Chapter 1 Solutions

1.3 The probability density function for the gamma distribution is

$$f(t \mid \alpha, \lambda) = \frac{\lambda^\alpha}{\Gamma(\alpha)} t^{\alpha-1} \exp(-\lambda t).$$

What is the MTTF for the gamma distribution?

By definition,

$$\begin{aligned} MTTF &= \int_{-\infty}^{\infty} t f(t) \mathrm{d}t \\ &= \int_0^\infty t \frac{\lambda^\alpha}{\Gamma(\alpha)} t^{\alpha-1} \exp(-\lambda t) \mathrm{d}t \\ &= \int_0^\infty \frac{\lambda^\alpha}{\Gamma(\alpha)} t^{\alpha} \exp(-\lambda t) \mathrm{d}t \end{aligned}$$

Recognizing that this is a $Gamma(\alpha+1, \lambda)$ distribution, we can write the integral as

$$\int_0^\infty \frac{\lambda^\alpha}{\Gamma(\alpha)} t^{\alpha} \exp(-\lambda t) \mathrm{d}t = \frac{\lambda^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha+1)}{\lambda^{\alpha+1}} = \frac{\alpha}{\lambda}.$$

1.4 The probability density function for the Weibull distribution is

$$f(t \mid \lambda, \beta, \theta) = \lambda\beta(t-\theta)^{\beta-1} \exp\left[-\lambda(t-\theta)^\beta\right], \qquad 0 \le \theta < t, \lambda > 0, \beta > 0.$$

What is the reliability function for the Weibull distribution?

By definition,

$$\begin{aligned} R(t) &= \int_t^\infty f(s) \mathrm{d}s \\ &= \int_t^\infty \lambda\beta(s-\theta)^{\beta-1} \exp[-\lambda(s-\theta)^\beta] \mathrm{d}s. \end{aligned}$$

Noticing that if $u = -\exp(-\lambda(s-\theta)^\beta)$, then $\mathrm{d}u = \lambda\beta(s-\theta)^{\beta-1}\exp(-\lambda(s-\theta)^\beta)\mathrm{d}s$, we have

$$\begin{aligned} R(t) &= -\exp(-\lambda(s-\theta)^\beta) \mid_t^\infty \\ &= \exp(-\lambda(t-\theta)^\beta). \end{aligned}$$

1.5 The probability density function for an exponential random variable is

$$f(t) = \lambda \mathrm{e}^{-\lambda t}, \qquad t > 0, \qquad \lambda > 0.$$

What is the average hazard rate for an exponential random variable?

In Example 1.1.3, it is shown that the cumulative hazard function is $H(t) = \lambda t$. Therefore, by definition, $AHR(t_1, t_2) = \frac{\lambda t_2 - \lambda t_1}{t_2 - t_1} = \frac{\lambda(t_2 - t_1)}{t_2 - t_1} = \lambda$.

1.7 Suppose that we are using the exponential distribution to model an item's lifetime.

(a) We observe that the item failed at 6 hours. What is the likelihood function for this observation?

Let $T \sim \textit{Exponential}(\lambda)$. Then $f(t \mid \lambda) = \lambda e^{-\lambda t}$ and the likelihood function is $l(\lambda \mid t = 6) = \lambda e^{-6\lambda}$.

(b) We observe that the item failed at some time between 5 and 10 hours. What is the likelihood function for this observation?

From Table 1.6, we know that the likelihood function for an interval censored observation has the form $F(t_R) - F(t_L)$. The cumulative distribution function for an exponential random variable is $F(t \mid \lambda) = 1 - \exp(-\lambda t)$. Consequently, the likelihood function is $l(\lambda \mid 5 \leq t \leq 10) = \exp(-5\lambda) - \exp(-10\lambda)$.

(c) We observe the item for 20 hours, and it does not fail. What is the likelihood function for this observation?

From Table 1.6, we know that the likelihood function for a right-censored observation has the form $1 - F(t_R)$. The cumulative distribution function for an exponential random variable is $F(t \mid \lambda) = 1 - \exp(-\lambda t)$. Consequently, the likelihood function is $l(\lambda \mid t \geq 20) = 1 - \exp(-20\lambda)$.

Chapter 2 Solutions

2.1 Suppose that we want to develop an informative prior distribution for the probability of observing heads when we flip a coin. Suppose that we think that the most likely probability of heads is 0.5 and that 0.75 would be "extreme." Find the parameters of a beta density so that the median is approximately 0.5 and the 0.9 quantile is 0.75.

