UNIVERSITY OF CAPE TOWN
IYUNIVESITHI YASEKAPA • UNIVERSITEIT VAN KAAPSTAD

Bayesian Decision Modelling

Assignment 2

Computational Assignment

Julian Albert
ALBJUL005

Department of Statistical Sciences
University of Cape Town
South Africa

June 11, 2019

Contents

Question 1

In this Question we are concerned with a variation of a hierarchical Bayes problem in which we have a parameter vector $\vec{\theta} = [\theta_1, \theta_2, \theta_3]'$ with common prior λ and data $\boldsymbol{X}$.

We are told that the $(n = 3)$ observations, X_1, X_2, X_3 are independently exponentially distributed and that the θ_i some common density, thus;

$$f(x_i|\theta_i) = \theta_i e^{(-\theta_i x_i)} \qquad \text{for} \qquad x_i > 0 \tag{1}$$

$$\pi(\theta_i|\lambda) = \lambda^2 \theta_i e^{(-\lambda\theta_i)} \qquad \text{for} \qquad \lambda > 0 \tag{2}$$

$$\pi(\lambda) \propto \lambda^{-1} \tag{3}$$

a)

First I will find the joint posterior on $\theta_1, \theta_2, \theta_3$ defined as

$$\pi(\theta_1, \theta_2, \theta_3|\cdot) \tag{4}$$

To find this I specify the likelihood for the data data and the posterior distribution of the parameters defined by

$$\pi(\theta_1, \theta_2, \theta_3, \lambda|\cdot) \propto L(\vec{\theta}; x)\pi(\vec{\theta}\ |\lambda)\pi(\lambda) \tag{5}$$

$$\propto L(\vec{\theta}; x)\prod_{i=1}^{3}\pi(\theta_i|\lambda)\pi(\lambda) \tag{6}$$

$$\propto \prod_{i=1}^{3}(\theta_i)^2 \exp\left[-\sum_{i=1}^{3}\theta_i x_i\right] \exp\left[-\lambda\sum_{i=1}^{3}\theta_i\right] \lambda^6\lambda^{-1} \tag{7}$$

$$\propto \prod_{i=1}^{3}(\theta_i)^2 \exp\left[-\sum_{i=1}^{3}\theta_i(x_i+\lambda)\right] \lambda^5 \tag{8}$$

To find $\pi(\vec{\theta}\ |\cdot)$ we need to integrate out the λ such that

$$\pi(\vec{\theta}\ |\cdot) \propto \int_\lambda \prod_{i=1}^{3}(\theta_i)^2 \exp\left[-\sum_{i=1}^{3}\theta_i(x_i+\lambda)\right] \lambda^5 \mathrm{d}\lambda$$

$$\propto \prod_{i=1}^{3}(\theta_i)^2 \exp\left[-\sum_{i=1}^{3}\theta_i x_i\right] \underbrace{\int_\lambda \lambda^5 \exp\left[-\lambda\sum_{i=1}^{3}\theta_i\right] \mathrm{d}\lambda}_{\text{Gamma}\left(6, \sum_{i=1}^{3}\theta_i\right)}$$

$$\propto \prod_{i=1}^{3}(\theta_i)^2 \exp\left[-\sum_{i=1}^{3}\theta_i x_i\right] \frac{\Gamma(6)}{\left(\sum_{i=1}^{3}\theta_i\right)^6}$$

$$\propto \frac{\prod_{i=1}^{3}(\theta_i)^2 \exp\left[-\sum_{i=1}^{3}\theta_i x_i\right]}{\left(\sum_{i=1}^{3}\theta_i\right)^6}$$

b)

Now we will find the functional form (i.e. ignoring integration constants) for the conditional posterior density

$$\pi(\theta_i|\vec{\theta}_j = \vec{\theta}_j^{\,0}) \qquad \text{for} \qquad j \neq i \tag{9}$$

To do this we consider the result from the preceding section and let $\pi(\theta_i|\vec{\theta}_j) = \pi(\theta_1|\theta_2, \theta_3)$. We can then drop θ_2, θ_3 in the numerator as they represent proportionality constants to get

$$\pi(\theta_1|\theta_2, \theta_3, \cdot) \propto \frac{\theta_1^2 e^{-\theta_1 X_1}}{\left(\sum_{i=1}^{3} \theta_i\right)^6} \tag{10}$$

We can easily deduce that the numerator follows a Gamma(3, X_1). The question states that this conditional density can be expressed as as the product of a gamma and a Pareto density. Thus we have that a Pareto must be represented by the denominator. To confirm this I will find the integrating constant by evaluating

$$\int_0^{\infty} \frac{1}{(\theta_1 + \theta_2 + \theta_3)^6} \mathrm{d}\theta_1$$

Let $u = \theta_1 + \theta_2 + \theta_3$ such that $\mathrm{d}u = \mathrm{d}\theta_1$ therefore the inverse of the integrating constant is

$$\begin{aligned}
\int_{(\theta_2+\theta_3)}^{\infty} \frac{1}{u^6} \mathrm{d}u &= -\frac{1}{5u^5}\Bigg|_{(\theta_2+\theta_3)}^{\infty} \\
&= \frac{1}{5(\theta_2 + \theta_3)^5}
\end{aligned}$$

Thus our conditional density can be expressed as as the product of a Gamma(3, X_i) and a Pareto(5, $\vec{\theta}_j$) density.

c)

To generate values from the Pareto distribution we can form the cumulative distribution function and make use of the probability inverse transformation.

$$\begin{aligned}
\int_0^{\theta_i} \frac{5(\vec{\theta}_j)^5}{(t + \vec{\theta}_j)^6} \mathrm{d}t &= 5(\vec{\theta}_j)^5 \left\{ -\frac{1}{5(t + \vec{\theta}_j)^5}\Bigg|_0^{\theta_i} \right\} \\
&= 5(\vec{\theta}_j)^5 \left\{ -\frac{1}{5(\theta_i + \vec{\theta}_j)^5} + \frac{1}{5(\vec{\theta}_j)^5} \right\} \\
&= 1 - \frac{\vec{\theta}_j^{\,5}}{(\theta_i + \vec{\theta}_j)^5}
\end{aligned}$$

Now set this equal to u and solve out

$$1 - \frac{\vec{\theta}_j^{\,5}}{(\theta_i + \vec{\theta}_j)^5} = u \tag{11}$$

$$\frac{\vec{\theta}_j}{(\theta_i + \vec{\theta}_j)} = \sqrt[5]{1-u} \tag{12}$$

$$\theta_i = \frac{\vec{\theta}_j}{\sqrt[5]{1-u}} - \vec{\theta}_j \tag{13}$$

To generate the $\theta_i' s$ from the conditional we can set up an acceptance/rejection sampling algorithm defined by

$$\text{Decision} \;=\begin{cases} \text{Reject,} & \dfrac{f(x)}{ch(x)} < u \\ \text{Accept,} & \dfrac{f(x)}{ch(x)} > u \end{cases}$$

