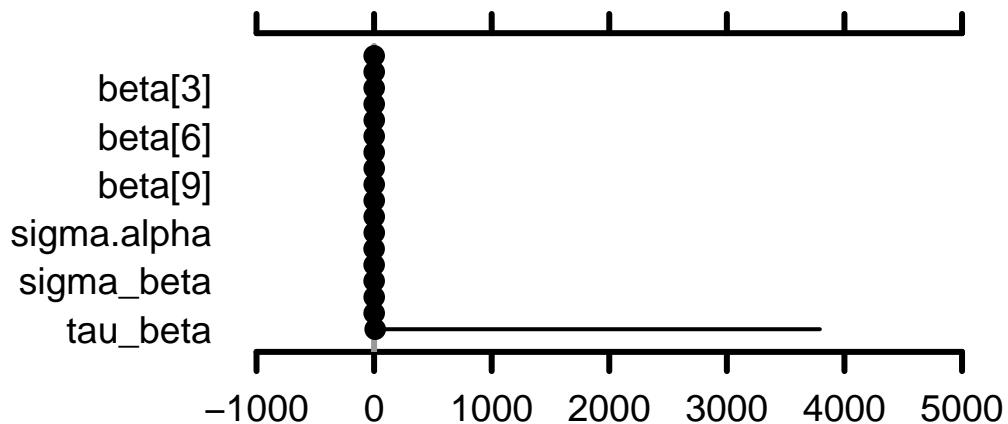
3. Write the code for the model. Compile the model and execute the MCMC to produce a coda object. Produce trace plots of the chains for model parameters, excluding  $\alpha$  and a summary table of these same parameters. Assess convergence and look at the effective sample sizes for each of these parameters. Do you think any of the chains need to be run for longer and if so why? Make a horizontal caterpillar plot for the the  $\alpha$ .

```

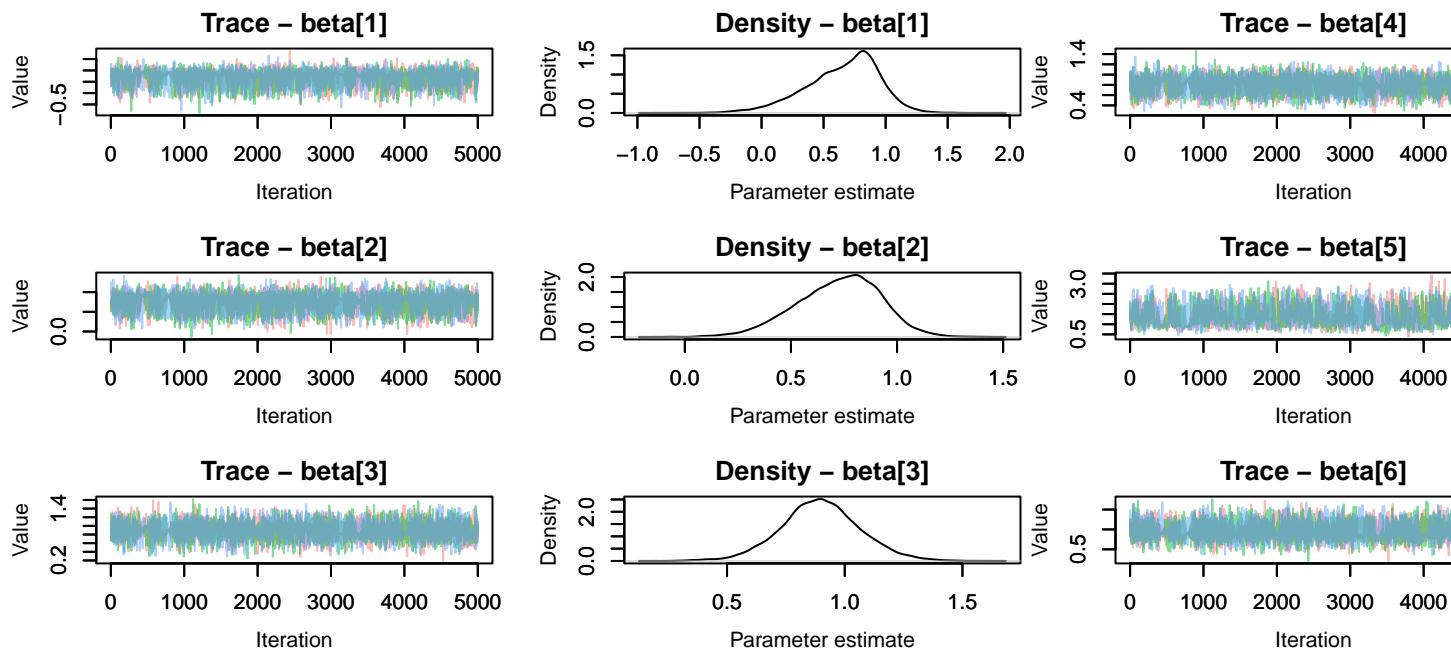
1 # random intercepts, group level effects JAGS code
