Concepts of Bayesian Data Analysis

Group 25: **Raïsa Carmen** *s0204278*, **Vancesca Dinh** *r0830510*, **Arthur Terlinden**, **Barouyr Demirjian** *r0916873*, **Chaojie Huang** *r0816192*

1 Task A: Cohort study smoking

Abbott, Yin, Reed and Yano performed a 12-year cohort study to investigate the association between smoking and stroke. Among 3435 smokers, 171 had a stroke; while among 4437 non-smokers, 117 has a stroke.

(1) Assuming a non-informative prior for the probability of disease among those exposed θ_+, give the analytical posterior for the probability of disease among those exposed. Do the same for the probability of disease among those not exposed θ_-.

In general, if $Y|\theta \sim Bin(n, \pi)$ is the data model and $\theta \sim Beta(\alpha, \beta)$ is the prior model, then the posterior model will be

$$\theta|y \sim Beta(\alpha + y, \beta + n - y)$$

.

In the Beta distribution, there are several possibilities for uninformative priors (Tuyl, Gerlach, and Mengersen 2008). $\alpha = 1, \beta = 1$ generates an uniform probability density function. Jeffrey prior sets $\alpha = 0.5, \beta = 0.5$. And Kerman proposes to use a "Neutral" prior where $\alpha = \frac{1}{3}, \beta = \frac{1}{3}$ (Kerman 2011). We will assume $\alpha = 1, \beta = 1$ which yields:

$$\theta|y \sim Beta(\alpha + y, \beta + n - y)$$

For non-smokers, we thus have:

$$\theta_-|y_- \sim Beta(1 + 117, 1 + 4437 - 117)$$

$$\theta_-|y_- \sim Beta(118, 4321)$$

For smokers, we have:

$$\theta_+|y_+ \sim Beta(1 + 171, 1 + 3435 - 171)$$

$$\theta_+|y_+ \sim Beta(172, 3265)$$

(2) Give some summary measures of the above posterior distributions.

For the non-smokers, the posterior expected value is

$$E(\theta_-|y_-) = \frac{\alpha + y}{\alpha + \beta + n} = \frac{1 + 117}{1 + 1 + 4437} = 0.02658256$$

the posterior mode is

$$Mode(\theta_-|y_-) = \frac{\alpha + y - 1}{\alpha + \beta + n - 2} = \frac{1 + 117 - 1}{1 + 1 + 4437 - 2} = 0.02636917$$

and the posterior variance is

$$Var(\theta_-|y_-) = \frac{(\alpha+y)(\beta+n-y)}{(\alpha+\beta+n)^2(\alpha+\beta+n+1)} = \frac{(1+117)(1+4437-117)}{(1+1+4437)^2(1+1+4437+1)} = 5.83 * 10^{-6}$$

For the smokers, the posterior expected value is

$$E(\theta_+|y_+) = \frac{\alpha+y}{\alpha+\beta+n} = \frac{1+171}{1+1+3435} = 0.05004364$$

the posterior mode is

$$Mode(\theta_+|y_+) = \frac{\alpha+y-1}{\alpha+\beta+n-2} = \frac{1+171-1}{1+1+3435-2} = 0.04978166$$

and the posterior variance is

$$Var(\theta_+|y_+) = \frac{(\alpha+y)(\beta+n-y)}{(\alpha+\beta+n)^2(\alpha+\beta+n+1)} = \frac{(1+171)(1+3435-171)}{(1+1+3435)^2(1+1+3435+1)} = 1.38 * 10^{-5}$$

(3) Can you visualise the posterior distribution of the relative risk, defined as

$$\theta_{RR} = \frac{\theta_+}{\theta_-}$$

Use a sample from the above derived analytical posterior distribution to answer this question. Give some summary measures of the posterior distribution of the relative risk. Can you conclude that there is an association between smoking and stroke?

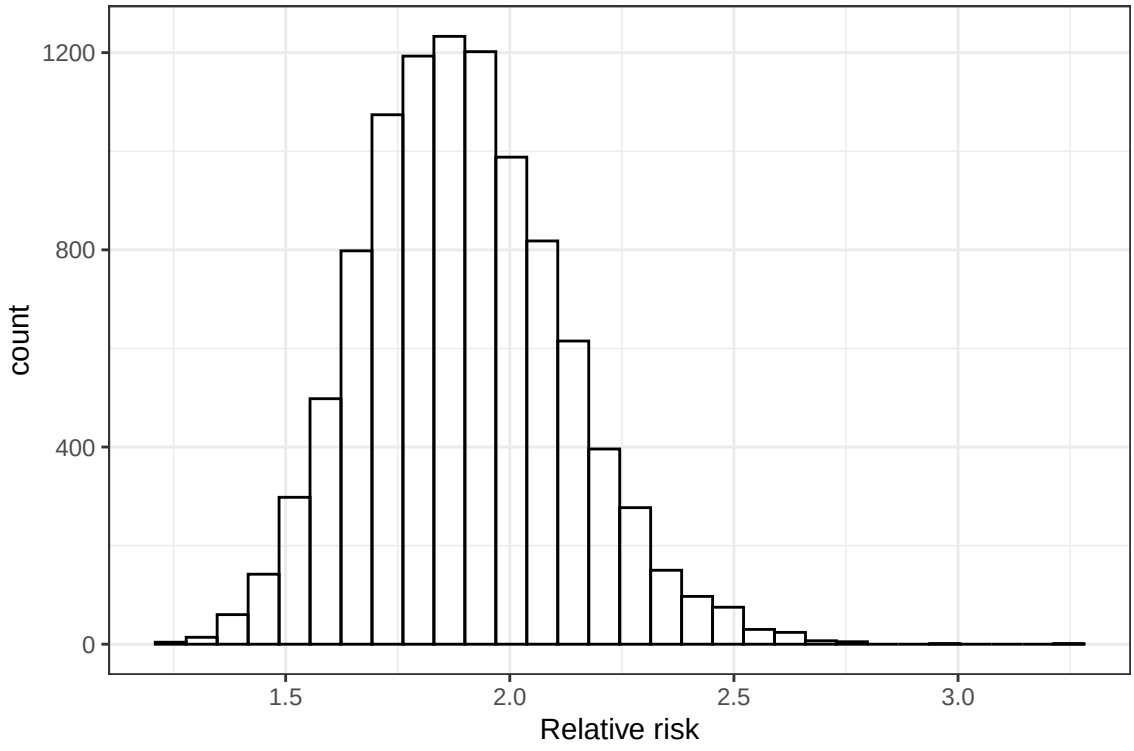

Figure 1: Posterior distribution of the relative risk.

We draw a random sample of size 10^4 from both posterior Beta distributions and calculate the relative risk based on those. Figure 1 shows a histogram of the random sample of the relative risk. If the relative risk is significantly different from one, we can conclude that there is indeed an association between smoking ans stroke. Let's investigate this.