Where

$$\frac{f(x)}{ch(x)} = \frac{\left\{\dfrac{\left(X_i^3 \theta_i^2 e^{-\theta_i X_i}\right)}{\Gamma(3)}\right\}\left\{\dfrac{\left(5(\vec{\theta}_j)^5\right)}{\left(\theta_i + \vec{\theta}_j\right)^6}\right\}}{c^1\left\{\dfrac{\left(5(\vec{\theta}_j)^5\right)}{\left(\theta_i + \vec{\theta}_j\right)^6}\right\}} \tag{14}$$

$$= \underbrace{\frac{X_i^3 \theta_i^2 \exp\left[-\theta_i X_i\right]}{\Gamma(3)}}_{\text{Gamma}(3, X_i)} \times \frac{2}{X_i} \tag{15}$$

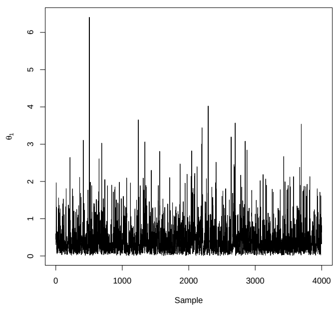
To obtain estimates of the posterior mean and standard error of each θ_i we need to sample from the posterior distributions for each θ_i, to do this we can utilise the following flow

Get $\theta_i \vert \vec{\theta}_j$ from Equation 13 $\rightarrow$ sub into Equation 15 $\rightarrow$ Update θ_i with accepted value.

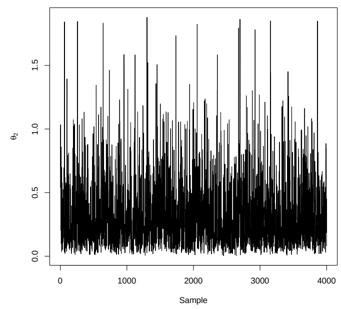
Using all 4000 samples we can plot the trace plots for our $\vec{\theta}$, this will allow us to visually determine a suitable burn-in period. The burn-in period is the period from K to N where parameter samples $\vec{\theta}^{(t \geq K)}$ are those sampled from the stationary distribution representative of the true posterior.

[1] Where c is taken to be the max of some objective, conveniently the maximum of a gamma distribution occurs at the mode, thus we can evaluate the constant to be the mode of a gamma given by $c = \dfrac{(\alpha - 1)}{\beta} = \dfrac{2}{X_i}$

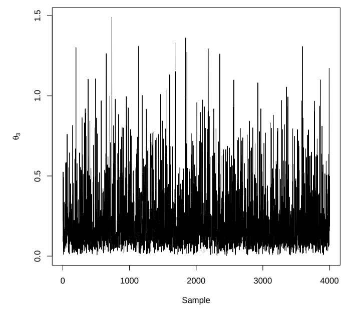
From Figure (1) we can see that the sampled values become more stable at around 2000 samples, as such a burn-in of 50% is used, we can see that the range of θ values is less subject to extreme values. Alternatively we could use the Gelman and Rubin diagnostic to assess stationarity, however, burn-in of 50% can be deemed sufficient in most cases.



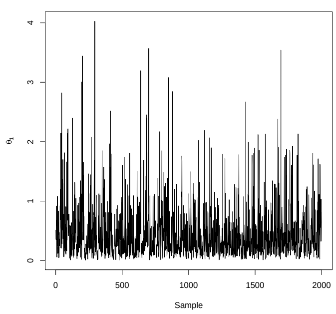
(a) θ_1



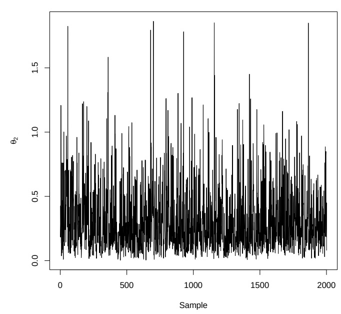
(b) θ_2



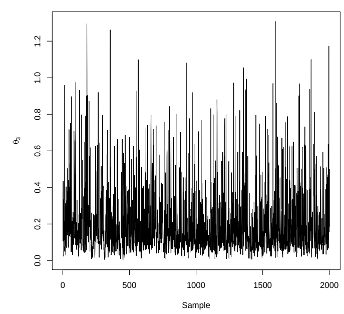
(c) θ_3



(d) θ_1



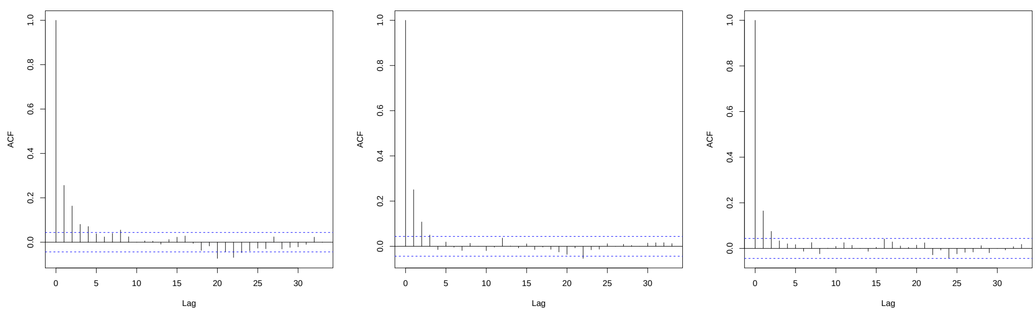
(e) θ_2



(f) θ_3

Figure 1: Top: Trace Plots for Full Samples
Bottom: Trace Plots Post Burn-in

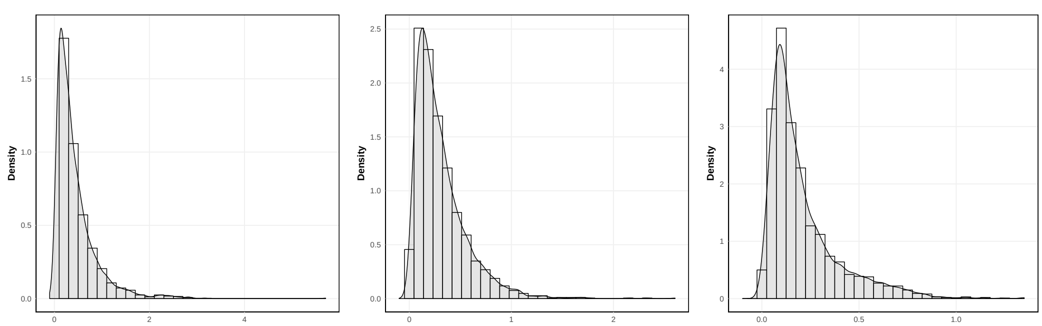
We can analyse the ACF plots for the post burn-in samples for our parameters to determine if there are any serial dependencies in the parameters. Figure (2) shows relatively independent samples, as such we can determine that the posterior mean comparisons are done for an independent sample.



