Data Analysis: Diet Change and Gut Microbiomes

Shih-Ni Prim

1 Introduction

The current project uses data from “Fat, fibre and cancer risk in African Americans and rural Africans” by O’Keefe et al. (2015). While the authors investigated the role of fat and fiber in cancer risks, we model the baseline data (130 taxa from 112 samples) using Bayesian hierarchical models. The current report analyzes this subset with a full model using a MCMC sampler that applies Gibbs and Metropolis sampling, introduces two simplified models, compares the three models using WAIC, evaluates our final model, and proposes a model for the full data to examine the association between diet change and gut microbiomes.

2 Model Overview

Our first model (M1) is provided below. Note that, to ensure that all values in α are greater than zero, we add 1 to the numerator and the number of taxa to the denominator, so the sum of the vector is 1.[1]

$$\mathbf{Y}_i|\theta_i, M_i \overset{ind}{\sim} \text{Multinomial}(N_i, \theta_i), \theta_i|M_i \overset{ind}{\sim} \text{Dirichlet}\{exp(M_i)\alpha\}, M_i \overset{iid}{\sim} \text{Normal}(\mu, \sigma^2)$$

$$\mu \sim \text{Normal}(m, s^2), \sigma^2 \sim \text{InvGamma}(a, b), \mathbf{Y}_i = (Y_{1i}, \dots, Y_{ti}), \alpha = \frac{(\sum_{i=1}^n \mathbf{Y}_i) + 1}{(\sum_{i=1}^n N_i) + t}, t = 130, n = 112$$

As seen above, the parameter α of the Dirichlet distribution is multiplied by e^{M_i}. M_i thus adjusts how diffuse or concentrated the probabilities of different categories are. Our data has 130 taxa, but only a few categories have higher probabilities. Figure 1 shows the probabilities of the 130 taxa from one random draw of the Dirichlet distribution with α as the parameters, which is quite concentrated, as well as α multiplied by three different values of e^{M_i}, which becomes more diffuse as M_i is around 3 and does not change much for $M_i > 5$. Through M_i, we can make the distribution of the probabilities more diffuse than in the first plot, but we do not want M_i to be too large, since the posterior distribution should be concentrated around

[1] Note we use the subscript t, instead of m, for the number of taxa, since m is used as the prior mean of μ.

some microbiomes. Thus, we set the hyperparameters of μ at $m = 5, s^2 = 4$. For the prior of σ^2, we select $shape = a = 1, scale = b = 1$, so the variance of M_i has high probabilities between 0 and 5, as shown in the sixth plot of Figure 1.