The following R code can be used to solve this problem. q is a vector of length 2 of quantiles, p is vector of length 2 of respective probabilities, *init* is a vector of length 2 of starting parameter values.

For this problem, $\alpha = \beta = 0.374$.

```
parametersolver = function(qu,p,init) {
  qu <- qu
  p <- p

  betaoptim = function(param) {
    q1 <- qu[1]
    q2 <- qu[2]
    p1 <- p[1]
    p2 <- p[2]
    (pbeta(q1,param[1],param[2])-p1)^2 + (pbeta(q2,param[1],param[2])-p2)^2
  }

  r = optim(init,betaoptim)

  v = unlist(r)
  t = c(v[1],v[2])
  print(t)
}

parametersolver(c(.5,.9),c(.5,.75),c(1,1))
```

2.2 Suppose that we are going to flip a coin 20 times.

(a) Using a beta distribution, write down a prior density that describes your uncertainty about the probability of "heads."

Using the prior from problem 2.1, we might use a Beta(0.374, 0.374). This distribution has median is 0.50, the 0.025 quantile is .00022, and the 0.975 quantile is 0.9998. This means there is 50% chance that the probability of heads is below 0.5, and it is very unlikely (a 5% chance) that the probability of getting heads will fall outside the interval (0.00022, 0.99998). The pdf of this distribution is

$$p(\pi) = \frac{\Gamma(0.748)}{\Gamma(0.374)\Gamma(0.374)} \pi^{0.374-1}(1-\pi)^{0.374-1} \qquad 0 \leq \pi \leq 1$$

(b) Flip a coin 20 times and record the outcomes. Write down the likelihood function for the observed data.

Let Y = number of heads in 20 flips of a coin. Then Y $\sim$ Binomial(20, π). My 20 flips of penny had 11 heads, 9 tails. So the likelihood is given by

$$Lik(\pi) = \left(\begin{array}{c} 20 \\ 11 \end{array} \right) \pi^{11}(1-\pi)^9$$

(c) Calculate the maximum likelihood estimate for the probability of "heads" and a 95% confidence interval.

$$\begin{aligned}
\log(Lik(\pi)) &= l(\pi) = log \left(\begin{array}{c} 20 \\ 11 \end{array} \right) + 11log(\pi) + 9log(1-\pi) \\
l^{'}(\pi) &= \frac{11}{\pi} - \frac{9}{1-\pi} = 0 \\
\Rightarrow \hat{\pi} &= \frac{11}{20} \\
s.e.(\hat{\pi}) &= \sqrt{\frac{\hat{\pi}(1-\hat{\pi})}{n}} = \sqrt{\frac{.55(.45)}{20}} = 0.111
\end{aligned}$$

A 95% confidence interval is given by $\hat{\pi} \pm z_{0.975} s.e.(\hat{\pi}) = (0.332, 0.768)$

(d) Calculate the posterior distribution for the probability of "heads" and a 95% credible interval.

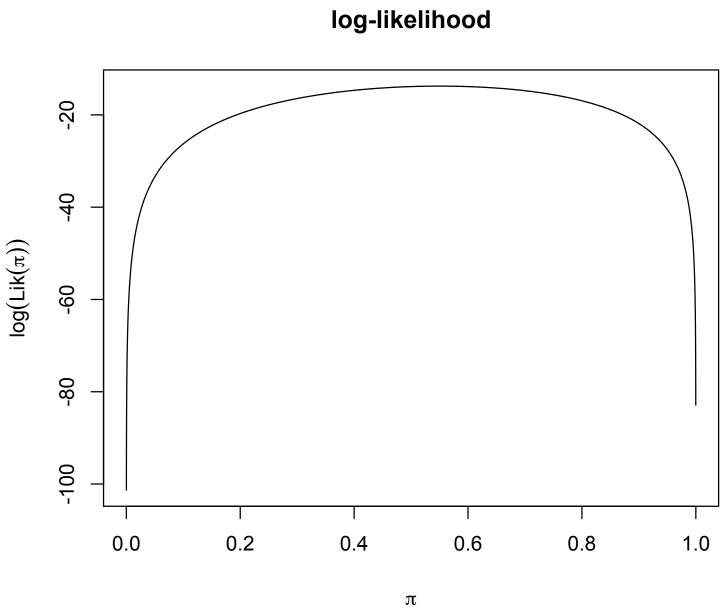
$$\begin{aligned}
p(\pi \,|\, y = 11) &\propto \pi^{11}(1-\pi)^9 \pi^{0.374-1}(1-\pi)^{0.374-1} \\
&\propto Beta(11.374, 9.374)
\end{aligned}$$