(a) θ_1 (b) θ_2 (c) θ_3

Figure 2: ACF Plots for all $\theta's$

Applying a 50% burn-in for the 4000 samples yields the conditional posterior densities illustrated in Figure 3.



(a) $\theta_1|\theta_2, \theta_3$ (b) $\theta_2|\theta_1, \theta_3$ (c) $\theta_3|\theta_1, \theta_2$

Figure 3: Posterior Densities for $\theta_i|\vec{\theta_j}\ \forall\ i, j$

We can compare the post burn-in posterior means with the MLE's where the MLE's are given by

$$\begin{aligned}
L(\theta_i|\vec{x}) &= \theta_i \exp[-\theta_i x_i] \\
l(\theta_i|\vec{x}) &= \theta_i - \theta_i x_i \\
\frac{dl(\theta_i|\vec{x})}{d\theta_i} &= \frac{1}{\theta_i} - x_i \\
\hat{\theta_i} &= \frac{1}{x_i}
\end{aligned}$$

Looking at Table 1 we see that the posterior means for $\{\theta_2, \theta_3\}$ are fairly similar to their MLE's. The difference between the posterior sample mean for θ_1 is however dissimilar to that of its MLE. This may be due to our modelling process, Bayesian hierarchical models assume commonality (λ) among the parameters ($\vec{\theta}$). The deviation may stem from outlier values in the data which influence how our parameters and their value relative to the communality, a problem that is not present in our MLE. Of interest is that the model preserves the relationship $\theta_1 > \theta_2 > \theta_3$

	Mean	MLE
θ_1	0.459	1.020
θ_2	0.310	0.306
θ_3	0.209	0.166

Table 1: Question 1 Results.

Question 2

In this Question we consider a hierarchical structure where the data follows a binomial distribution $Bin(n_i, \theta_i)$, thus our data is

n_i	20	16	15	25	22
x_i	2	5	5	8	10
$\hat{\theta}_i = \frac{x_i}{n_i}$	0.100	0.313	0.300	0.320	0.455

Table 2: Data for Question 2.

Where each $\theta_i \propto \theta_i^{\alpha-1}(1-\theta_i)^{\beta-1}$, we can see that as θ_i follows a distribution with parameters α, β that we need a hierarchical structure. Assuming that α, β are uninformative and both follow a uniform distribution $\mathcal{U} \sim (0, 10)$ we can build a model using R2OpenBUGS. Model building using R2OpenBUGS simply requires the specification of priors and a sampling distribution to produce posterior results, such a model in shown by Algorithm (1)

Algorithm 1 Bayesian Heirarchical Model

procedure MODEL1(data)

    Specify the "Hyperpriors" on α, β
    $\alpha \leftarrow$ Value from prior distribution ▷ $\pi(\alpha) \sim \mathcal{U}(0, 10)$
    $\beta \leftarrow$ Value from prior distribution ▷ $\pi(\beta) \sim \mathcal{U}(0, 10)$

    Specify the Sampling Distribution.
    for $i \leftarrow 1, N$ **do**
        $\theta_i \sim Beta(\alpha, \beta)$ ▷ Prior Distribution
        $x_i \sim Bin(\theta_i, n_i)$ ▷ Data Distribution
    end for

end procedure

a)

I simulate this model using R2OpenBUGS which utilises an MCMC approach, the settings are set to produce 5000 iterations for 3 chains. We can see the expectation and 95% credibility intervals for our posterior $\theta_i^{\prime}s$ in Table 3

	mean	2.5%	97.5%	Δ
θ_1	0.182	0.053	0.352	0.300
θ_2	0.323	0.155	0.528	0.373
θ_3	0.333	0.158	0.535	0.377
θ_4	0.324	0.177	0.488	0.311
θ_5	0.415	0.250	0.591	0.341

Table 3: Question 2 - Output.

The 95% credibility interval for our $\theta_i^{\prime}s$ can be interpreted as stating our $\theta_i^{\prime}s$ values have a 95% chance of lying in the interval. We can see that $\{\theta_2, \theta_3\}$ are similar in expectation, however, $\{\theta_1, \theta_5\}$ are noticeably smaller and larger (respectively). The interval is smaller for θ_1 indicating we are more certain of this mean estimate being in the interval compared to the other estimates.

	$\Delta = (\hat{\theta}_i - \theta_i)$	Shrinkage
1	-0.082	0.820
2	-0.010	0.032
3	-0.033	0.110
4	-0.004	0.013
5	+0.040	0.088

Table 4: Shrinkage to the mean.

Let's define shrinkage as the change between some crude local estimates $\hat{\theta}_i$ from Table 2 and the posterior expectation over the posterior mean. From Table 4 we can see that for our smallest θ_i we have that our crude estimate is less than its expectation whilst for our largest θ_i our crude estimate is larger than its expectation.

We can formally define shrinkage to the mean by finding an expression for the posterior mean of a beta distribution.

$$
\begin{aligned}
\pi(\vec{\theta}|\alpha, \beta, \vec{x}) &\propto L(\vec{\theta}; \vec{x})\pi(\vec{\theta}|\alpha, \beta)\pi(\alpha)\pi(\beta) \\
&\propto \prod_i L\left(x_i; \theta_i, \lambda\right) \pi\left(\theta_i|\alpha, \beta\right) \times \frac{1}{10} \times \frac{1}{10} \\
&\propto \prod_i L\left(x_i; \theta_i, \lambda\right) \pi\left(\theta_i|\alpha, \beta\right)
\end{aligned}
$$