The mean relative risk in our sample is 1.895 and the variance is 0.05. The median relative risk is 1.883. The minimal relative risk is our sample is 1.209, meaning that the probability that the relative risk $<= 1$ is equal to zero in our sample! Indeed, 95% of our sample lies between 1.499 and 2.378. We can thus confidently say that smokers have a statistically significantly higher risk of getting a stroke than non smokers.

(4) Write jags, OpenBugs or Nimble code, to obtain an MCMC samples for the above problem.

(5) Check convergence of the MCMC chain.

(6) Compare the summary measures obtained from the MCMC chain with the results obtained from questions (1)-(3).

(7) What is the attributable risk of smoking to the incidence of stroke? The attributable risk is defined as

$$\theta_{AR} = \frac{\theta_{RR} - 1}{\theta_{RR}}$$

Extend your Bayesian MCMC code to derive the answer.

2 Task B: Dose-response model

The first project concerns determining the dose-response relationship of a possible toxic product. Diethylene Glycol Dimethyl Ether (DYME), also referred to as diglyme, bis(2-methoxyethyl) ether is a high-volume industrial chemical with diverse applications. It is used to make industrial solvents, cosmetics, protective coatings, solvents in chemical synthesis, and is used in manufacturing of textile dyes. Price et al. (1987) describe a study in which timed-pregnant CD-1 mice were dosed by gavage with DYME in distilled water. Dosing occurred during the period of major organogenesis and structural development of the foetuses (gestational age 6 through 15). Relating the dose of DYME to the incidence of malformations in foetuses gives the following results:

Dose	Number of foetuses	Number of malformations
0.0	282	67
62.5	225	34
125.0	290	193
250.0	261	250
500.0	141	141

(1) Assume that the likelihood of the experiment is specified by

$$y \sim binomial(N, \pi)$$

$$logit(\pi) = \alpha + \beta d.$$

Here β is the parameter of interest. Take vague priors for α and β. Write jags, OpenBugs or Nimble code for this problem. Take 2 MCMC chains with different starting values, and check convergence with the appropriate techniques.

To denote the uncertainty of parameters α and β, we specify two vague priors as a massive variance (extremely small precision) for these two parameters. The first vague prior is:

$$\alpha \sim N(0, 10000)$$

$$\beta \sim N(0, 10000)$$

And the second vague prior is:

$$\alpha \sim t(0, 0.0001, 5)$$

$$\beta \sim t(0, 0.0001, 5)$$

Our parameter settings include running 2 MCMC chains with different starting values for α and β. The first chain starts with ($\alpha = 0$, $\beta = 0$), while the second chain starts with ($\alpha = -0.5$,$\beta = 0.1$). We run each chain for a total of 10000 iterations, with the first 5000 iterations used as a burn-in period. We will use Nimble for this task.

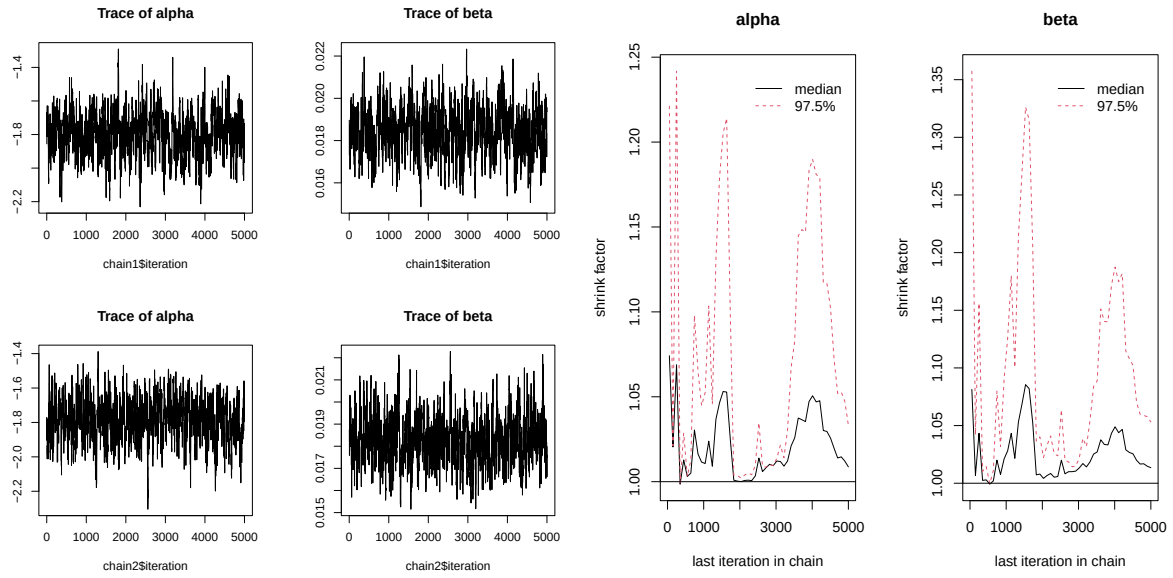

Figure 2: Trace and Gelman-Rubin diagnostic plots of α and β (prior: normal distribution)

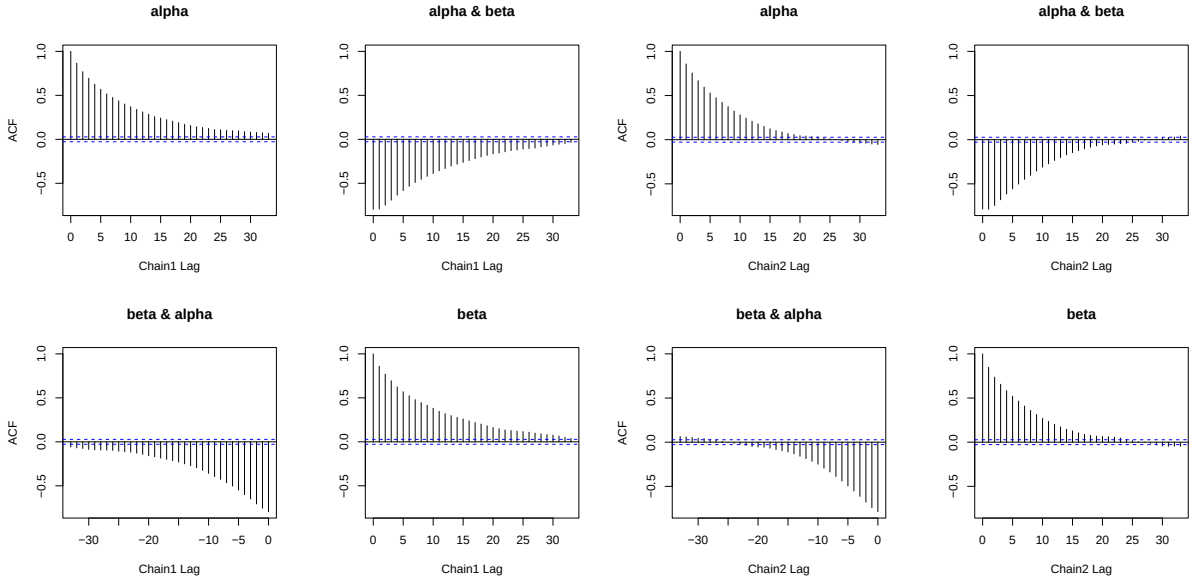

Figure 3: Autocorrelation plots of α and β (prior: normal distribution)