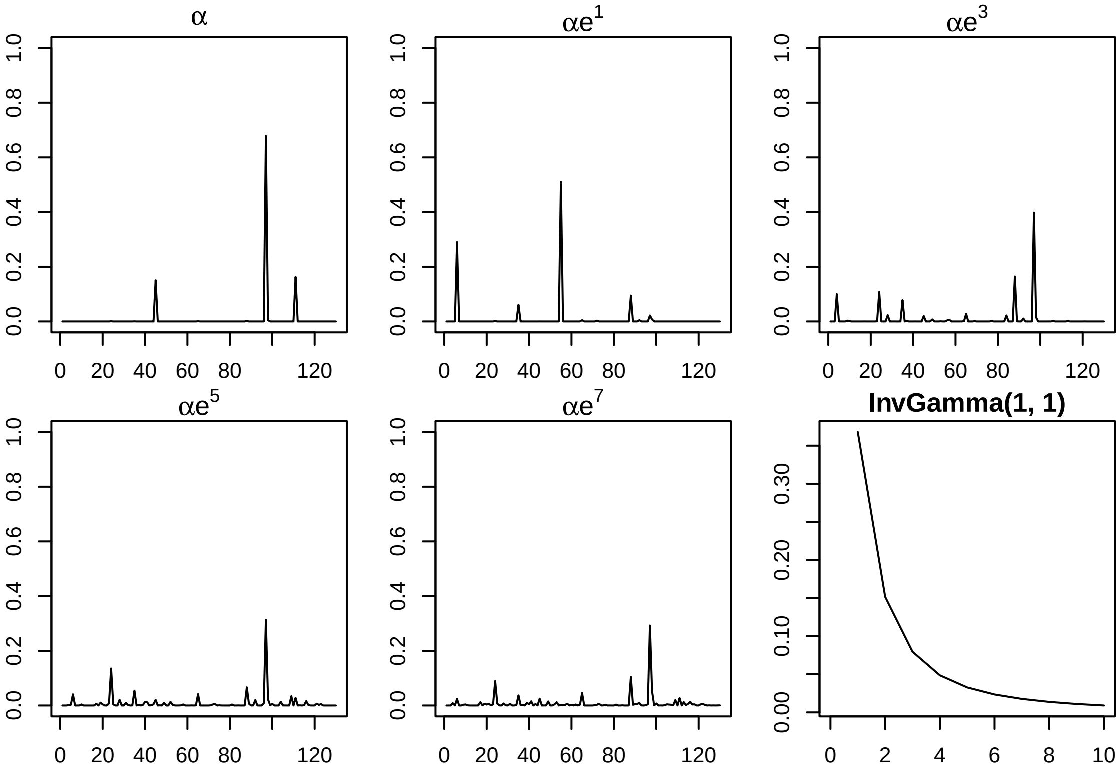

Figure 1: Different Values of M_i on the Distribution of Probabilities and InvGamma(1,1)

The full conditional distributions of θ_i, μ, and σ^2 are provided below.[2]

$$\theta_i|\text{rest} \sim \text{Dirichlet}(\mathbf{Y}_i + e^{M_i}\alpha), \mu|\text{rest} \sim \text{N}\left(\frac{s^2 \sum_{i=1}^{n} M_i + m\sigma^2}{ns^2 + \sigma^2}, \frac{\sigma^2 s^2}{ns^2 + \sigma^2}\right)$$

$$\sigma^2|\text{rest} \sim \text{InvGamma}\left(a + \frac{n}{2}, \frac{\sum_{i=1}^{n}(M_i - \mu)^2}{2} + b\right)$$

2.1 MCMC Sampler

Our MCMC sampler for M1 uses Gibbs sampling for θ_i, μ, and σ^2 and Metropolis sampling for M_i. The candidate distribution is set as normal centering with the previous draw of M_i. The initial values of standard deviations are set as 1 for all M_i, and tuning of the standard deviation is used in the burn-in period to keep

[2] See the appendix for the derivations.

the acceptance rates around 0.4. For Metropolis sampling, we only use the non-zero components of θ_i and the corresponding components in α to calculate the log-likelihood to avoid generating infinity or negative infinity. For initial values, θ_i are set as the sample proportions for each sample, M_i are set as 1, μ and σ^2 are set as $m = 4$ and $s^2 = 4$. Besides generating random draws of θ_i, μ, σ^2, and M_i, the MCMC sampler calculates the log-likelihood with respect to the multinomial distribution for Watanabe–Akaike information criterion (WAIC). θ_i are stored in a list of matrices, which will be used for posterior predictive checks for model evaluation. Trace plots–eight of which are shown in Figure 2–suggest that all parameters quickly converge, so these initial values are unlikely to change the results.