Where

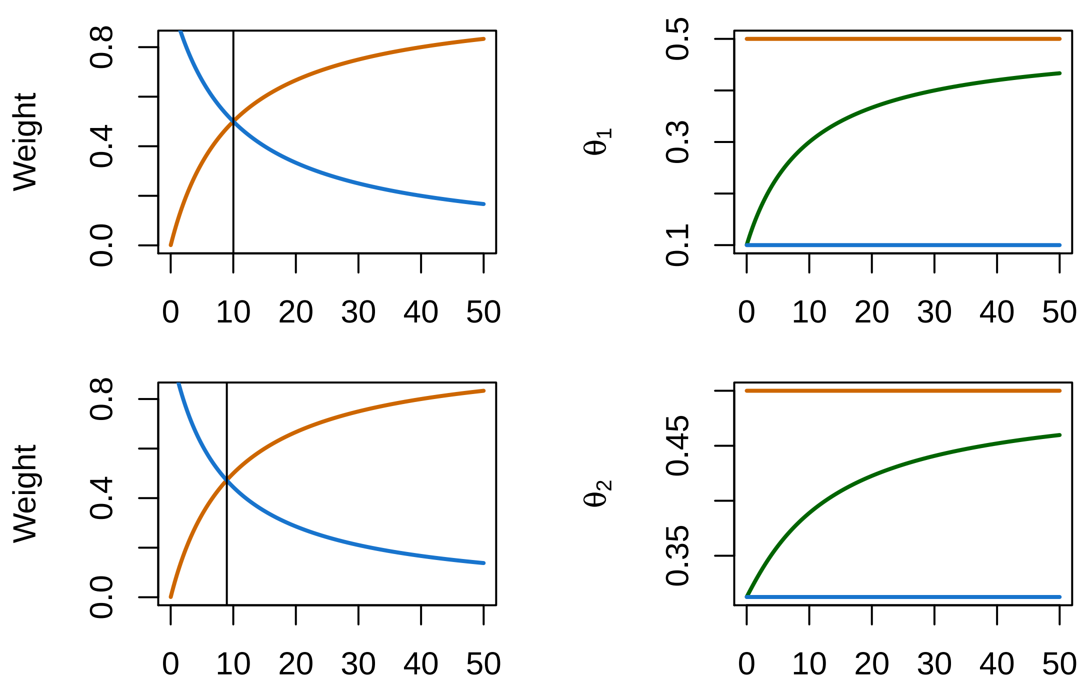
$$
\begin{aligned}
\pi\left(\theta_i|\alpha, \beta, x_i\right) &\propto L\left(x_i; \theta_i\right) \pi\left(\theta_i|\alpha, \beta\right) \\
&\propto \theta_i^{x_i}\left(1-\theta_i\right)^{n_i - x_i} \theta_i^{(}\alpha - 1)\left(1-\theta_i\right)^{\beta-1} \\
&\propto \theta_i^{x_i+\alpha-1}\left(1-\theta_i\right)^{n_i - x_i + \beta - 1} \\
&\sim Beta\left(x_i + \alpha, n_i - x_i + \beta\right)
\end{aligned}
$$

We can thus state our posterior mean as

$$\begin{aligned}
\mathbb{E}(\theta_i|x_i) &= \frac{\alpha^*}{\alpha^*+\beta^*} \\
&= \frac{\alpha+x_i}{n_i-x_i+\beta+\alpha+x_i} \\
&= \frac{\alpha+x_i}{n_i+\beta+\alpha} \\
&= \frac{\alpha}{n_i+\beta+\alpha}+\frac{x_i}{n_i+\beta+\alpha} \\
&= \frac{\alpha+\beta}{n_i+\beta+\alpha}\left(\frac{\alpha}{\alpha+\beta}\right)+\frac{n_i}{n_i+\beta+\alpha}\left(\frac{x_i}{n_i}\right)
\end{aligned}$$

b)

To examine the sensitivity of our results to the assumed prior distributions for α and β we consider a range of $\alpha = \beta$ values for which we can then calculate the necessary weights as per the decomposition above. Figure (4) illustrates this calculation and plot, the left column pertains to the weights whilst the second column pertains to the means. In the left column we see the concave curve pertaining to the prior, with the convex curve being the data. When $\alpha, \beta = 0$ we only weight the mean using the data, however as α and β increase we get increases for the prior weight whilst decreasing the data weight thus the posterior mean tends toward the prior mean. We can determine that when α, β are small the posterior mean tends towards the data's mean (characterised by the MLE). This result concurs with what we saw in Table 4