To assess the convergence of the model with a prior normal distribution, we employed several diagnostic tools including trace plots, the Gelman-Rubin diagnostic test, and autocorrelation plots. We observed from the trace plots of α and β (refer to Figure 2) that the estimates from each chain quickly stabilized around a steady state. Additionally, both chains were found to converge around the same conclusion. Furthermore, we conducted the

Gelman-Rubin diagnostic test, which showed that the estimated potential scale reduction factors of α and β were both 1. These results suggest that our model has converged well. Moreover, the Gelman-Rubin diagnostic plots (refer to Figure 2) support this conclusion, as both the potential scale reduction factors of α and β were found to decrease quickly and remain stable as the number of iterations increased. We also examined the autocorrelation plots (refer to Figure 3), which indicated low autocorrelation. The autocorrelation decreased and remained around zero as the lag number increased, indicating that the chains have mixed well. Overall, these diagnostic tools suggest that our model that the prior is normal distribution has converged well, and the inference based on the Markov chain Monte Carlo simulation is reliable.

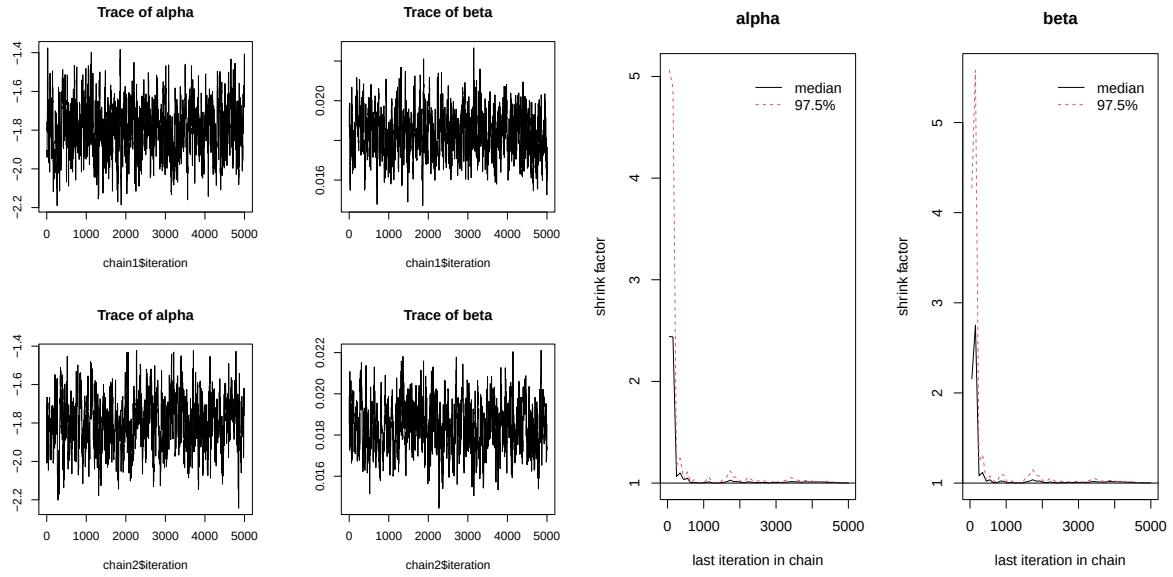

Figure 4: Trace and Gelman-Rubin diagnostic plots of α and β (prior: t-distribution)

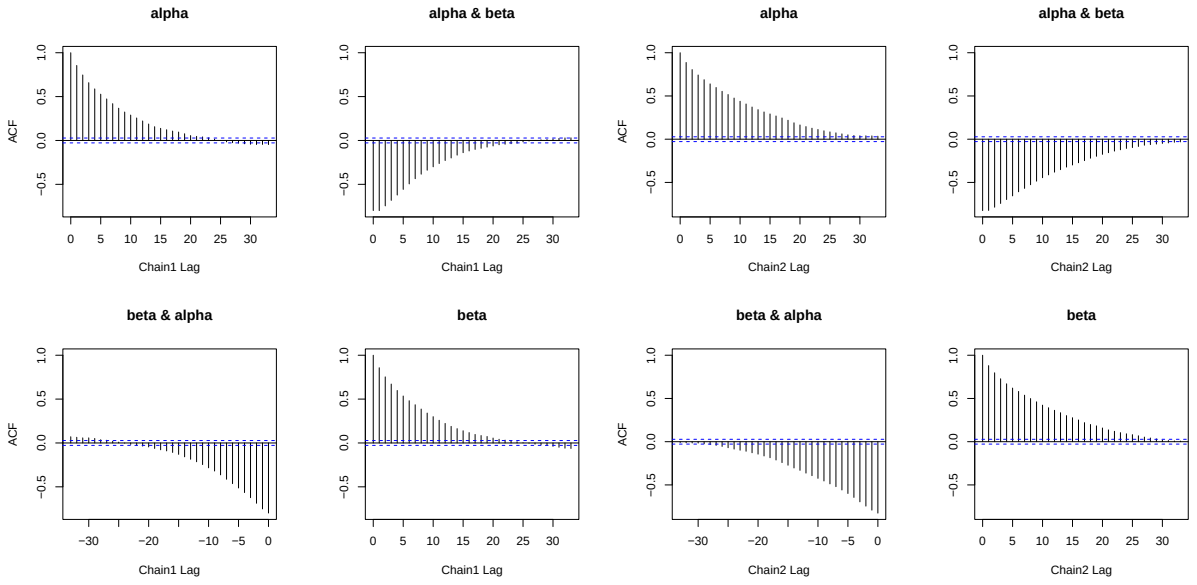

Figure 5: Autocorrelation plots of α and β (prior: t-distribution)

We analyzed the convergence of the model with a prior t-distribution using trace plots, autocorrelation plots, and Gelman-Rubin diagnostic plots, which are presented in Figure 4 and Figure 5. We observed that all the trends remained consistent with the plots obtained using a normal distribution for α and β. Moreover, the estimated potential scale reduction factors of α and β were found to be equal to 1, indicating good convergence of the model. These results provide compelling evidence that the model with a prior t-distribution has achieved good convergence

Table 1: Bayesian posterior measures of α and β

	Mean	Median	St.Dev	HPD Interval
N(0, 10000)				
alpha	-1.7931153	-1.7912688	0.1253469	[-2.052, -1.568]
beta	0.0183311	0.0182931	0.0010990	[0.016, 0.02]
t(0, 0.0001, 5)				
alpha.1	-1.7945744	-1.7922911	0.1326002	[-2.059, -1.537]
beta.1	0.0183608	0.0183749	0.0011560	[0.016, 0.02]

and that the inference derived from the Markov chain Monte Carlo simulation is reliable.