2 {
3   sink("RandomCarbonFertilizerJAGS.R")
4   cat(
5     "model{
6
7     # priors
8     sigma ~ dunif(0,100)
9     tau.reg <- 1/sigma^2
10
11    kappa ~ dnorm (0,.0001)
12    eta ~ dnorm (0,.0001)
13    sigma.alpha ~ dunif(0,100)
14    tau.alpha<-1/sigma.alpha^2
15
16    mu.beta ~ dnorm(0,.0001)
17    sigma_beta ~ dunif(0,100)
18    tau_beta <- 1/sigma_beta^2
19
20    # loop over fertilizer
21    for(k in 1:n.ferts){
22      beta[k]~dnorm(mu.beta,tau_beta)
23    }
24
25    # loop over sites itself
26    for (j in 1:n.sites){
27      alpha[j]~dnorm(kappa+eta*w[j],tau.alpha)
28    }
29
30    # loop within sites
31    for (i in 1:length(log.emission)) {
32      log_mu[i] <- alpha[group[i]] + beta[fertilizer[i]] * log.n.input.centered[i]
33      log.emission[i] ~ dnorm(log_mu[i], tau.reg)
34    }
35
36  }
37
38  ",fill = TRUE)
39  sink()
40 }

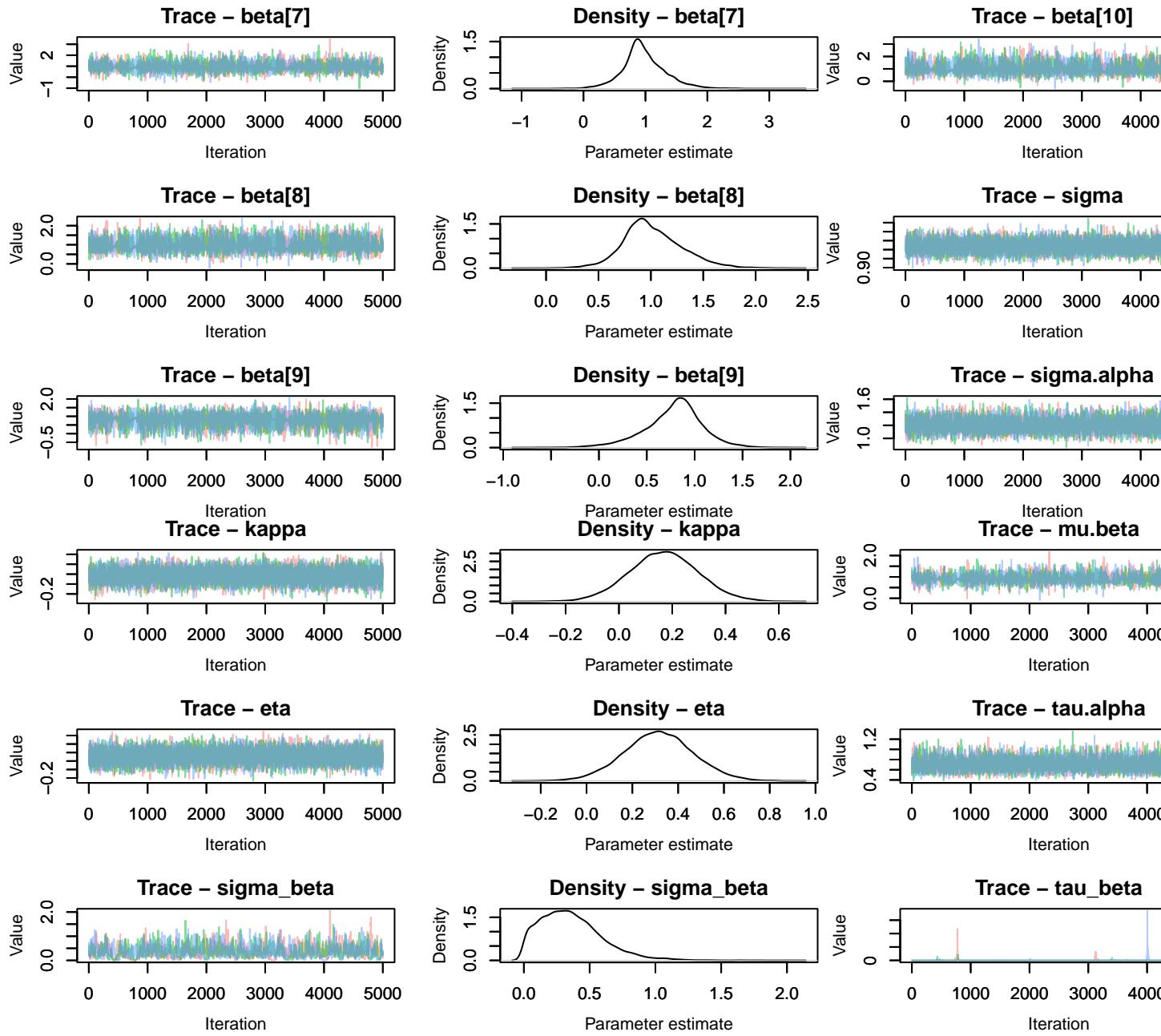
## Compiling model graph
## Resolving undeclared variables
## Allocating nodes
## Graph information:
##   Observed stochastic nodes: 563
##   Unobserved stochastic nodes: 123
##   Total graph size: 3220
##
```

```
## Initializing model
```