Using R to find the 95% credible interval yields (0.337, 0.751).

```
qbeta(.025,11.374,9.374)
qbeta(.975,11.374,9.374)
```

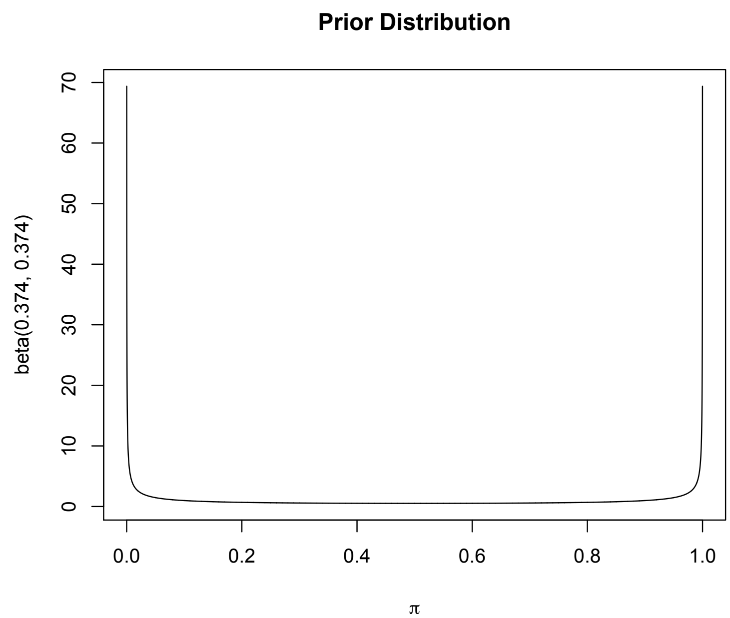
(e) Plot the log-likelihood function.

```
pi <- seq(0,1, by=.0001)
l <- 11*log(pi) + 9*log(1-pi)
plot(pi,l, type='l', main='log-likelihood', xlab=expression(pi),
     ylab=expression(Log-lik(pi)))
```



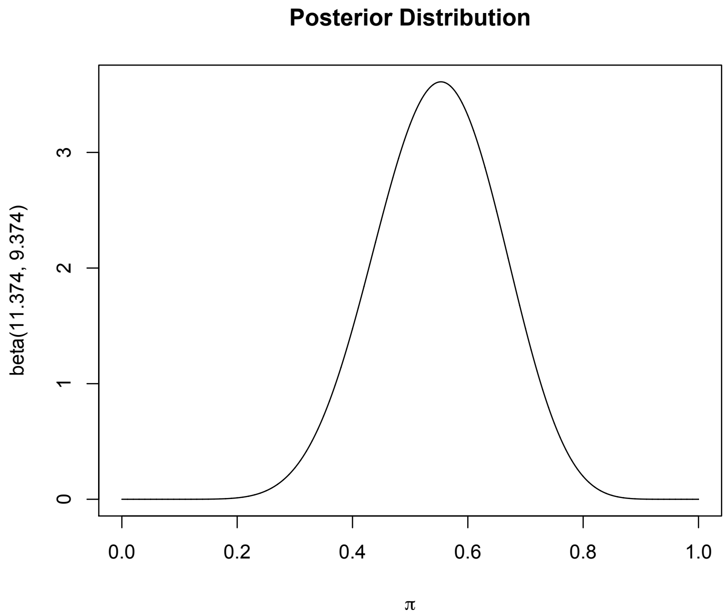
(f) Plot the prior density.

```
alpha <- .374
beta <- .374
beta <- (1/beta(alpha,beta)) * pi^(alpha-1) * (1-pi)^(beta-1)
plot(pi, beta, type='l', main='Prior Distribution',
     xlab=expression(pi), ylab='beta(0.374, 0.374)')
```



(g) Plot the posterior density.

```
alpha <- 11.374
beta <- 9.374
beta <- (1/beta(alpha,beta)) * pi^(alpha-1) * (1-pi)^(beta-1)
plot(pi, beta, type='l', main='Posterior Distribution',
     xlab=expression(pi), ylab='beta(11.374, 9.374)')
```



(h) Calculate the Bayes' factor comparing a uniform prior density to your informative prior density.