(2) Summarize all results graphically and summarize with the usual Bayesian posterior measures. What do you conclude from these?

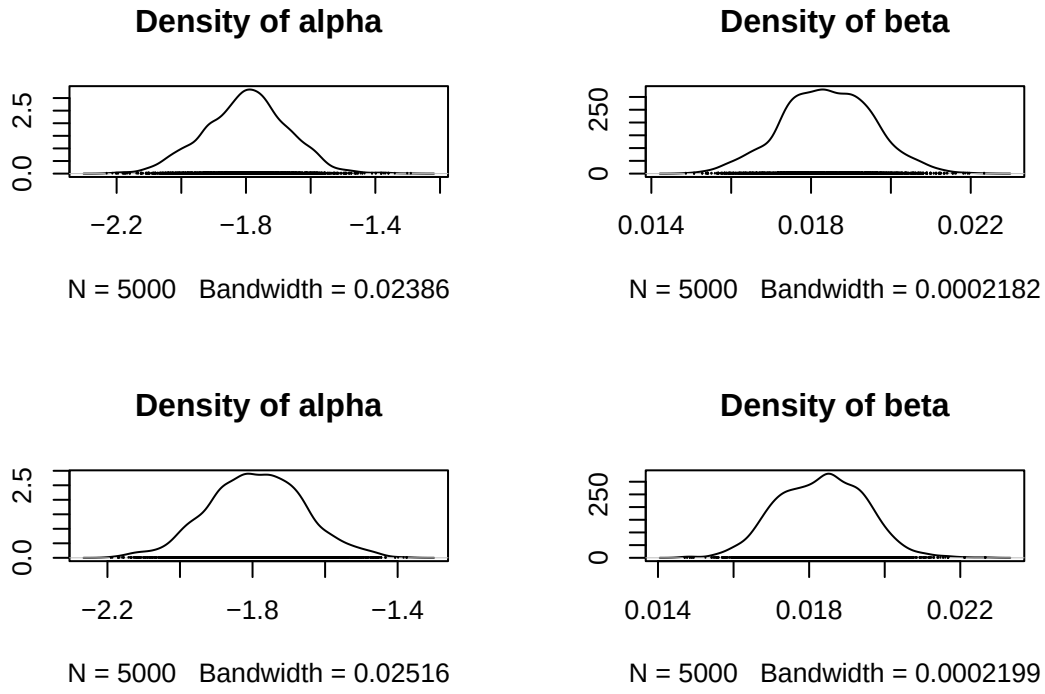

Figure 6: Density plots of α and β with normal distribution (top) and t distribution (bottom).

The density plots with a normal distributed prior and a t distributed prior, as presented in Figure 6. The plots for both α and β display smooth distributions with similar mean values, suggesting that both priors lead to similar approximations of the true posterior distribution.

Based on the Bayesian posterior measures of α and β (Table 1), it can be concluded that the probability of malformations is ?????0.143????? in the absence of any administered dose. As the dose increases, the probability of malformations also increases, indicating a positive dose-response relationship. Additionally, the 95% highest posterior density (HPD) interval, which captures the 95% most plausible parameter values, does not include the value of 0, providing evidence for the existence of a dose effect.

Table 2: Posterior measures of BMD.

BMD (Mean)	HPD interval	Prior
Normal distribution	17.191	[15.25, 19.153]
t-distribution	17.186	[15.279, 19.26]

(3) Plot the posterior dose-response relationship together with the observed probabilities of a malformation per dose.

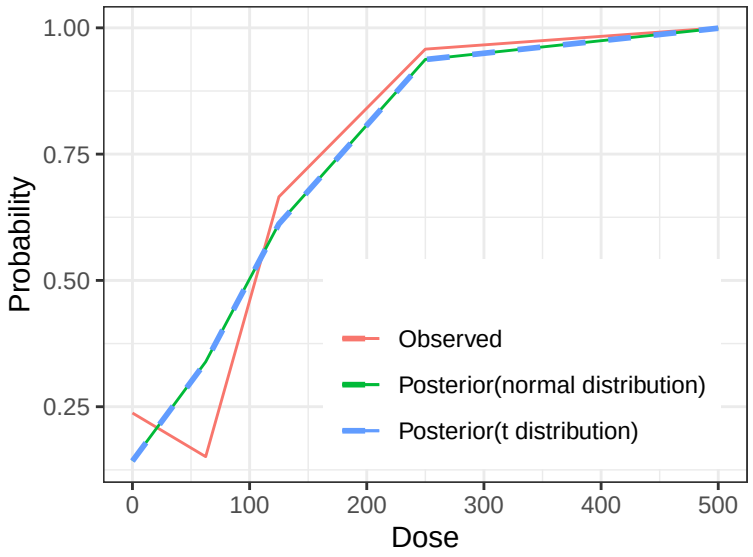

Figure 7: Posterior dose-response relationship and observed probabilities.

Figure 7 indicates that the trends in the posteriors exhibit a dose-response relationship that is similar to the observed probabilities, with the exception of the dose range of 50-70. Moreover, the posterior with t-distributed priors closely aligns with the posterior with normal distributed priors. In general, all three lines demonstrate an increasing probability of malformations as the dose increases, with this increase leveling off as the dose approaches approximately 400.

(4) A safe level of exposure can be defined as a dose corresponding to a very small increase in excess risk of q, e.g. q = 0.05. This is called the Benchmark dose (BMD) d^* and can be obtained by solving the equation

$$r(d^*) = \frac{P(d^*) - P(0)}{1 - P(0)} = q$$

with $P(d)$ the probability of an adverse effect at dose level d. For a logistic regression with a linear dose model, the BMD is given by

$$BMD = \frac{logit(q^*) - \alpha}{\beta}$$

with $q^* = q(1 - P(0)) + P(0)$. Determine the posterior estimate of the safe level of exposure for DYME corresponding with an excess risk of $q = 0.05$.

The posterior mean values for BMD and the corresponding HPD interval are shown in Table 2. The lower bounds of the HPD interval for the prior normal distribution (19.153 for Normal distribution and 19.26 for t-distribution) should be considered as the Benchmark does.

(5) As an alternative, a safe level of exposure can be obtained from a threshold model, defined as

$$y \sim binomial(N, \pi)$$

$$logit(\pi) = \alpha + \beta(d - \tau)I(d > \tau)$$

, with τ the threshold dose below which there is no excess risk. Write code for this model, and summarize the results. How do these results compare with previous results?

In this threshold model, we essentially fit a piecewise linear regression for $logit(\pi)$:

- if the dose is smaller than τ, $I(d < \tau) = 0$ meaning the intercept is α and the slope is zero.
- if the dose is larger than τ, $I(d < \tau) = 1$ meaning the intercept is $\alpha - \beta\tau$ and the slope is β

We will use the same vague priors from a Normal distribution:

$$\alpha \sim N(0, 10000)$$

$$\beta \sim N(0, 10000)$$

All other tuning parameter such as initial values and burn-in are chosen the same as in the previous model. We will test several models with $\tau \in \{0, 62.5, 125, 250, 500\}$. We will select the model that minimises the WAIC.