### Parameter Estimate





|                | mean      | sd           | 2.5%         | 50%       | 97.5%     | Rhat |
|----------------|-----------|--------------|--------------|-----------|-----------|------|
| ## beta[1]     | 0.6524910 | 2.943808e-01 | -0.008420908 | 0.6968467 | 1.1410407 | 1.00 |
| ## beta[2]     | 0.7234489 | 1.973231e-01 | 0.311901081  | 0.7388387 | 1.0774575 | 1.00 |
| ## beta[3]     | 0.8969473 | 1.696023e-01 | 0.561920210  | 0.8949498 | 1.2403741 | 1.00 |
| ## beta[4]     | 0.7962269 | 1.396504e-01 | 0.511897803  | 0.8018954 | 1.0587658 | 1.00 |
| ## beta[5]     | 1.4125616 | 4.358353e-01 | 0.758958705  | 1.3691137 | 2.3438986 | 1.00 |
| ## beta[6]     | 0.9714324 | 1.919373e-01 | 0.617230866  | 0.9591071 | 1.3702920 | 1.00 |
| ## beta[7]     | 0.9756363 | 3.549145e-01 | 0.295713440  | 0.9382155 | 1.7663828 | 1.00 |
| ## beta[8]     | 1.0181926 | 2.842052e-01 | 0.506349973  | 0.9825686 | 1.6408399 | 1.00 |
| ## beta[9]     | 0.7801365 | 3.072079e-01 | 0.084109924  | 0.8122625 | 1.3521857 | 1.00 |
| ## beta[10]    | 1.0736659 | 3.890979e-01 | 0.414491362  | 1.0023513 | 1.9857140 | 1.00 |
| ## sigma       | 1.0185104 | 3.405029e-02 | 0.955600374  | 1.0173703 | 1.0884580 | 1.00 |
| ## sigma.alpha | 1.2024527 | 1.000758e-01 | 1.020792562  | 1.1967648 | 1.4161466 | 1.00 |

```

## kappa          0.1683210 1.290479e-01 -0.085204351 0.1684052      0.4257763 1.00
## eta           0.3150159 1.510151e-01  0.019239366 0.3148303      0.6149731 1.00
## sigma_beta   0.3594342 2.360186e-01  0.016241975 0.3301775      0.9022940 1.00
## mu.beta       0.9297842 1.804142e-01  0.593299659 0.9169124      1.3217650 1.00
## tau.alpha    0.7059746 1.170913e-01  0.498635945 0.6982041      0.9596768 1.00
## tau_beta     1808.3369954 2.988652e+04  1.228298455 9.1728671 3790.7256469 1.12
##               n.eff
## beta[1]        2105
## beta[2]        2341
## beta[3]        4117
## beta[4]        3603
## beta[5]         813
## beta[6]        2451
## beta[7]        3631
## beta[8]        3389
## beta[9]        5129
## beta[10]       2935
## sigma          6540
## sigma.alpha   5240
## kappa          9849
## eta            9313
## sigma_beta    733
## mu.beta        3576
## tau.alpha     5534
## tau_beta      1145

```

4. How do you assess whether fertilizer type a good predictor of N<sub>2</sub>O emission? How would we compare the slope for fertilizer type 1 to type 5?

Fertilizer does not appear to be a good predictor for N<sub>2</sub>O emissions because most of our  $\beta$  coefficients fall between 0 and 1, meaning they are not informative. The slope for fertilizer 5 is over double that of fertilizer 1, so our predicted emissions will be higher using fertilizer 5 than when using the same amount of fertilizer 1.

### Diagramming and writing the random coefficients model

Last we return to the random intercepts model where you assumed that different sites had different intercepts but the same slope, which is to say that individual sites had emission responses to fertilizer that were parallel. This seems unreasonable (particularly when you look at site-level plots of the data), representing the need to model group effects on intercepts and slopes. The idea is that both the slope and the intercept are random variables drawn from a distribution of slopes and intercepts where variation in the values of the random variable is attributable to unspecified spatial differences among sites. It is tempting to simply add a distribution for the slopes in the same way you modeled the intercepts, and you will see papers where this is done (wrongly). However, when you seek to understand group effects on multiple parameters you must account for the way that the parameters covary. Write a model for group effects on slope and intercepts. Exploit the following hints:

Represent the slope and intercept as a two element vector for each group and use a multivariate normal distribution in the same way you used a normal for the intercept. So the individual slopes ( $\alpha_j$ ) and intercept ( $\beta_j$ ) will be drawn from

$$\begin{pmatrix} \alpha_j \\ \beta_j \end{pmatrix} \sim \text{multivariate normal}(\begin{pmatrix} \mu_\alpha \\ \mu_\beta \end{pmatrix}, \Sigma)$$

The second parameter in the multivariate normal is a variance-covariance matrix representing how the slope and intercept covary. It has variance terms on the diagonal, and covariance terms on the off-diagonal, i.e.,

$$\Sigma = (\zeta_\alpha^2 \text{Cov}(\alpha, \beta) \text{Cov}(\alpha, \beta) \zeta_\beta^2)$$

The covariance terms are defined as  $\text{Cov}(\alpha, \beta) = \rho \zeta_\alpha \zeta_\beta$  where  $\rho$  is the coefficient of correlation between  $\alpha$  and  $\beta$ .