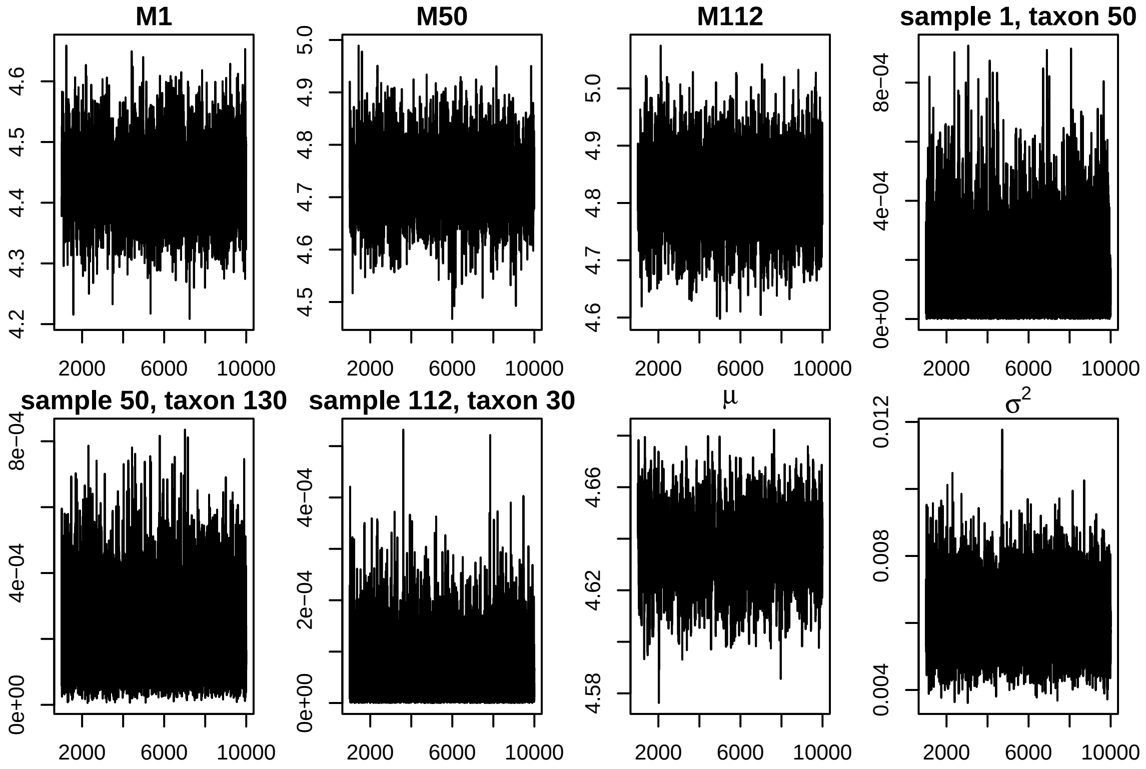

Figure 2: Example Trace Plots

2.2 Two Alternative Models and Model Selection

The second model (M2) set $M_i = 0$ for all i, thus the model becomes:

$$\mathbf{Y}_i|\theta_i \overset{ind}{\sim} \text{Multinomial}(N_i, \theta_i), \theta_i \overset{iid}{\sim} \text{Dirichlet}(\alpha)$$

The full conditional distribution of θ_i is then $\theta_i|\text{rest} \sim \text{Dirichlet}(\mathbf{Y}_i + \alpha)$. This model removes several layers of the hierarchical model from M1, so now the probability vector of each of the samples is determined by the sum of $\mathbf{Y}_i$ and the same vector α. We call this model the *pooled ratio* model.

The third model (M3) set $M_i = \infty$ for all i, thus $\theta_i = \alpha$. The model becomes:

$$\mathbf{Y}_i \overset{iid}{\sim} \text{Multinomial}(N_i, \alpha)$$

This model further simplifies the previous two models, and the different samples are now assumed to be distributed as the multinomial distribution with the parameter α. This model sets the probabilities of each taxon, so we call this model the *fixed ratio* model.

We calculate WAIC for each model, as shown in Table 1, and select the model with the lowest WAIC–M2, the pooled ratio model–as our final model for later discussion.