Figure 8 shows that the lowest WAIC is reached when $\tau = 62.5$. This means that, for doses ≤ 62.5, $logit(\pi) = \alpha$, independent of the dose. For doses > 62.5, $logit(\pi) = \alpha + \beta(d - 62.5)$.

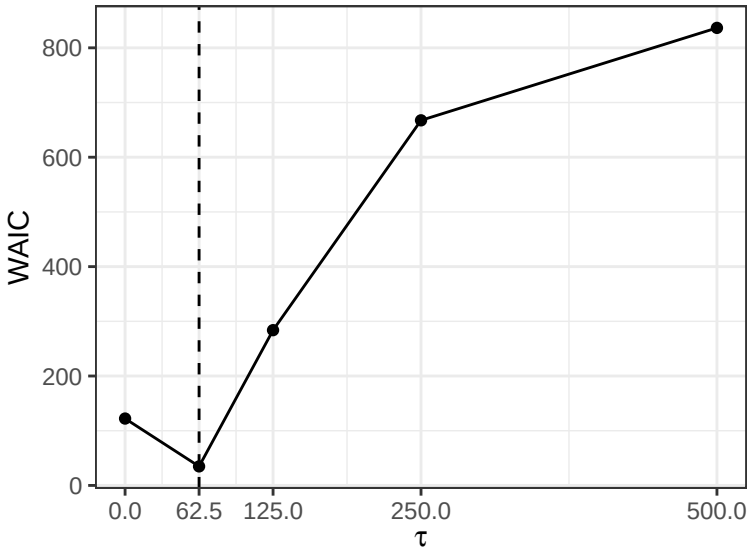

Figure 8: WAIC for each value of τ.

For the model with the lowest WAIC, we observe from the trace plots of α and β (refer to Figure 9) that the estimates from each chain quickly stabilized around a steady state. Additionally, both chains were found to converge around the same conclusion. The Gelman-Rubin diagnostic test showed that the estimated potential scale reduction factors of α and β were both 1. These results suggest that our model has converged well. Moreover, the Gelman-Rubin diagnostic plots (Figure 9) support this conclusion, as both the potential scale reduction factors of α and β were found to decrease quickly and remain stable as the number of iterations increased. We also examined the autocorrelation plots (refer to Figure 10), which indicated low autocorrelation. The autocorrelation decreased and remained around zero as the lag number increased, indicating that the chains have mixed well. Overall, these diagnostic tools suggest that our model that the has converged well, and the inference based on the Markov chain Monte Carlo simulation is reliable.

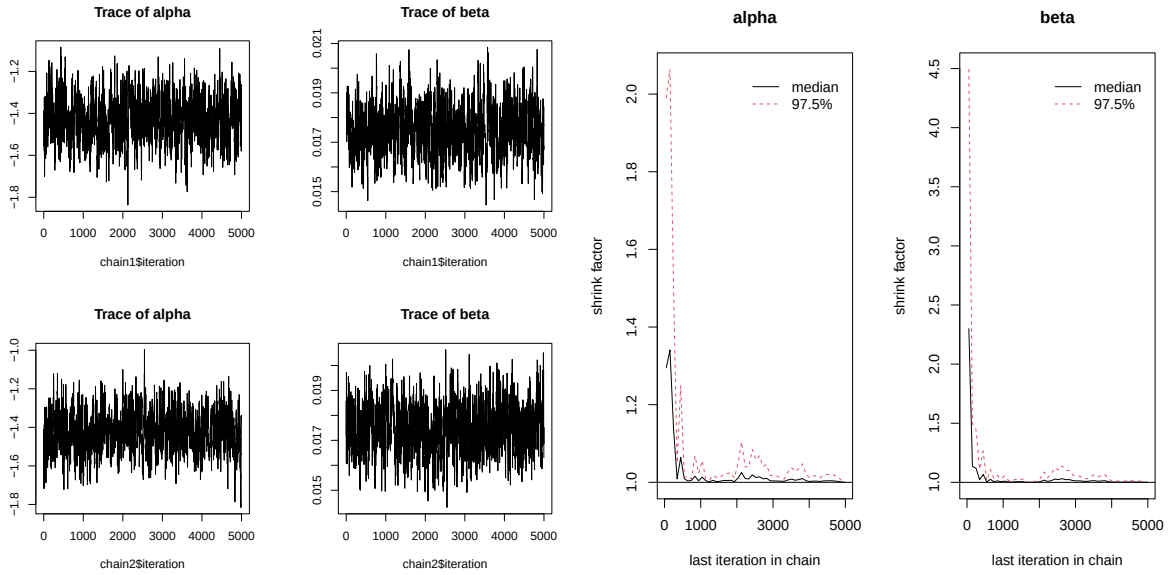

Figure 9: Trace and Gelman-Rubin diagnostic plots of α and β for the threshold model with the lowest WAIC.

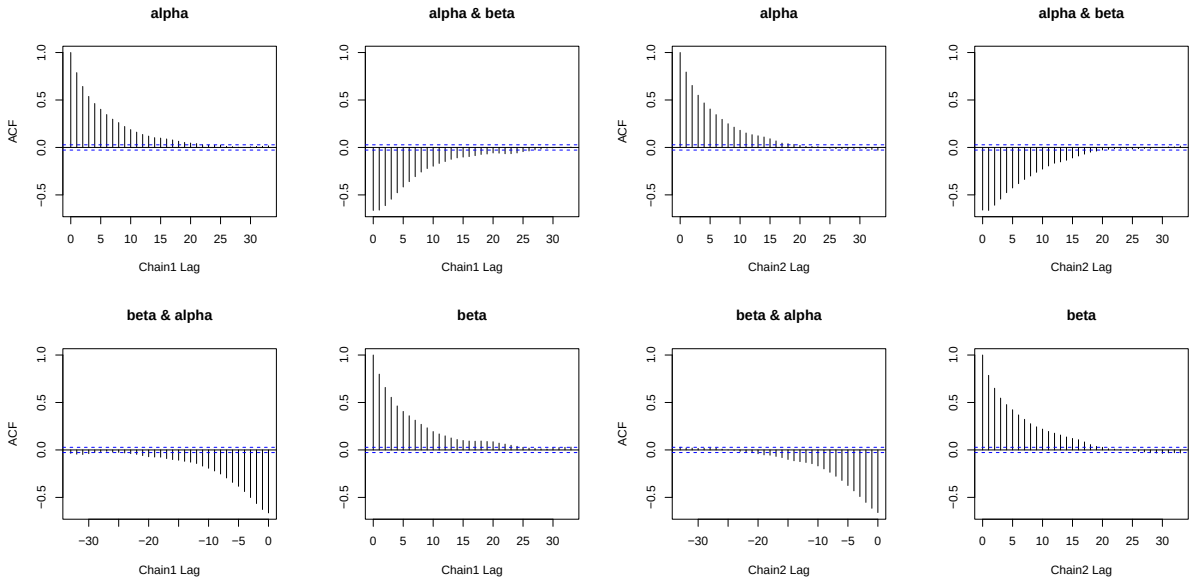

Figure 10: Autocorrelation plots of α and β for the threshold model with the lowest WAIC.