Let M_1 be the model with a uniform prior and M_2 be the model with a Beta(0.374, 0.374) prior. Then

$$\begin{aligned}
m_1(11 \,|\, M_1) &= \int_0^1 \binom{20}{11} \pi^{11}(1-\pi)^9 d\pi \\
&= \binom{20}{11} \frac{\Gamma(12)\Gamma(10)}{\Gamma(22)} \int_0^1 \frac{\Gamma(22)}{\Gamma(12)\Gamma(10)} \pi^{12-1}(1-\pi)^{10-1} d\pi \\
&= 1/21 \\
&= 0.04761905
\end{aligned}$$

$$\begin{aligned}
m_2(11 \,|\, M_2) &= \int_0^1 \binom{20}{11} \pi^{11}(1-\pi)^9 \pi^{0.374-1}(1-\pi)^{0.374-1} d\pi \\
&= \binom{20}{11} \frac{\Gamma(11.374)\Gamma(9.374)}{\Gamma(20.748)} \\
&= 0.1175762
\end{aligned}$$

So the Bayes' factor in favor of M_1 is $\frac{0.04761905}{0.1175762} = 0.405$.

2.3 Consider again the fluid breakdown times introduced in Sect. 2.5. Two models were proposed for these data. The first incorporated a normal likelihood function and a noninformative prior distribution; the second a normal likelihood function and a conjugate inverse-gamma/normal prior distribution. Now suppose that the properties of the manufacturing process were controlled when these samples of lubricant were produced so that it is known that the true mean of the sample values must lie between 6.0 and 7.4 (on the original measurement scale). No further information is available concerning the value of the variance parameter σ^2.

(a) Assume that the joint prior distribution for (μ, σ^2) is proportional to $1/\sigma^2$ whenever $\mu \in (\log(6.0), \log(7.4))$, and is 0 otherwise. Find an expression for a function that is proportional to the joint posterior distribution.

$$p(\mu, \sigma^2 \,|\, \mathbf{y}) \propto (\sigma^2)^{-\frac{n}{2}-1} \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mu)^2\right] I[\mu \in (\log(6.0), \log(7.4))]$$

(b) Find a function that is proportional to the marginal posterior distribution of μ.

$$p(\mu \,|\, \mathbf{y}) \propto \left[1 + \frac{(\mu - \bar{y})^2}{(n-1)s^2/n}\right]^{-n/2} I[\mu \in (\log(6.0), \log(7.4)]$$

(c) Find a function that is proportional to the marginal posterior distribution of σ^2.

$$p(\sigma^2 \,|\, \mathbf{y}) \propto \int_{\log(6.0)}^{\log(7.4)} (\sigma^2)^{-n/2-1} \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \mu)^2\right] d\mu$$

2.4 Show that the beta distribution is the conjugate prior distribution for the binomial likelihood.

Suppose that we have data $X \sim Binomial(n, \pi)$, and that the prior distribution for π is $Beta(\alpha, \beta)$. Then the posterior distribution of π is given as

$$\begin{aligned} p(\pi \,|\, x, \alpha, \beta) &\propto \pi^x (1-\pi)^{n-x} \pi^{\alpha-1} (1-\pi)^{\beta-1} \\ &\propto \pi^{x+\alpha-1} (1-\pi)^{n-x+\beta-1}, \end{aligned}$$

which is $Beta(x + \alpha, n - x + \beta)$.

2.5 Show that the gamma distribution is the conjugate prior distribution for the mean of a Poisson likelihood.

Suppose that we have data $X_i \sim Poisson(\lambda)$, and that the prior distribution for λ is $Gamma(\alpha, \beta)$. Then the posterior distribution of π is given as

$$\begin{aligned} p(\lambda \,|\, \vec{x}, \alpha, \beta) &\propto \lambda^{\sum_{i=1}^n x_i} \exp(-n\lambda)\lambda^{\alpha-1}\exp(-\beta\lambda) \\ &\propto \lambda^{\alpha+\sum_{i=1}^n x_i - 1}\exp(-(\beta+n)\lambda), \end{aligned}$$

which is $Gamma(\alpha + \sum_{i=1}^n x_i, n + \beta)$.