5. Draw a Bayesian network and write out the posterior and joint distribution for a linear regression model of  $\text{N}_2\text{O}$  emission ( $y_{ijk}$ ) on fertilizer addition ( $x_{ijk}$ ).

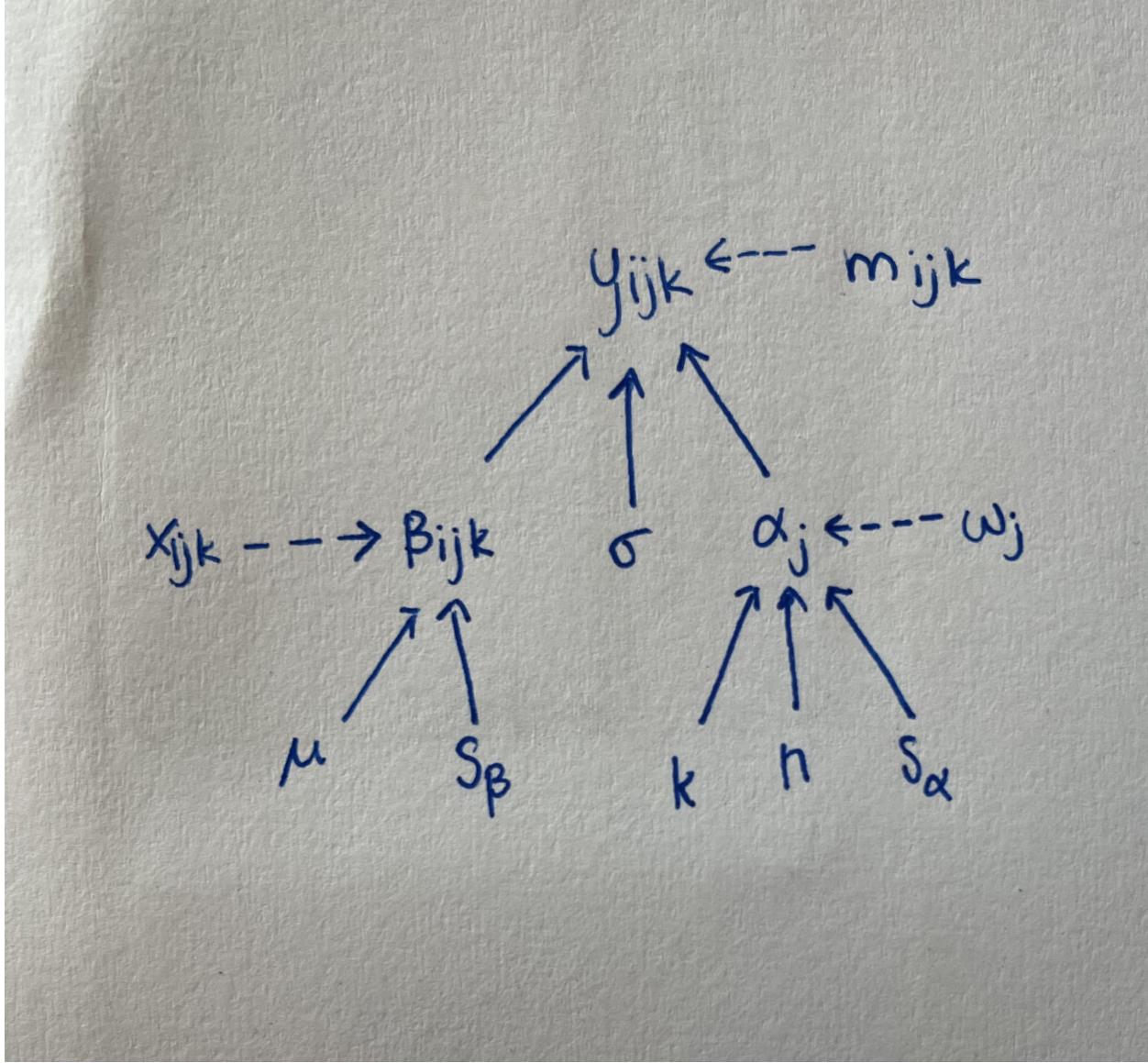


Figure 2: Directed acyclic graph for the joint distribution.