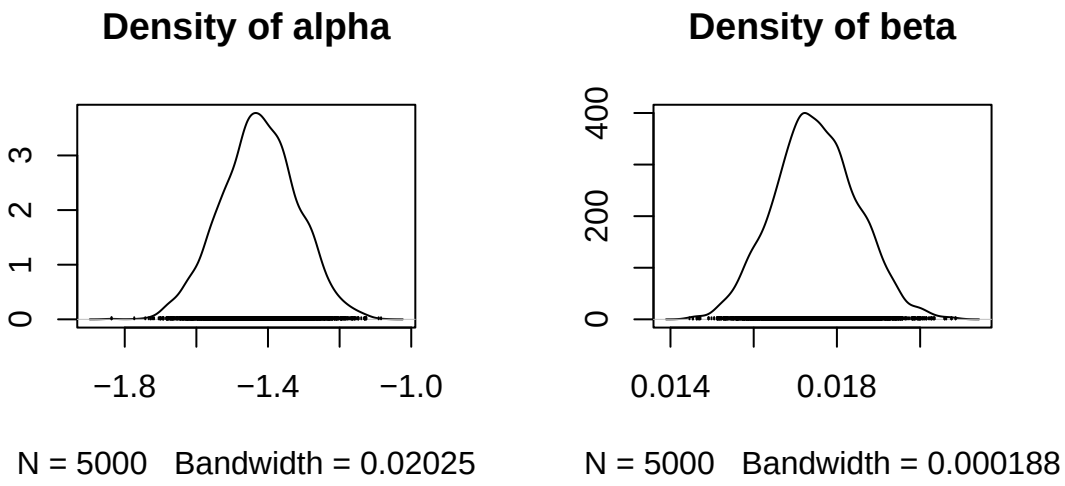

Figure 11: Density plots of α and β for the threshold model with the lowest WAIC.

Table 3: Bayesian posterior measures of α and β

	Mean	Median	St.Dev	HPD Interval
alpha	-1.4220679	-1.422377	0.106187	[-1.631, -1.219]
beta	0.0174669	0.017468	0.000989	[0.016, 0.019]

The density plots are presented in Figure 11. The plots for both α and β display smooth distributions. Table 3 summarizes the posterior measures of α and β.

Figure 12 shows that the fit of the posterior dose-response relationship with the observed probabilities improved combared to Figure 7, especially in the range with lower doses.

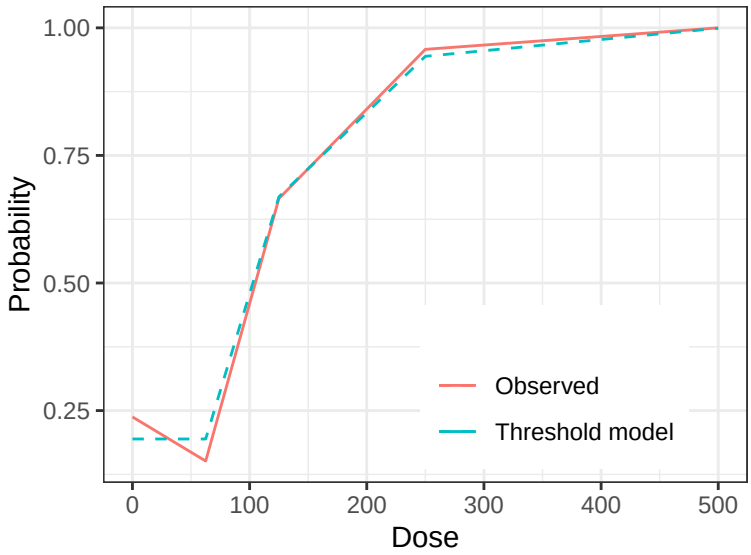

Figure 12: Posterior dose-response relationship and observed probabilities for the model with the lowest WAIC.