Table 1: WAIC of the Models

	M1 (Full hierarchical model)	M2 (Pooled ratio)	M3 (Fixed ratio)
WAIC	7.6166582×10^4	7.4972051×10^4	1.7055996×10^6

2.3 Model Evaluation: Posterior Preditive Checks

For our chosen model (pooled ratio model), we perform posterior predictive checks by generating a 130×112 matrix for each iteration and then comparing summary statistics of the observed data to the same summary statistics of the 9000 generated datasets. Since the observed data is right-skewed with lots of values close to zero and some large values and that the large values are more pertinent, we choose the 90% percentile as a threshold value at 133. Figure 3 shows the five summary statistics: 25th percentile, percentage above threshold, 25th percentile above the threshold, 90th percentile, and the maximum. The vertical lines show the summary statistics of the observed data, while the histograms show the distributions of the summary statistics from generated data. The plots show that, for percentages above threshold, 90th percentile, and maximum, the vertical lines are somewhere in the middle of the distributions and the p-values are far from 0 and 1. For 25th percentile (overall) and 25th percentile above the threshold, the p-values are 0. The results suggest that our pooled ratio model does a reasonable job making predictions for large values but misses the smaller values. The inherent traits of the data–with most values close to zero and some values much greater–seems a major challenge. If predicting large values is indeed an important task, we should consider applying variable selection methods to improve our modeling and predictions.

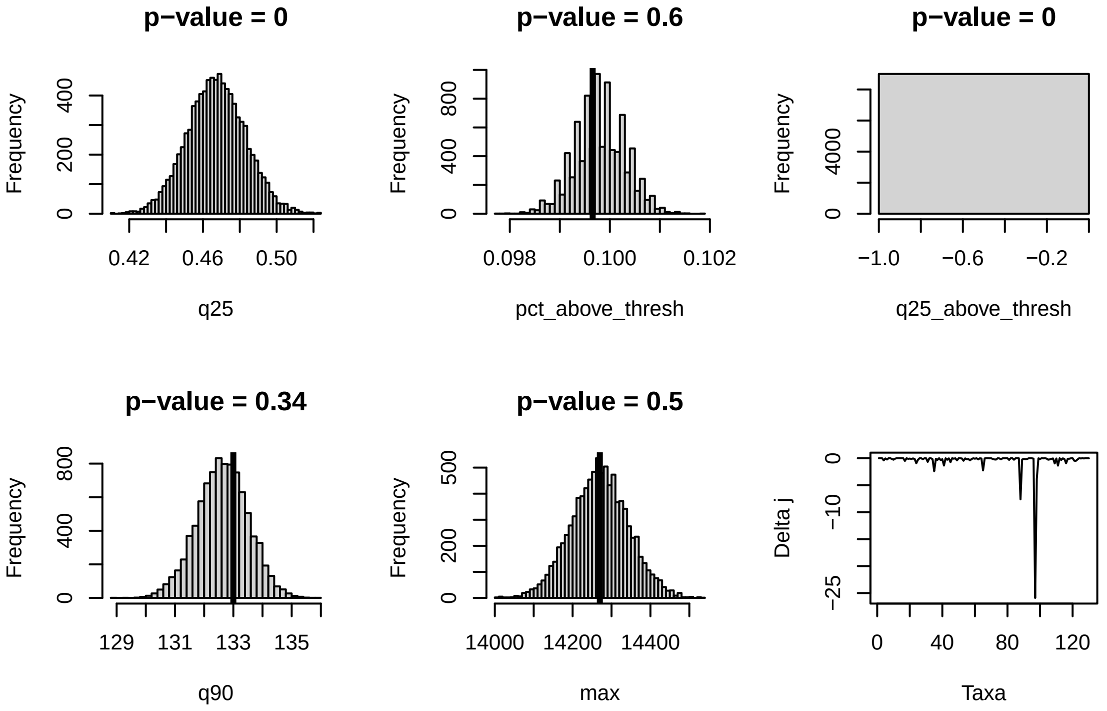

Figure 3: Posterior Predictive Checks (plots 1-5) and $\triangle_j$ for the 130 taxa (plot 6)