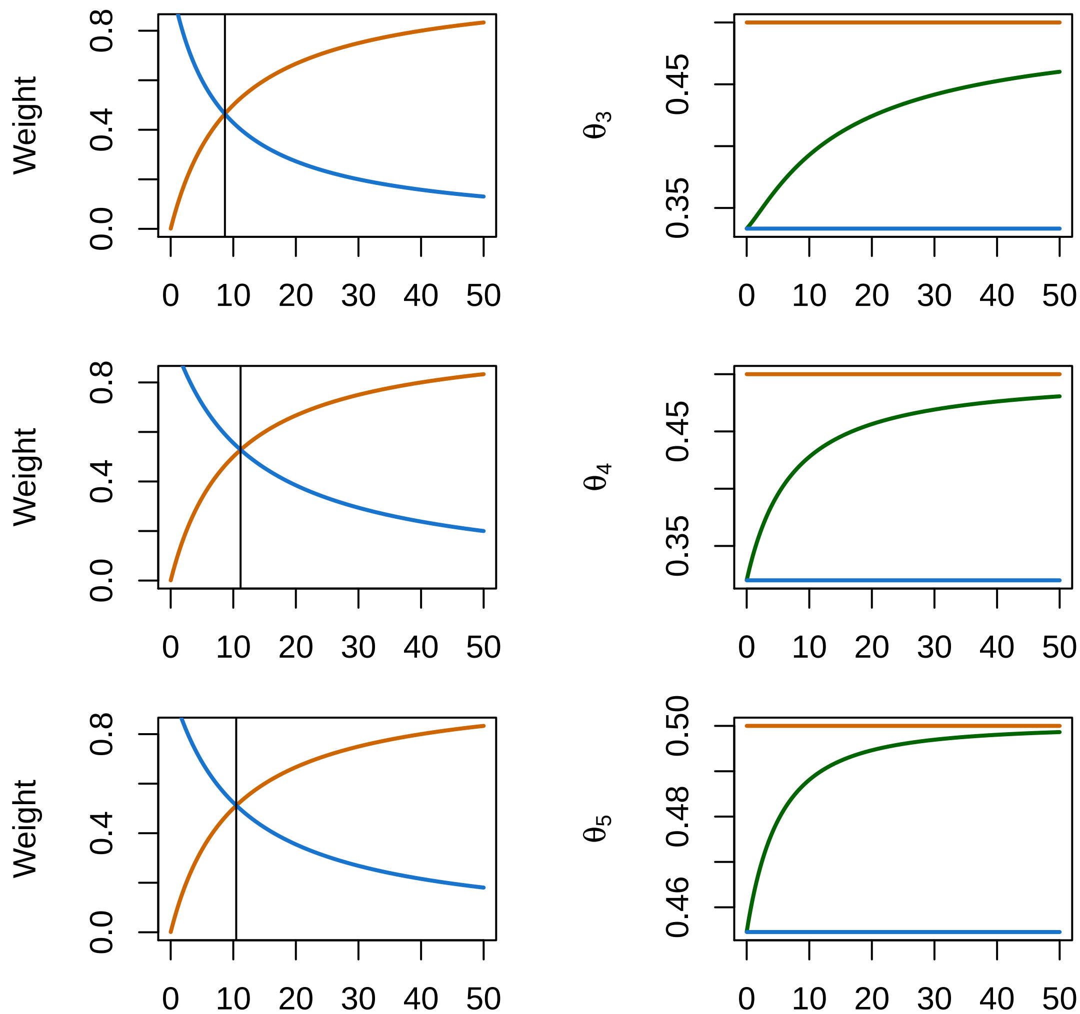


Figure 4: The Weights and Posterior Means for all θ_i's

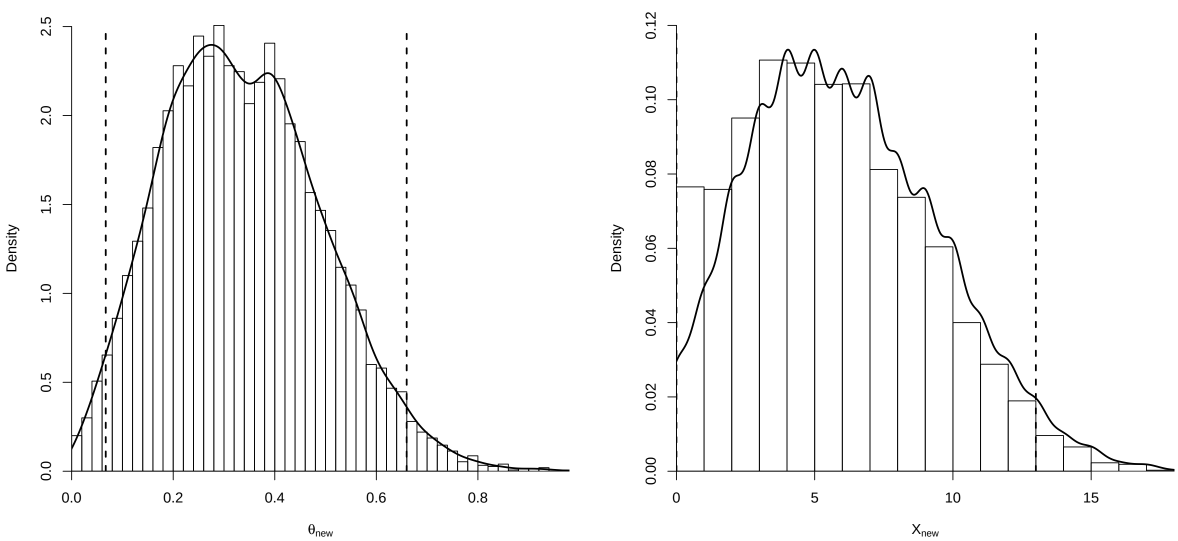
c)

The last part of this question is to produce a predictive distribution for a binomially distributed variable from a new population, i.e. for a sample of size 18 and a new θ_i arising from the same prior distribution. To do this we consider the new variable as an unknown "parameter".

	mean	2.5%	97.5%	Δ
θ_{new}	0.337	0.067	0.660	0.593
X_{new}	6	0	13	-

Table 5: Question 2c - θ_{new} and X_{new} Results.

Doing this will yield the curves in Figure 5, in 5a we determine the mean of our new parameter sampled from the true posterior to be ≈ 0.337 (similar to other θ_i's in Table 3). Lastly we can determine that given a sample size of 18, when we draw a new variable X from a new distributions we will see observe 6 successes and we can state with 95% surety that the number of successes will be in the interval [0, 13].



(a) θ_{new}

(b) Predicitve Distribution of X

Figure 5: Histograms for the Predictive Distribution Results with 95% Credibility Intervals.

Question 3

Here I will use R2OpenBUGS to construct posterior estimates and graphical representations of the posterior distributions for

a) the mean of X

b) $\mathbb{P}(X > 1.2)$ or the probability that that X survives for more than 1.2 time steps.

First we know the decision maker states that the median of X lies between 0.2 and 1.5, we need to express this as a bound on λ conditional on v. Thus,

$$\begin{array}{rcccl}
0.2 & < & \frac{1}{k^{1/v}}\ln(2)^{1/v} & < & 1.5 \\
\frac{0.2}{\ln(2)^{1/v}} & < & \frac{1}{k^{1/v}} & < & \frac{1.5}{\ln(2)^{1/v}} \\
\frac{\ln(2)^{1/v}}{1.5} & < & k^{1/v} & < & \frac{\ln(2)^{1/v}}{0.2} \\
\frac{\ln(2)^{1/v}}{1.5^v} & < & k & < & \frac{\ln(2)^{1/v}}{0.2^v} \\
a & < & k & < & b
\end{array}$$

Where we will assume the corresponding conditional prior for λ is uniform over this bound i.e $\lambda \sim \mathcal{U}(a, b)$ with $v \sim Pa(2, 1)$. Having obtained expressions for λ, a, b and v we can move on to obtaining the desired posteriors. This is done using Algorithm (2) implemented in R2OpenBUGS.

Algorithm 2 Bayesian Heirarchical Model

procedure PART1(data)

  Specify the "Hyperpriors" on a, b, v
  $v \leftarrow$ Value from prior distribution ▷ $\pi(v) \sim Pa(2, 1)$
  $a \leftarrow$ Value from bound calc. ▷ $a = \log(2)/(0.2)^v$
  $b \leftarrow$ Value from bound calc. ▷ $b = \log(2)/(1.5)^v$
  $\lambda \leftarrow$ Value from prior distribution ▷ $\pi(\lambda) \sim \mathcal{U}(a, b)$

  Specify the Sampling Distribution.
  for $i \leftarrow 1, N$ **do**
    $x_i \sim Weibull(v, \lambda)$ ▷ Data Distribution
  end for

end procedure

It is worth noting that finding $\mathbb{P}(X > 1.2)$ using R2OpenBUGS is relatively simple, we can use the step function such that the model generated the correct posterior by using

$$\mathbb{P}(X > 1.2) \leftarrow 1 - step(1.2 - X)$$

This will yield the same answer as defining the variable using the CDF formulation shown below.

$$F(X) = \int_0^x v\lambda x^{v-1}\exp(-\lambda x^v)dx \tag{16}$$

let $u = x^v$ and $du = vx^{v-1}dx$

$$F(X) = \lambda \int_0^x \exp(-\lambda u) du \tag{17}$$

$$= \lambda \left[-\frac{1}{\lambda} \exp(-\lambda u) \Big|_0^{x^v} \right] \tag{18}$$

$$= 1 - \exp(-\lambda x^v) \tag{19}$$

Therefore

$$\mathbb{P}(X > 1.2) = 1 - F(1.2) \tag{20}$$

$$= \exp(-\lambda 1.2^v) \tag{21}$$

Now we can begin to address the objectives above.

a) we define the mean to be

$$\mu = \frac{\Gamma(1 + 1/v)}{\lambda^{1/v}}$$

which can be explicitly defined in R2OpenBUGS whilst objective.

b) requires us to estimate $\mathbb{P}(X > 1.2)$ as above.