A Appendix

A.1 All code for the report

```
#---------------------------load all packages----------------------------------#
library(tidyverse)
library(rprojroot)
library(kableExtra)
library('nimble')
library(coda)
library(cowplot)
library(latex2exp)

knitr::opts_chunk$set(echo = TRUE)
knitr::opts_knit$set(
  root.dir = find_root(criterion = has_file("BayesianInference.Rproj")))
#-------------------------A3: posterior relative risk--------------------------#
#set the parameters
set.seed(2023)
n <- 10000
alpha_ns <- 118
beta_ns <- 4321
alpha_smoker <- 172
beta_smoker <- 3265
#draw random values from both posteriors ans calculate relative risk
random_smoker <- rbeta(n, alpha_smoker, beta_smoker)
random_ns <- rbeta(n, alpha_ns, beta_ns)
random_rr <- random_smoker / random_ns

d <- data.frame(smoker = random_smoker,
                non_smoker = random_ns,
                relative_risk = random_rr)
#2.5 quantile lowest, median and 97.5%
quantiles <- round(quantile(random_rr, probs = c(0.025, 0.5, 0.975)),3)


ggplot(d) +
  geom_histogram(aes(x = relative_risk),
                 color = "black", fill = "transparent") +
  theme_bw() +
  xlab("Relative risk")
#------------------------------setup task B--------------------------------#
n <- 5
# Dose level
dose <- c(0,62.5,125,250,500)
# Number of fetus
N <- c(282,225,290,261,141)
# Number of malformations
y <- c(67,34,193,250,141)
#--------------------------------Question B1---------------------------------#
init1 <- list(alpha = 0, beta = 0)
init2 <- list(alpha = -0.5, beta = 0.1)
initial.values <- list(init1, init2)
```

```r
# MCMC settings
n.iter <- 10000  # iterations
n.burnin <- 5000 # burn-in
n.chains <- 2    # chains

# Model settings
model.data <- list('dose' = dose, 'N' = N,'y' = y)
model.constant <- list('n' = n)

# Model 1
# A prior with Normal Distribution
model_1 <- nimbleCode({
  # Specify a vague prior with normal distribution
  alpha ~ dnorm(0, sd = 10000)
  beta ~ dnorm(0, sd = 10000)
  # likelihood
  for (i in 1:n) {
    logit(p[i]) <-  alpha + beta * dose[i]
    y[i] ~ dbin(p[i],N[i])
  }})

# Output of Model 1
mcmc.output1 <- nimbleMCMC(code = model_1,
                           data = model.data,
                           constants = model.constant,
                           inits = initial.values,
                           niter = n.iter,
                           nburnin = n.burnin,
                           summary = TRUE,
                           nchains = n.chains
)
# Trace plots
pdf("images/trace_normal.pdf")
par(mfrow = c(2,2))
traceplot(as.mcmc(mcmc.output1$samples$chain1), xlab = "chain1$iteration")
traceplot(as.mcmc(mcmc.output1$samples$chain2), xlab = "chain2$iteration")
dev.off()
# Gelman-Rubin diagnostic plots
combinedchains1 = mcmc.list(as.mcmc(mcmc.output1$samples$chain1), as.mcmc(mcmc.output1$samples$chain2))
gelman.diag(combinedchains1)
pdf("images/gelman_normal.pdf")
gelman.plot(combinedchains1,xlim = c(0,5000))
dev.off()
# Autocorrelation plots
par(mfrow = c(2, 2))
#densplot(as.mcmc(mcmc.output1$samples$chain1), main = "Chain1 Density of alpha")
#densplot(as.mcmc(mcmc.output1$samples$chain2), main = "Chain2 Density of alpha")
pdf("images/ac1_normal.pdf")
acf(as.mcmc(mcmc.output1$samples$chain1), xlab = 'Chain1 Lag')
dev.off()
pdf("images/ac2_normal.pdf")
acf(as.mcmc(mcmc.output1$samples$chain2), xlab = 'Chain2 Lag')
dev.off()
# Model 2
```

```r
# A prior with t Distribution
model_2 <- nimbleCode({
  # Specify a vague prior with t distribution
  alpha ~ dt(0, 0.0001, 5)
  beta ~ dt(0, 0.0001, 5)
  # likelihood
  for (i in 1:n) {
    logit(p[i]) <-  alpha + beta * dose[i]
    y[i] ~ dbin(p[i],N[i])
  }})

# Output of Model 1
mcmc.output2 <- nimbleMCMC(code = model_2,
                           data = model.data,
                           constants = model.constant,
                           inits = initial.values,
                           niter = n.iter,
                           nburnin = n.burnin,
                           summary = TRUE,
                           nchains = n.chains
)

# Trace plots
pdf("images/trace_t.pdf")
par(mfrow = c(2,2))
traceplot(as.mcmc(mcmc.output2$samples$chain1), xlab = "chain1$iteration")
traceplot(as.mcmc(mcmc.output2$samples$chain2), xlab = "chain2$iteration")
dev.off()
# Autocorrelation plots
par(mfrow = c(2,2))
densplot(as.mcmc(mcmc.output2$samples$chain1), main = "Chain1 Density of alpha")
densplot(as.mcmc(mcmc.output2$samples$chain2), main = "Chain2 Density of alpha")

pdf("images/ac1_t.pdf")
acf(as.mcmc(mcmc.output2$samples$chain1), xlab = 'Chain1 Lag')
dev.off()
pdf("images/ac2_t.pdf")
acf(as.mcmc(mcmc.output2$samples$chain2), xlab = 'Chain2 Lag')
dev.off()
# Gelman-Rubin diagnostic plots
combinedchains2 = mcmc.list(as.mcmc(mcmc.output2$samples$chain1), as.mcmc(mcmc.output2$samples$chain2))
gelman.diag(combinedchains2)
pdf("images/gelman_t.pdf")
gelman.plot(combinedchains2,xlim = c(0,5000))
dev.off()
#-------------------------------Question B2-----------------------------------#
par(mfrow = c(2,2))
densplot(as.mcmc(mcmc.output1$samples$chain1,mcmc.output1$samples$chain2))
densplot(as.mcmc(mcmc.output2$samples$chain1,mcmc.output2$samples$chain2))
# Normal distribution
#mcmc.output1$summary
samples_n <- rbind(mcmc.output1$samples$chain1,mcmc.output1$samples$chain2)
HPD1 <- as.data.frame(round(HPDinterval(as.mcmc(samples_n)),3)) %>%
  mutate(interval = paste0("[", lower, ", ", upper, "]"))
```

```r
# t distribution
#mcmc.output2$summary
samples_t <- rbind(mcmc.output2$samples$chain1,mcmc.output2$samples$chain2)
HPD2 <- as.data.frame(round(HPDinterval(as.mcmc(samples_t)),3)) %>%
  mutate(interval = paste0("[", lower, ", ", upper, "]"))

as.data.frame(rbind(mcmc.output1$summary$all.chains[, -c(4, 5)],
                    mcmc.output2$summary$all.chains[, -c(4, 5)])) %>%
  mutate(HPD = c(HPD1$interval, HPD2$interval)) %>%
  kable(booktabs = TRUE,
        caption = "Bayesian posterior measures of  and ",
        col.names = c("Mean", "Median", "St.Dev", "HPD Interval")) %>%
  kableExtra::group_rows(group_label = "N(0, 10000)",
                         start_row = 1, end_row = 2) %>%
  kableExtra::group_rows(group_label = "t(0, 0.0001, 5)",
                         start_row = 3, end_row = 4)
#------------------------------Question B3----------------------------------#
# posterior with normal distribution
chains_output1 <- data.frame(mcmc.output1[[1]])
chain1_output1 <- chains_output1[, 1:2] %>%
  rename("alpha" = "chain1.alpha", "beta" = "chain1.beta")
chain2_output1 <- chains_output1[, 3:4] %>%
  rename("alpha" = "chain2.alpha", "beta" = "chain2.beta")
df_output1 <- rbind(chain1_output1, chain2_output1)
# get the mean value of alpha and beta
alpha_n <- round(mean(df_output1$alpha), 3)
beta_n <- round(mean(df_output1$beta), 3)

# posterior with t distribution
chains_output2 <- data.frame(mcmc.output2[[1]])
chain1_output2 <- chains_output2[, 1:2] %>%
  rename("alpha" = "chain1.alpha", "beta" = "chain1.beta")
chain2_output2 <- chains_output2[,3:4] %>%