Lastly, we compare the microbiomes of the two nationalities, "AAM" and "AFR" by finding the value

$$\triangle_j = \left(\sum_{i;X_i=AAM} \theta_{ji}/n_{AAM} \right) - \left(\sum_{i;X_i=AFR} \theta_{ji}/n_{AFR} \right)$$

As shown in the sixth plot in Figure 3, $\triangle_j$ are close to zero for most taxa but are quite far from zero for a few taxa. (The minimum is -25.913 for taxon 97, with the name "Prevotella melaninogenica et rel.") Furthermore, out of the 130 taxa, 124 $\triangle_j$ are negative, suggesting that θ_i tend to be smaller for African Americans than for rural Africans. Both findings are worth further investigation.

2.4 Model Proposal for Full Dataset

Going beyond the baseline data, we propose a model for the full datasest to examine the association between diet change and the makeup of microbiomes. The full dataset is a matrix with the dimension 130 × 222. The 222 samples came from 38 individuals (21 African Americans and 17 rural Africans). We propose to include covariates in our proposed model: sex (female, male), nationality (AAM, AFR), bmi group (lean, overweight, obese), and time point within group (1, 2). We are more interested in whether diet change is associated with the change of gut microbiomes, so we do not propose to include two of the covariates–group (DI: dietary intervention, ED: initial endoscopy days, HE: home environment) and time point (1, 2, 3, 4, 5, 6)–since they indicate finer distinctions than the periods before and after the diet change. Our proposed hierarchical model is described below.

$$\mathbf{Y}_i|\theta_i \overset{ind}{\sim} \text{Multinomial}(N_i, \theta_i), \theta_i \overset{ind}{\sim} \text{Normal}(\beta_{0i} + \mathbf{X}_i \beta_i, \sigma^2 I), \beta_i \overset{ind}{\sim} \text{Normal}(0, \lambda_i^2 \tau^2)$$

$$\lambda_i \sim \text{Half Cauchy}(1), \sigma^2 \sim \text{InvGamma}(a, b), i = 1, 2, \ldots, 222$$

$$\mathbf{N}_i = \sum_{j=1}^{t} Y_{ji}, \beta_i = (\beta_{1i}, \ldots, \beta_{135i})^T, \mathbf{X}_{i,130\times 135} = \begin{bmatrix} \mathbf{A}_i & \mathbf{B}_i \end{bmatrix}, \mathbf{A}_i = \mathbf{I}_{130\times 130}$$

$\mathbf{B}_i$ is a 130×5 matrix that has the same components for each row: $x_{131i}, x_{132i}, x_{133i}, x_{134i}, x_{135i}$ where

$$\begin{cases} x_{131i} = 1 & \text{if sex = female for ith sample, 0 otherwise} \\ x_{132i} = 1 & \text{if nationality = AAM for ith sample, 0 otherwise} \\ x_{133i} = 1 & \text{if bmi group = lean for ith sample, 0 otherwise} \\ x_{134i} = 1 & \text{if bmi group = overweight for ith sample, 0 otherwise} \\ x_{135i} = 1 & \text{if time point within group = 1 for ith sample, 0 otherwise} \end{cases}$$

The horseshoe prior will shrink β_i towards zero, and we could set a threshold and more closely examine the taxa and covariates corresponding to β_{ji} greater than the threshold. Furthermore, β_{135i} could be used for inference; for example, if most of the 95% credible intervals of β_{135i} do not include zero, we could conclude that the diet change is associated with the changes of gut microbiomes.

3 Conclusion

It is no easy task to analyze data about 130 taxa of gut microbiomes to answer a seemingly straightforward question: “Can diet change affect the makeup of microbiomes?” Nonetheless, by constructing three models assuming the multinomial distribution on the baseline data, we were able to show the differences between the two populations through $\triangle_j$. We propose a model that incorporates covariates, including sex, bmi groups, and diet change, as well as variable selection through the use of a shrinkage prior to hopefully zoom in on the important taxa to clarify the associations between diet and microbiomes.