$$[\alpha, \beta, \sigma, \mu_\alpha, \mu_\beta, \sigma_\alpha, \sigma_\beta, \rho \mid y] \propto \prod_{i=1}^n \prod_{j=1}^k [\log(y_{ij}) \mid \alpha_j + \beta_j x_{ij}, \sigma^2][\binom{\alpha_j}{\beta_j} \mid \binom{\mu_\alpha}{\mu_\beta}, \Sigma][\mu_\alpha][\mu_\beta][\sigma^2][\sigma_\alpha^2][\sigma_\beta^2][\rho]$$

6. Write out the posterior and joint distributions for a linear regression model of N<sub>2</sub>O emission ( $y_{ij}$ ) on fertilizer addition ( $x_{ij}$ ). Choose appropriate distributions for each random variable.

$$\begin{aligned} [\alpha, \beta, \sigma, \mu_\alpha, \mu_\beta, \sigma_\alpha, \sigma_\beta, \rho \mid y] &\propto \prod_{i=1}^n \prod_{j=1}^k \text{normal}(\log(y_j) \mid \alpha_j + \beta_j x_{ij}, \sigma^2) \\ &\times \text{MVN}\left(\begin{pmatrix} \alpha_j \\ \beta_j \end{pmatrix} \mid \begin{pmatrix} \mu_\alpha \\ \mu_\beta \end{pmatrix}, \Sigma\right) \\ &\times \text{normal}(\mu_\alpha \mid 0, 1000) \\ &\times \text{normal}(\mu_\beta \mid 0, 1000) \\ &\times \text{uniform}(\sigma_\alpha^2 \mid 0, 100) \\ &\times \text{uniform}(\sigma_\beta^2 \mid 0, 100) \\ &\times \text{uniform}(\sigma^2 \mid 0, 100) \\ &\times \text{uniform}(\rho \mid 0, 100) \end{aligned}$$

### Fitting the random coefficients model with JAGS

We now want to allow *both* slopes and intercepts to vary by site as described in the math exercise. This is a pretty challenging coding problem, so I thought it would be more useful for you to study code that works than to try to write the code yourself. So take a careful look at the following and discuss it with your group. Are the slope and intercepts correlated? How could you make a probabilistic statement about the correlation?