Both of these objectives can be achieved by appended Algorithm (2)

Algorithm 3 Bayesian Heirarchical Model

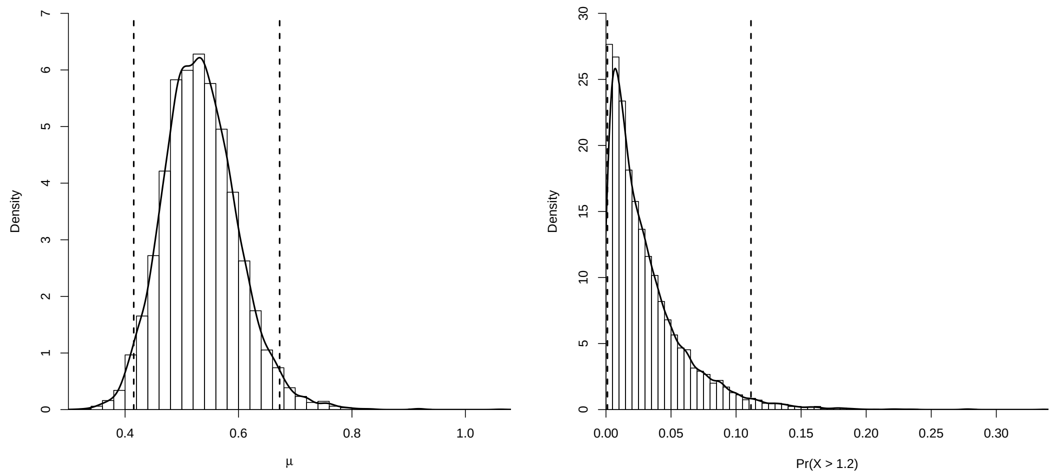
```
procedure PART2(data)
    Algorithm 2 until here...

    log(μ) ← Γ(1 + 1/v) − (1/v) log(λ)
    μ ← exp[log(μ)] = μ
    P(X > 1.2) ← Equation (20)
end procedure
```

Solving this gives the results in Table 6,

	mean	2.5%	97.5%	Δ
μ	0.533	0.415	0.673	0.258
$\mathbb{P}(X > 1.2)$	0.031	0.001	0.111	0.110

Table 6: Question 3 - Mean and $\mathbb{P}(X > 1.2)$ Results.

With posterior densities as per Figure 6. The mean was estimated to be ≈ 0.533, with a 95% credibility interval of with width 0.258. We can determine that the expected lifetime of X is on average 0.533 time units and we are 95% sure that the average lifetime of an object X will be between 0.415 and 0.673 time units.



(a) μ (b) $\mathbb{P}(X > 1.2)$

Figure 6: Posterior Densities from MCMC simulations with 95% Credibility Intervals.

The posterior estimate for is $\mathbb{P}(X > 1.2)$ 0.031 with a 95% CI of [0.001, 0.111]. Similar to the mean lifetime we can determine that the probability that an object X will last for longer than 1.2 time steps is small. On average this event occurs with a chance of 0.03 and we are 95% certain that this probability will be between 0.001 and 0.111. We can conclude that the probability of object X having a lifespan of more than 1.2 time steps is unlikely, with an upper bound of 11%. This is supported by the right-skew distribution for $\mathbb{P}(X > 1.2)$ in Figure (6b)

- END -

References

[1] Einstein, A. Zur Elektrodynamik bewegter Körper. (German) [On the electrodynamics of moving bodies]. *Annalen der Physik 322*, 10 (1905), 891–921.

[2] Goossens, M., Mittelbach, F., and Samarin, A. *The LaTeX Companion.* Addison-Wesley, Reading, Massachusetts, 1993.

[3] Knuth, D. Knuth: Computers and typesetting.

Appendix

R Code

Listing 1: Question 1 Code

```

# UCT ASSIGNMENT
# Author: Julian Albert
# Date:   22/05/2019

# Clean Environment
rm(list = ls()); dev.off()
# Assignment prelim script, useful packages and custom ggplot theme
source("assignment_prelim_script.R")
## 0. Load all the packages for the assignment question
if (!require("pacman")) install.packages("pacman")
#p_load()

# 1. Define functions ----

## 1.1 Parameters
X <- c(0.98, 3.27, 6.04)
theta_init <- c(1, 1, 1)
npars <- length(theta_init)
nsamps <- 4000

## 1.2 Conditional theta_i from inverse probability transform
func.theta_i <- function(theta_j_vec)
{
  u <- runif(1, 0, 1) # random uniform
  theta_j <- sum(theta_j_vec) # j not i summed
  theta_i <- theta_j/((1 - u)^(1/5)) - theta_j # IPT

  return(theta_i)
}

func.accept_reject <- function(pars)
{
  tmp.theta <- pars[1]
  tmp.shape <- pars[2] # 3 in our case
  tmp.rate <- pars[3] # Xi in our case
  tmp.mode <- (tmp.shape - 1)/tmp.rate
  tmp.c <- dgamma(tmp.mode, tmp.shape, tmp.rate)
  tmp.u <- runif(1, 0, 1)

  tmp.target <- dgamma(tmp.theta, tmp.shape, tmp.rate) * tmp.c
  accepted_theta <- ifelse(tmp.target >= tmp.u, tmp.theta, NA)

  return(accepted_theta)
}

# 2. Run accept/reject to get priors for theta ----

### Initialise vectors
tmp.accepted_theta <- numeric()
accepted_theta <- matrix(0, nsamps, npars)

for(j in 1:nsamps)
{

  for(i in 1:npars){
    ### need to specify theta_i | theta_j
    theta_i <- theta_init[i]
    theta_j <- theta_init[-i]
    ### get theta_i | theta_j and run accept reject
    theta_i_conditional <- func.theta_i(theta_j)
    pars <- c(theta_i_conditional, 3, X[i])
    tmp.accepted_theta[i] <- func.accept_reject(pars)
    ### if not accepted, keep trying until accepted
    while( is.na(tmp.accepted_theta[i]) ){
```

```
      theta_i_conditional <- func.theta_i(theta_j)
      pars <- c(theta_i_conditional, 3, X[i])
      tmp.accepted_theta[i] <- func.accept_reject(pars)
    }
    theta_init[i] <- tmp.accepted_theta[i] # theta_i is the accepepted theta
  }
  ### store vector of theta's
  accepted_theta[j, ] <- theta_init
}

# 3. Plots ----

burnin_perc <- 0.5
full_samps_init <- 1
burnt_samps_init <- floor(nsamps*burnin_perc)
burin_samps <- burnt_samps_init:nsamps
plot_thetadf <- data.frame(theta1 = accepted_theta[burin_samps, 1],
                           theta2 = accepted_theta[burin_samps, 2],
                           theta3 = accepted_theta[burin_samps, 3])

setwd('../Figs')
pdf('Theta1_acf.pdf')
acf(plot_thetadf$theta1, main = '')
dev.off()

pdf('Theta2_acf.pdf')
acf(plot_thetadf$theta2, main = '')
dev.off()

pdf('Theta3_acf.pdf')
acf(plot_thetadf$theta3, main = '')
dev.off()

pdf('Theta1_pre_trace.pdf')
plot(plot_thetadf$theta1,
     type = 'l', xlab = 'Sample',
     ylab = bquote(theta[1]))
dev.off()
pdf('Theta2_pre_trace.pdf')
plot(plot_thetadf$theta2, type = 'l', xlab = 'Sample',
     ylab = bquote(theta[2]))
dev.off()
pdf('Theta3_pre_trace.pdf')
plot(plot_thetadf$theta3, type = 'l', xlab = 'Sample',
     ylab = bquote(theta[3]))
dev.off()

pdf('Theta1_post_trace.pdf')
plot(plot_thetadf$theta1,
     type = 'l', xlab = 'Sample',
     ylab = bquote(theta[1]))
dev.off()
pdf('Theta2_post_trace.pdf')
plot(plot_thetadf$theta2, type = 'l', xlab = 'Sample',
     ylab = bquote(theta[2]))
dev.off()
pdf('Theta3_post_trace.pdf')
plot(plot_thetadf$theta3, type = 'l', xlab = 'Sample',
     ylab = bquote(theta[3]))
dev.off()









max_thetas <- apply(plot_thetadf, 2, max) # save maxs for range

setwd('../Figs')
pdf('Theta1_posterior.pdf')
```

```
ggplot(plot_thetadf, aes(x = theta1)) +
  geom_histogram(aes(y = ..density..), col = 'black', fill = 'white') +
  geom_density(alpha = 0.1, fill = 'black') +
  xlim(-0.1, max_thetas[1]) +
  theme_university() +
  labs(y = 'Density', x = '')
dev.off()

pdf('Theta2_posterior.pdf')
ggplot(plot_thetadf, aes(x = theta2)) +
  geom_histogram(aes(y = ..density..), col = 'black', fill = 'white') +
  geom_density(alpha = 0.1, fill = 'black') +
  xlim(-0.1, max_thetas[2]) +
  theme_university() +
  labs(y = 'Density', x = '')
dev.off()

pdf('Theta3_posterior.pdf')
ggplot(plot_thetadf, aes(x = theta3)) +
  geom_histogram(aes(y = ..density..), col = 'black', fill = 'white') +
  geom_density(alpha = 0.1, fill = 'black') +
  xlim(-0.1, max_thetas[3]) +
  theme_university() +
  labs(y = 'Density', x = '')
dev.off()

setwd('../Code')
```