4 Appendix: Derivation of Full Conditional Distributions

The full hierarchical model for M1 is:

$$\left[\prod_{i=1}^{n} f(\mathbf{Y}_i|\theta_i)\right]\left[\prod_{i=1}^{n} \pi(\theta_i|M_i)\right]\left[\prod_{i=1}^{n} \pi(M_i|\mu,\sigma^2)\right]\pi(\mu|m,s^2)\pi(\sigma^2|a,b)$$

4.1 θ_i

$$P(\theta_i|\text{rest}) \propto f(\mathbf{Y}_i|\theta_i)\pi(\theta_i|M_i) \propto \left(\theta_{1i}^{y_1 i} \ldots \theta_{ti}^{y_t i}\right)\left(\theta_{1i}^{e^{M_i}\alpha_1 - 1} \ldots \theta_{ti}^{e^{M_i}\alpha_t - 1}\right) \propto \theta_{1i}^{y_{1i}+e^{M_i}\alpha_1-1} \ldots \theta_{ti}^{y_{ti}+e^{M_i}\alpha_t-1}$$

$$\theta_i|\text{rest} \sim \text{Dirichlet}(\mathbf{Y}_i + e^{M_i}\alpha)$$

4.2 μ

$$P(\mu|\text{rest}) \propto \left[\prod_{i=1}^{n} \pi(M_i|\mu,\sigma^2)\right]\pi(\mu|m,s^2) \propto \left[\prod_{i=1}^{n} e^{-\frac{(M_i-\mu)^2}{2\sigma^2}}\right]e^{-\frac{(\mu-m)^2}{2s^2}} \propto e^{-\frac{s^2\sum_{i=1}^{n}(M_i-\mu)^2+\sigma^2(\mu-m)^2}{2\sigma^2 s^2}}$$

$$\propto e^{-\frac{s^2\sum_{i=1}^{n}(\mu^2-2M_i\mu)+\sigma^2\mu^2-2m\sigma^2\mu}{2\sigma^2 s^2}} \propto e^{-\frac{(ns^2+\sigma^2)\mu^2-2(s^2\sum_{i=1}^{n}M_i+m\sigma^2)\mu}{2\sigma^2 s^2}} \propto e^{-\frac{(\mu-(s^2\sum_{i=1}^{n}M_i+m\sigma^2)/(ns^2+\sigma^2))^2}{2(\sigma^2 s^2/(ns^2+\sigma^2))}}$$

$$\mu|\text{rest} \sim \text{N}\left(\frac{s^2\sum_{i=1}^{n}M_i+m\sigma^2}{ns^2+\sigma^2}, \frac{\sigma^2 s^2}{ns^2+\sigma^2}\right)$$

4.3 σ^2

$$P(\sigma^2|\text{rest}) \propto \left[\prod_{i=1}^{n} \pi(M_i|\mu,\sigma^2)\right]\pi(\sigma^2|a,b) \propto \left[\prod_{i=1}^{n} \frac{1}{\sqrt{\sigma^2}}e^{-\frac{(M_i-\mu)^2}{2\sigma^2}}\right]\left[(\sigma^2)^{-a-1}e^{-\frac{b}{\sigma^2}}\right]$$

$$\propto (\sigma^2)^{-\frac{n}{2}}e^{-\frac{\sum_{i=1}^{n}(M_i-\mu)^2}{2\sigma^2}}(\sigma^2)^{-a-1}e^{-\frac{b}{\sigma^2}} \propto (\sigma^2)^{-(a+\frac{n}{2})-1}e^{-\frac{\frac{1}{2}\sum_{i=1}^{n}(M_i-\mu)^2+b}{\sigma^2}}$$

$$\sigma^2|\text{rest} \sim \text{InvGamma}\left(a+\frac{n}{2}, \frac{\sum_{i=1}^{n}(M_i-\mu)^2}{2}+b\right)$$