Write the code for the model. Compile the model and execute the MCMC to produce a coda object. Produce trace plots of the chains for model parameters, excluding  $\alpha$  and a summary table of these same parameters. Assess convergence and look at the effective sample sizes for each of these parameters. Do you think any of the chains need to be run for longer and if so why? Make a horizontal caterpillar plot for the the  $\alpha$ .

```

1  {
2  sink("RandomCoefJAGS.R")
3  cat("
4  model{
5
6    # priors for within site model
7    sigma ~ dunif(0, 200)
8    tau.reg <- 1 / sigma^2
9
10   # likelihood
11   # note that the data have been log-transformed in R prior to running this model
12   for (i in 1:length(log.emission)) {
13     log_mu[i] <- alpha[group[i]] + beta[group[i]] * log.n.input.centered[i]
14     log.emission[i] ~ dnorm(log_mu[i], tau.reg)
15   }
16
17   # model for group intercept and slope
18   for (j in 1:n.sites) {
19     alpha[j] <- B[j, 1]  # group level intercept
20     beta[j]  <- B[j, 2]  # group level slope
21     B[j, 1:2] ~ dmnorm(B.hat[j, 1:2], Tau.B)
22     B.hat[j, 1] <- mu.alpha  # required by JAGS syntax
23     B.hat[j, 2] <- mu.beta   # required by JAGS syntax
24   }
}

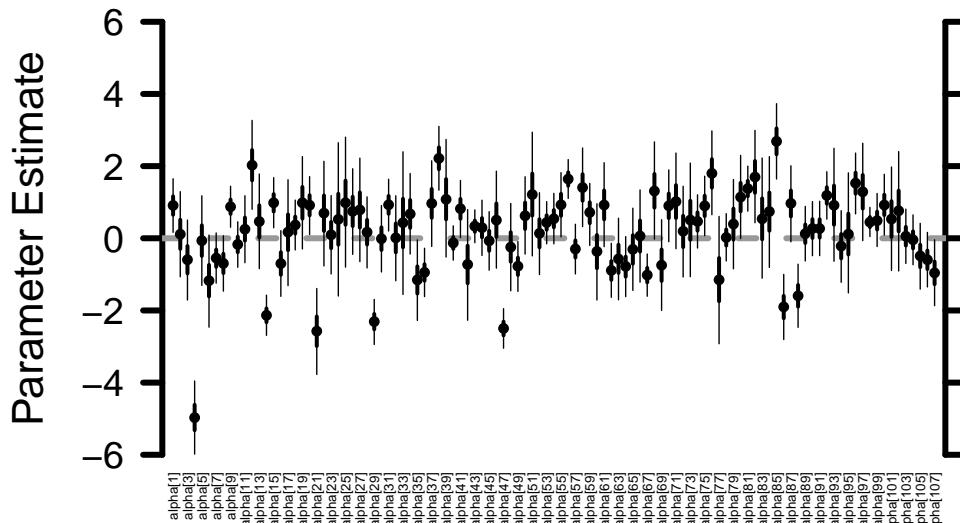
```