  rename("alpha" = "chain2.alpha", "beta" = "chain2.beta")
df_output2 <- rbind(chain1_output2, chain2_output2)
# get the mean value of alpha and beta
alpha_t <- round(mean(df_output2$alpha), 3)
beta_t <- round(mean(df_output2$beta), 3)

# Create a dataframe for plotting
df_plotting <- data.frame(cbind(dose,N,y)) %>%
  mutate(prob_observed = y/N,# Observed
         prob_n = expit(alpha_n + beta_n * dose),# Posterior with normal distr
         prob_t = expit(alpha_t + beta_t * dose)# Posterior with t distr
         )

ggplot(df_plotting, aes(x = dose)) +
  geom_line(aes(y = prob_observed, color = "Observed")) +
  geom_line(aes(y = prob_n, color = "Posterior(normal distribution)")) +
  geom_line(aes(y = prob_t, color = "Posterior(t distribution)"),
            linetype = "dashed", linewidth = 1) +
  labs(x = "Dose", y = "Probability", color = "") +
  theme_bw() +
```

```r
  theme(legend.position = c(0.65, 0.25))
#-------------------------------Question B4----------------------------------#
# Excess risk q=0.05
# prior Normal distribution
df_output1_bmd <- df_output1 %>%
  # Caculate BMD
  mutate(P0 = exp(alpha) / (1 + exp(alpha))) %>%
  mutate(q.star = (0.05 * (1 - P0)) + P0) %>%
  mutate(bmd = (logit(q.star) - alpha) / beta)

# HPD Interval
hpd_n <- as.data.frame(round(HPDinterval(as.mcmc(df_output1_bmd)), 3)) %>%
  mutate(interval = paste0("[", lower, ", ", upper, "]"))

# prior t distribution
df_output2_bmd <- df_output2 %>%
  # Caculate BMD
  mutate(P0 = exp(alpha) / (1 + exp(alpha))) %>%
  mutate(q.star = (0.05 * (1 - P0)) + P0) %>%
  mutate(bmd = (logit(q.star) - alpha) / beta)
# HPD Interval
hpd_t <- as.data.frame(round(HPDinterval(as.mcmc(df_output2_bmd)), 3)) %>%
  mutate(interval = paste0("[", lower, ", ", upper, "]"))

data.frame(Prior = c("Normal distribution", "t-distribution"),
           mean = c(round(mean(df_output1_bmd$bmd), 3),
                    round(mean(df_output2_bmd$bmd), 3)),
           hpd = c(hpd_n[5, "interval"], hpd_t[5, "interval"])) %>%
  kable(booktabs = TRUE,
        col.names = c("BMD (Mean)", "HPD interval", "Prior"),
        caption = "Posterior measures of BMD.") %>%
  kableExtra::kable_styling()
#-------------------------Question B-----------------------------------------#
#possibly interesting reference for piecewise linear regression in R with nimble
#https://gkonstantinoudis.github.io/nimble/PiecewiseLinear.html
init1 <- list(alpha = 0, beta = 0)
init2 <- list(alpha = -0.5, beta = 0.1)
initial.values <- list(init1, init2)

# MCMC settings
n.iter <- 10000  # iterations
n.burnin <- 5000 # burn-in
n.chains <- 2    # chains


testmodel <- function(tau = 0){
  # Model settings
  indicator <- 1*(dose > tau)
  model.data <- list('dose' = dose, 'N' = N,'y' = y, 'indicator' = indicator)
  model.constant <- list('n' = n)

  # Model 1
  # A prior with Normal Distribution
  model <- nimbleCode({
```

```
    # Specify a vague prior with normal distribution
    alpha ~ dnorm(0, sd = 10000)
    beta ~ dnorm(0, sd = 10000)
    # likelihood
    for (i in 1:n) {
      logit(p[i]) <-  alpha + beta * dose[i] * indicator[i]
      y[i] ~ dbin(p[i],N[i])
    }})

  # Output of Model 1
  mcmc.output <- nimbleMCMC(code = model,
                            data = model.data,
                            constants = model.constant,
                            inits = initial.values,
                            niter = n.iter,
                            nburnin = n.burnin,
                            summary = TRUE,
                            nchains = n.chains,
                            WAIC = TRUE
  )
  return(mcmc.output)
}
tau_values <- c(0,62.5,125,250,500)
results <- map(tau_values, testmodel)
save(results, file = "output/results_threshold_model.Rdata")
waic <- sapply(X = seq(length(tau_values)),
      FUN = function(x) {results[[x]]$WAIC$WAIC})
#plot the relationship between WAIC and tau
data.frame(waic = waic,
           tau = tau_values) %>%
  ggplot(aes(x = tau, y = waic)) +
  geom_point() +
  geom_line() +
  theme_bw() +
  ylab("WAIC") + xlab(TeX("$\\tau$")) +
  geom_vline(aes(xintercept = tau_values[which(waic == min(waic))]),
             linetype = "dashed") +
  scale_x_continuous(breaks = tau_values)

chosen_model <- results[[which(waic == min(waic))]]
# Trace plots
pdf("images/trace_threshold.pdf")
par(mfrow = c(2,2))
traceplot(as.mcmc(chosen_model$samples$chain1), xlab = "chain1$iteration")
traceplot(as.mcmc(chosen_model$samples$chain2), xlab = "chain2$iteration")
dev.off()
# Gelman-Rubin diagnostic plots
combinedchains1 = mcmc.list(as.mcmc(chosen_model$samples$chain1),
                            as.mcmc(chosen_model$samples$chain2))
gelman.diag(combinedchains1)
pdf("images/gelman_threshold.pdf")
gelman.plot(combinedchains1,xlim = c(0,5000))
dev.off()
# Autocorrelation plots
```

```r
par(mfrow = c(2, 2))
pdf("images/ac1_threshold.pdf")
acf(as.mcmc(chosen_model$samples$chain1), xlab = 'Chain1 Lag')
dev.off()
pdf("images/ac2_threshold.pdf")
acf(as.mcmc(chosen_model$samples$chain2), xlab = 'Chain2 Lag')
dev.off()
par(mfrow = c(1,2))
densplot(as.mcmc(chosen_model$samples$chain1,chosen_model$samples$chain2))
samples_n <- rbind(chosen_model$samples$chain1,chosen_model$samples$chain2)
HPD1 <- as.data.frame(round(HPDinterval(as.mcmc(samples_n)),3)) %>%
  mutate(interval = paste0("[", lower, ", ", upper, "]"))

as.data.frame(chosen_model$summary$all.chains[, -c(4, 5)]) %>%
  mutate(HPD = HPD1$interval) %>%
  kable(booktabs = TRUE,
        caption = "Bayesian posterior measures of   and  ",
        col.names = c("Mean", "Median", "St.Dev", "HPD Interval")) %>%
  kableExtra::kable_styling()
#plot posterior vs observed probabilities
chains_output1 <- data.frame(chosen_model[[1]])
chain1_output1 <- chains_output1[, 1:2] %>%
  rename("alpha" = "chain1.alpha", "beta" = "chain1.beta")
chain2_output1 <- chains_output1[, 3:4] %>%
  rename("alpha" = "chain2.alpha", "beta" = "chain2.beta")
df_output1 <- rbind(chain1_output1, chain2_output1)
# get the mean value of alpha and beta
alpha_n <- round(mean(df_output1$alpha), 3)
beta_n <- round(mean(df_output1$beta), 3)

# Create a dataframe for plotting
df_plotting <- data.frame(cbind(dose, N, y)) %>%
  mutate(indicator = 1*(dose > tau_values[which(waic == min(waic))]),
         prob_observed = y/N,
         threshold_model = expit(alpha_n + beta_n * dose * indicator)
         )
ggplot(df_plotting, aes(x = dose)) +
  geom_line(aes(y = prob_observed, color = "Observed")) +
  geom_line(aes(y = threshold_model, color = "Threshold model"), linetype = "dashed") +
  theme_bw() +
  theme(legend.position = c(0.7, 0.2)) +
  scale_color_discrete("") +
  ylab("Probability") +
  xlab("Dose")
```

Bibliography

Kerman, Jouni. 2011. "Neutral Noninformative and Informative Conjugate Beta and Gamma Prior Distributions."

Tuyl, Frank, Richard Gerlach, and Kerrie Mengersen. 2008. "A Comparison of Bayes–Laplace, Jeffreys, and Other Priors: The Case of Zero Events." *The American Statistician* 62 (1): 40–44.