5 Code

```r
# a function to calculate sample proportions for each column separately
divide <- function(x){
  sums <- colSums(x)
  x2 <- x
  for (i in 1:ncol(x)){
    x2[,i] <- x[,i]/sums[i]
  }
  return(x2)
}
# log posterior function
log_post <- function(theta, M, alpha, mu, sigma2){
  theta <- t(as.matrix(theta))
  post <- ddirichlet(theta, exp(M)*alpha, log = T, sum.up = T) +
    dnorm(M, mu, sqrt(sigma2), log = TRUE)
  return(post)
}
# model 1
MCMC_diet1 <- function(Y, alpha, iters, burn, m, s2, a, b){
  library(DirichletReg)
  library(invgamma)
  tik <- proc.time()
  Ni <- colSums(Y)
  N <- sum(Ni)
  t <- dim(Y)[1]
  n <- dim(Y)[2]
  # thetas has iters matrices, each of dimension mxn
  thetas <- list()
  keepers <- matrix(0, nrow = iters, ncol = n+2)
  colnames(keepers) <- c(paste0("M", 1:n), "mu", "sigma2")
  loglike <- matrix(0, nrow = iters, ncol = n)
  # record keeping
  att <- acc <- rep(0, n)
  MH <- rep(1, n)
  # initial thetas set as sample proportion of each sample
  thetas[[1]] <- divide(Y)
  keepers[1,] <- c(rep(1, n), m, s2)
  loglike[1,] <- 0
  for (iter in 2:iters){
    # Gibbs sampling
    # update thetas
    thetas[[iter]] <- matrix(0, t, n)
    for (i in 1:n){
      thetas[[iter]][,i] <- rdirichlet(1, t(Y[,i] + exp(keepers[iter-1, i])*alpha))
      loglike[iter,i] <- dmultinom(Y[,i], prob = thetas[[iter]][,i], log = TRUE)
    }
    # update mu
    meanz <- (s2*sum(keepers[iter-1, 1:n]) +
                t*keepers[iter-1, n+2])/(n*s2+keepers[iter-1, n+2])
    varz <- keepers[iter-1, n+2]*s2/(n*s2+keepers[iter-1, n+2])
    keepers[iter, n+1] <- rnorm(1, meanz, sqrt(varz))
    # update sigma2
    newa <- a+n/2
    newb <- sum((keepers[iter-1, 1:n]-keepers[iter, n+1])^2)/2+b
    keepers[iter, n+2] <- rinvgamma(1, shape = newa, scale = newb)
    # Metropolis for Mi
    for (j in 1:n){
      att[j] <- att[j]+1
      can <- rnorm(1, mean = keepers[iter-1, j], sd = MH[j])
      # only look at observations with non-zero thetas
      # this method does not seem to work
      thetaz <- thetas[[iter]][,j][which(thetas[[iter]][,j] != 0)]
      alphaz <- alpha[which(thetas[[iter]][,j] != 0)]
      curlp <- log_post(thetaz, keepers[iter-1, j],
                        alphaz, keepers[iter, n+1], keepers[iter, n+2])
      canlp <- log_post(thetaz, can, alphaz, keepers[iter, n+1], keepers[iter, n+2])
      R = canlp - curlp
      if (!is.na(R)){
        if (log(runif(1)) < R){
          acc[j] <- acc[j]+1
          keepers[iter, j] <- can
        }
        else {
          keepers[iter, j] <- keepers[iter-1, j]
        }
      }
    }
      if(iter<burn){for(j in 1:length(att)){if(att[j]>50){
        if(acc[j]/att[j] < 0.3){MH[j] <- MH[j]*0.8}
        if(acc[j]/att[j] > 0.5){MH[j] <- MH[j]*1.2}
        acc[j] <- att[j] <- 0
      }}}
    }
  tok <- proc.time()