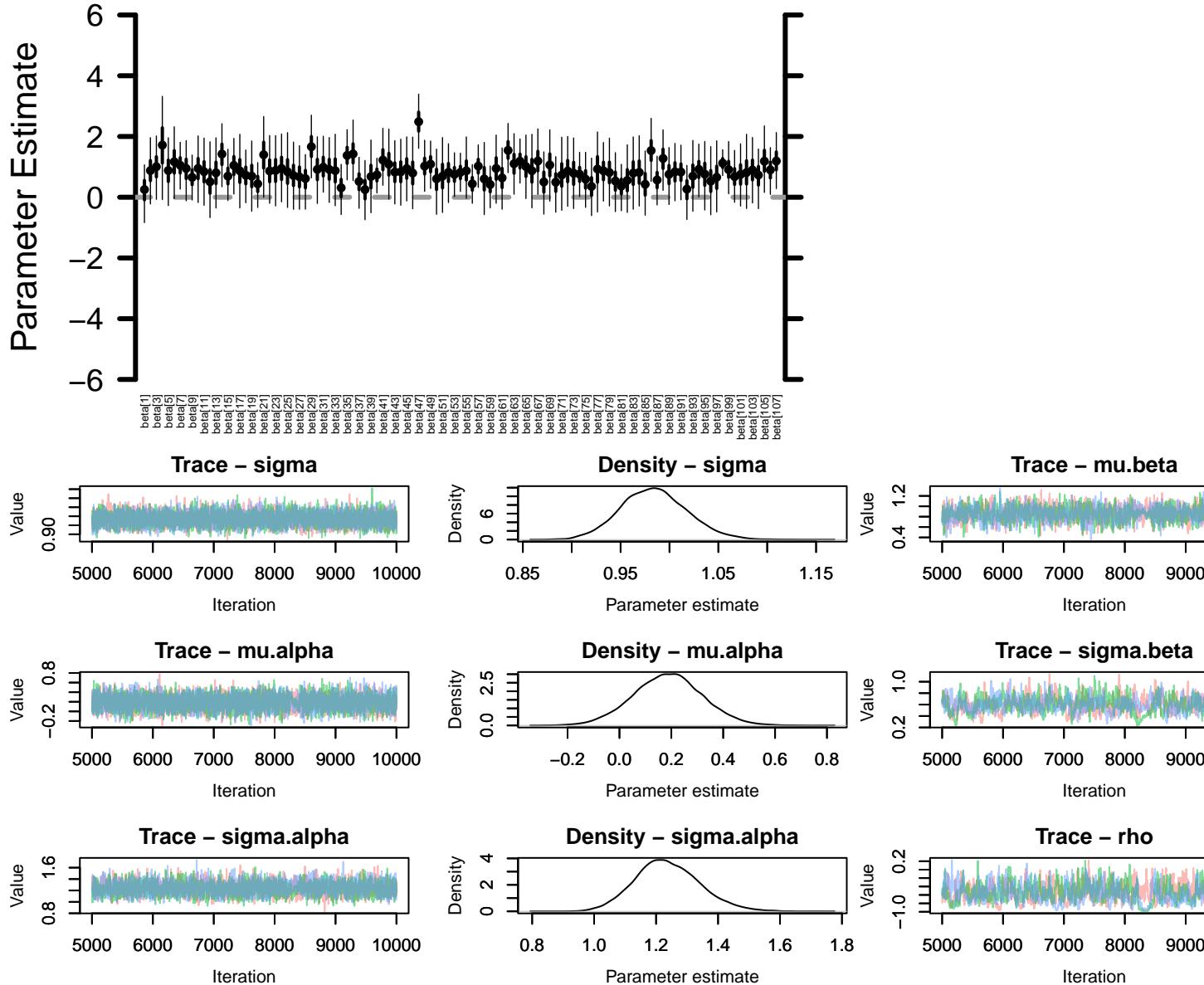
```

25
26   # priors
27   mu.alpha ~ dnorm(0, .0001)
28   mu.beta ~ dnorm(0, .0001)
29
30   # inverse of covariance matrix
31   Tau.B[1:2, 1:2] <- inverse(Sigma.B[1:2, 1:2])
32
33   # diagonal elements of covariance matrix
34   Sigma.B[1, 1] <- sigma.alpha^2
35   sigma.alpha ~ dunif(0,200)
36   Sigma.B[2, 2] <- sigma.beta^2
37   sigma.beta ~ dunif(0,200)
38
39   # covariance is correlation coef. x product of standard deviations
40   Sigma.B[1, 2] <- rho * sigma.alpha * sigma.beta
41   Sigma.B[2, 1] <- Sigma.B[1,2]
42   rho ~ dunif(-1, 1)
43
44 }
45
46 ",fill = TRUE)
47 sink()
48 }
```

```

## Compiling model graph
## Resolving undeclared variables
## Allocating nodes
## Graph information:
##   Observed stochastic nodes: 563
##   Unobserved stochastic nodes: 113
##   Total graph size: 2713
##
## Initializing model
```





```

##          mean      sd   2.5%    50%  97.5% Rhat n.eff
## sigma     0.985  0.034  0.921  0.984  1.054    1 10156
## mu.alpha  0.187  0.133 -0.073  0.186  0.448    1  9096
## sigma.alpha 1.238  0.105  1.044  1.232  1.459    1  3932
## mu.beta   0.858  0.124  0.612  0.859  1.100    1  2458
## sigma.beta 0.598  0.132  0.342  0.596  0.868    1   851
## rho       -0.553  0.206 -0.919 -0.567 -0.122    1   809

```

Based on our correlation measure  $\rho$ , our slopes and intercepts have a negative correlation of approximately 0.56. However,  $\rho$ ,  $\mu_\beta$ , and  $\sigma_\beta$  should be run for longer as their traceplots appear less 'grassy' than we would expect from a converged model.