Listing 2: Question 2 Code

```

# UCT ASSIGNMENT
# Author: Julian Albert
# Date:   25/05/2019

# Clean Environment
rm(list = ls()); dev.off()
## 0. Load all the packages for the assignment question
if (!require("pacman")) install.packages("pacman")
p_load(R2OpenBUGS, tidyverse, coda)

# 1. Data ----

X_dat <- c(2, 5, 5, 8, 10)
n_dat <- c(20, 16, 15, 25, 22)
N <- length(X_dat)

data <- list('N', 'X_dat', "n_dat")

# 2. OpenBUGS Model 1 ----

model1 <- function(){

  # HyperPriors
  prior_alpha ~ dunif(0, 10)
  prior_beta ~ dunif(0, 10)

  # Sampling Dist
  for(i in 1:N){
    theta[i] ~ dbeta(prior_alpha, prior_beta) # Prior
    X_dat[i] ~ dbin(theta[i], n_dat[i]) # Data Distribution
  }

}

write.model(model1, "model1.txt")
model.file1 = paste(getwd(), "model1.txt", sep="/")
file.show("model1.txt")

sim1 <- bugs(data, inits = NULL, model.file = model.file1,
             parameters = c("theta", "prior_alpha", "prior_beta"),
             n.chains = 3, n.iter = 5000, debug = TRUE, codaPkg = TRUE)

```

```r
sim1$summary
sim1.coda <- read.bugs(sim1)

# 3. OpenBUGS Model 2 ----

n_new <- 18
data <- list('N', 'X_dat', "n_dat", "n_new")

model2 <- function(){

  # HyperPriors
  prior_alpha ~ dunif(0, 10)
  prior_beta ~ dunif(0, 10)

  # Sampling Dist
  for(i in 1:N){
    theta[i] ~ dbeta(prior_alpha, prior_beta) # Prior
    X_dat[i] ~ dbin(theta[i], n_dat[i]) # Data Distribution
  }

  # New dist
  theta_new ~ dbeta(prior_alpha, prior_beta)
  X_new ~ dbin(theta_new, n_new)

}

write.model(model2, "model2.txt")
model.file2 = paste(getwd(), "model2.txt", sep="/")
file.show("model2.txt")

sim2 <- bugs(data, inits = NULL, model.file = model.file2,
             parameters = c("theta", "prior_alpha", "prior_beta",
                            "theta_new", "X_new"),
             n.chains = 3, n.iter = 5000, debug = TRUE)

sim2$summary
colnames(sim2$sims.matrix)

# new theta
plot_thetanew <- sim2$sims.matrix[, 8]
plot_thetanew_CI <- c(sim2$summary[8, 3],
                      sim2$summary[8, 7])

pdf("q2theta_new.pdf")
hist(plot_thetanew, probability = TRUE,
     xaxs = 'i', yaxs = 'i', breaks = 50,
     main = '',
     #bquote("Distribution of Pr(X > 1.2) including 95% Confidence Intervals"),
     xlab = bquote(theta[new]))
lines(density(plot_thetanew), lwd = 2)
abline(v = plot_thetanew_CI[1], lwd = 2, lty = 2)
abline(v = plot_thetanew_CI[2], lwd = 2, lty = 2)

# new theta
plot_Xnew <- sim2$sims.matrix[, 9]
plot_Xnew_CI <- c(sim2$summary[9, 3],
                  sim2$summary[9, 7])

pdf("q2X_new.pdf")
hist(plot_Xnew, probability = TRUE,
     xaxs = 'i', yaxs = 'i', breaks = 20,
     main = '', ylim = c(0, 0.12),
     #bquote("Distribution of Pr(X > 1.2) including 95% Confidence Intervals"),
     xlab = bquote(X[new]))
lines(density(plot_Xnew), lwd = 2)
abline(v = plot_Xnew_CI[1], lwd = 2, lty = 2)
abline(v = plot_Xnew_CI[2], lwd = 2, lty = 2)
```