  out <- list(theta = thetas, rest = keepers, acc_rate = acc/att,
              time = tok - tik, loglike = loglike)
  return(out)
}

waic_cal <- function(mat){
  mi <- colMeans(mat)
  vi <- apply(mat, 2, FUN = var)
  waic <- -2*sum(mi)+2*sum(vi)
  return(waic)
}
M1 <- MCMC_diet1(Y, alpha, 10000, 1000, 5, 4, 1, 1)
# model 2: pooled ratio
MCMC_diet2 <- function(Y, alpha, iters){
  library(DirichletReg)
  library(invgamma)
  tik <- proc.time()
  Ni <- colSums(Y)
  N <- sum(Ni)
  t <- dim(Y)[1]
  n <- dim(Y)[2]
  # thetas has iters matrices, each of dimension mxn
  thetas <- list()
  loglike <- matrix(0, nrow = iters, ncol = n)
  # initial thetas set as sample proportion of each sample
  thetas[[1]] <- divide(Y)
  loglike[1,] <- 0
  for (iter in 2:iters){
    # Gibbs sampling
    # update thetas
    thetas[[iter]] <- matrix(0, t, n)
    for (i in 1:n){
      thetas[[iter]][,i] <- rdirichlet(1, t(Y[,i] + alpha))
      loglike[iter,i] <- dmultinom(Y[,i], prob = thetas[[iter]][,i], log = TRUE)
    }
  }
  tok <- proc.time()
  out <- list(theta = thetas, time = tok - tik, loglike = loglike)
  return(out)
}
M2 <- MCMC_diet2(Y, alpha, iters = 10000)
# model 3: fixed ratio
MCMC_diet3 <- function(Y, alpha){
  tik <- proc.time()
  Ni <- colSums(Y)
  N <- sum(Ni)
  t <- dim(Y)[1]
  n <- dim(Y)[2]
  loglike <- matrix(0, 1, ncol = n)
  for (i in 1:n){
      loglike[1, i] <- dmultinom(Y[,i], prob = alpha, log = T)
  }
  tok <- proc.time()
  out <- list(time = tok - tik, loglike = loglike)
}
```

```
M3 <- MCMC_diet3(Y, alpha)
# posterior predictive checks
PPC <- function(list, Y, iters, burn, thresh){
  library(moments)
  Ni <- colSums(Y)
  n <- ncol(Y)
  t <- nrow(Y)
  D <- matrix(0, nrow = (iters-burn), ncol = 5)
  colnames(D) <- c("q25", "pct_above_thresh","q25_above_thresh", "q90", "max")
  for (iter in (burn+1):iters){
    preds <- matrix(0, nrow = t, ncol = n)
    for (i in 1:n){
      preds[,i] <- Ni[i]*list[[iter]][,i]
    }
    D[iter-burn,] <- c(quantile(preds,0.25), mean(preds>thresh),
                       quantile(preds>thresh,0.25),quantile(preds,0.90),max(preds))
  }
  D <- as.data.frame(D)
  return(D)
}
thresh <- quantile(c(Y), .9)
ppc2 <- PPC(M2$theta, Y, 10000, 1000, thresh)
strings <- c(Y)
sum_stats <- c(quantile(strings,.25), mean(strings>thresh),
               quantile(strings[strings>thresh],0.25),quantile(strings,0.90),max(strings))
# rearrange the result so that each matrix is for 1 taxon
M2_mat <- list()
for (j in 1:130){
  M2_mat[[j]] <- matrix(0, 112, iteration)
  for (i in 1:iteration){
    M2_mat[[j]][,i] <- M2$theta[[i]][j,]
  }
}
# calculate delta j
delta <- rep(0, 130)
ind_AAM <- which(X == "AAM")
naam <- length(ind_AAM)
ind_AFR <- which(X == "AFR")
nafr <- length(ind_AFR)
for (i in 1:130){
  meanz <- rowMeans(M2_mat[[i]])
  delta[i] <- sum(meanz[ind_AAM]/naam - sum(meanz[ind_AFR])/nafr)
}
```