Listing 3: Question 2 Shrinkage Curves - Code

```r

# pars
n <- c(20, 16, 15, 25, 22)
```

```
x <- c(2, 5, 5, 8, 10)
alpha <- 10
beta <- 10

# function for specifying first term of posterior mean
func.prior <- function(alpha, beta, ni)
{
  return (list("weight" = (alpha + beta)/(ni + alpha + beta),
               "mean"= (alpha)/(alpha + beta)))
}

# function for specifying second term of posterior mean
func.data <- function(alpha, beta, ni, xi)
{
  return (list("weight"=(ni)/(ni + alpha + beta),
               "mean"= (xi)/(ni)))
}

prior_weights_means <- sapply(n, function(x) func.prior(alpha, beta, x))
data_weights_means <- sapply(1:5, function(i) func.data(alpha, beta, n[i], x[i]))

means <- seq(0.01, 50, length = 1000)

prior_list <- list()
for(i in 1:length(n))
{
  ni <- n[i]
  prior_list[[i]] <- sapply(means,
                            function(x) func.prior(x, x, ni))
}

data_list <- list()
for(i in 1:length(n))
{
  data_list[[i]] <- sapply(means,
                           function(tmp) func.data(tmp, tmp, n[i], x[i]))
}

post_means_list <- lapply(1:5, function(i)
  (as.vector(unlist(prior_list[[i]][1,]))
   * as.vector(unlist(prior_list[[i]][2,])))
  + (as.vector(unlist(data_list[[i]][1,]))
     *as.vector(unlist(data_list[[i]][2,])))
)

for(i in 1:length(n))
{
  test1 <- as.vector(unlist(data_list[[i]][1, ])) -
    as.vector(unlist(prior_list[[i]][1, ]))
  testminval <- min(abs(test1))
  equal_line <- which(abs(test1) == testminval)

pdf(paste('plot_shrinkage', i, '.pdf' ,sep = ''), height = 3, width = 6)
par(mfrow = c(1, 2))
plot(means, prior_list[[i]][1, ],
     type = 'l', col = 'darkorange3', lwd = 2,
     ylab = bquote("Weight"), xlab = '')
lines(means, data_list[[i]][1, ],
      col = 'dodgerblue3', lwd = 2)
abline(v = means[equal_line])
plot(means, post_means_list[[i]],
     ylab = bquote(theta[.(i)]),
     xlab = '',
     type = "l", col="darkgreen", lwd = 2,
     ylim=c(min(unlist(data_list[[i]][2,])),
            max(unlist(prior_list[[i]][2,]))))
lines(means, prior_list[[i]][2,], lwd=2, col="darkorange3")
lines(means, data_list[[i]][2,], lwd=2, col="dodgerblue3")
dev.off()

}
```

Listing 4: Question 3 Code

```r

# UCT ASSIGNMENT
# Author: Julian Albert
# Date:   25/05/2019

# Clean Environment
rm(list = ls()); dev.off()
## 0. Load all the packages for the assignment question
if (!require("pacman")) install.packages("pacman")
p_load(R2OpenBUGS)

# 1. Data ----

X_dat <- c(0.32, 0.63, 0.73, 0.38, 0.54, 0.95, 0.60, 1.03, 0.66, 0.41,
           0.48, 0.27, 0.60, 0.21, 0.39, 0.28, 1.03, 0.68, 0.10, 0.69)

N <- length(X_dat)

data <- list('N', 'X_dat')

model3 <- function(){

  # Hyper-prior
  v ~ dpar(2, 1)
  a <- log(2)/(pow(1.5, v))
  b <- log(2)/(pow(0.2, v))
  lambda ~ dunif(a, b)

  for(i in 1:N){
    # Data distribution
    X_dat[i] ~ dweib(v, lambda)
  }

  # Mean
  logmu <- loggam(1 + 1/v) - (1/v)*log(lambda)
  mu <- exp(logmu)
  x ~ dweib(v, lambda)
  pmorethan <- 1 - step(1.2 - x)
  pmorethanplot <- exp(-lambda * pow(1.2, v))

}

write.model(model3, "model3.txt")
model.file3 = paste(getwd(), "model3.txt", sep="/")
file.show("model3.txt")

sim3 <- bugs(data, inits = NULL, model.file = model.file3,
             parameters = c("x", 'mu', 'pmorethan', 'pmorethanplot',
                            "v", "a", "b", "lambda", 'logmu'),
             n.chains = 3, n.iter = 5000, debug = TRUE)

sim3$summary

# plots
colnames(sim3$sims.matrix)

# mean distribution
plot_mu <- sim3$sims.matrix[, 2]
plot_mu_CI <- c(sim3$summary[2, 3],
                sim3$summary[2, 7])

pdf("q3mu_posterior.pdf")
hist(plot_mu, probability = TRUE,
     xaxs = 'i', yaxs = 'i',
     ylim = c(0, 7), breaks = 50,
     main = '',
     #bquote("Distribution of"~mu~"including 95% Confidence Intervals"),
     xlab = expression(mu))
lines(density(plot_mu), lwd = 2)
abline(v = plot_mu_CI[1], lwd = 2, lty = 2)
abline(v = plot_mu_CI[2], lwd = 2, lty = 2)

```

```
# p>1.2 distribution
plot_pmorethan <- sim3$sims.matrix[, 4]
plot_pmorethan_CI <- c(sim3$summary[4, 3],
                       sim3$summary[4, 7])

pdf("q3pmorethan_posterior.pdf")
hist(plot_pmorethan, probability = TRUE,
     xaxs = 'i', yaxs = 'i',
     ylim = c(0, 30), breaks = 50,
     main = '',
     #bquote("Distribution of Pr(X > 1.2) including 95% Confidence Intervals"),
     xlab = "Pr(X > 1.2)")
lines(density(plot_pmorethan), lwd = 2)
abline(v = plot_pmorethan_CI[1], lwd = 2, lty = 2)
abline(v = plot_pmorethan_CI[2], lwd = 2, lty = 2)
```

Plagiarism Declaration Form

A copy of this form, completed and signed, to be attached to all coursework submissions to the Statistical Sciences Department.

COURSE CODE:	**STA5061Z**
COURSE NAME:	**Bayesian Decision Modelling**
STUDENT NAME:	**Julian Albert**
STUDENT NUMBER:	**ALBJUL005**
GROUP NUMBER:	**N/A**

PLAGIARISM DECLARATION

- I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
- I have used a generally accepted citation and referencing style. Each contribution to, and quotation in, this tutorial/report/project from the work(s) of other people has been attributed, and has been cited and referenced.
- This tutorial/report/project is my own work.
- I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as his or her own work.
- I acknowledge that copying someone else's assignment or essay, or part of it, is wrong, and declare that this is my own work.
- Agreement to this statement does not exonerate me from the University's plagiarism rules.

Signature: Date: June 11, 2019