2.6 Show that the gamma distribution is the conjugate prior distribution for the exponential likelihood.

Suppose that we have data $X_i \sim Exponential(\lambda)$, and that the prior distribution for λ is $Gamma(\alpha, \beta)$. Then the posterior distribution of π is given as

$$\begin{aligned} p(\lambda \,|\, \vec{x}, \alpha, \beta) &\propto \lambda^n \exp(-\lambda \sum_{i=1}^n x_i)\lambda^{\alpha-1}\exp(-\beta\lambda) \\ &\propto \lambda^{n+\alpha-1}\exp(-(\beta + \sum_{i=1}^n x_i)\lambda), \end{aligned}$$

which is $Gamma(\alpha + n, \beta + \sum_{i=1}^n x_i)$.

2.7 Derive the mean and variance for the lognormal distribution.

$$\begin{aligned} E[X] &= \int_0^\infty x \frac{1}{x\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}(\log(x)-\mu)^2\right] dx \\ &= \int_{-\infty}^\infty \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}y^2)\exp(y+\mu)dy \\ &= \int_{-\infty}^\infty \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}(y-\sigma^2)^2)\exp(\mu+\frac{\sigma^2}{2})dy \\ &= \exp(\mu + \frac{\sigma^2}{2}). \end{aligned}$$

$$\begin{aligned} E[X^2] &= \int_0^\infty x^2 \frac{1}{x\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}(\log(x)-\mu)^2\right] dx \\ &= \int_{-\infty}^\infty \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}y^2)\exp(2y+2\mu)dy \\ &= \int_{-\infty}^\infty \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{1}{2\sigma^2}(y-2\sigma^2)^2)\exp(2\mu+2\sigma^2)dy \\ &= \exp(2\mu + 2\sigma^2). \end{aligned}$$

The mean of the lognormal distribution is $E[X] = \exp(\mu + \frac{\sigma^2}{2})$, and the variance is $Var[X] = E[X^2] - (E[X]^2) = \exp(2\mu + 2\sigma^2) - \exp(2\mu + \sigma^2)$.

2.8 Suppose we are using an Exponential(λ) distribution to model the lifetimes of n items.

(a) Find the maximum likelihood estimator of λ.

The log-likelihood function for λ is $\log(L(\lambda \,|\, \vec{t})) = n\log(\lambda) - \lambda \sum_{i=1}^{n} t_i$. Taking derivatives,

$$\frac{d}{d\lambda}\log(L(\lambda \,|\, \vec{t})) = \frac{n}{\lambda} - \sum_{i=1}^{n} t_i.$$

Setting the derivative equal to zero and solving for λ gives $\frac{n}{\lambda} = \sum_{i=1}^{n} t_i$, or $\hat{\lambda} = \frac{n}{\sum_{i=1}^{n} t_i}$.

(b) Assume n is large and find the standard error of $\hat{\lambda}$.

The second derivative of the log-likelihood is

$$\frac{d^2}{d\lambda^2}\log(L(\lambda \,|\, \vec{t})) = -\frac{n}{\lambda^2}.$$

The observed Fisher information is $\frac{\hat{\lambda}}{\sqrt{n}} = \frac{\sqrt{n}}{\sum_{i=1}^{n} t_i}$.

(c) Suppose that we observed $n = 50$ items and that $\sum_{i=1}^{50} t_i = 25$. Find a 90% confidence interval for λ.

A 90% confidence interval for λ is

$$(\hat{\lambda} - 1.645 se(\hat{\lambda}), \hat{\lambda} + 1.645 se(\hat{\lambda})),$$

which is $(50/25 - 1.645\sqrt{(}50)/25, 50/25 + 1.645\sqrt{(}50)/25) = (2 - 0.465, 2 + 0.465) = (1.535, 2.464)$.

(d) Suppose that $\lambda \sim Gamma(1, 2)$. Find the posterior distribution for λ.

From Exercise 2.6, the posterior distribution is $Gamma(\alpha + n, \beta + \sum_{i=1}^{n} x_i) = Gamma(1 + 50, 2 + 25) = Gamma(51, 27)$.

(e) Suppose that we observed $n = 50$ items and that $\sum_{i=1}^{50} t_i = 25$. What is the posterior probability that λ falls in the 90% confidence interval found in (c)?

$$\int_{1.535}^{2.464} \frac{27^{51}}{\Gamma(51)} \exp(-27x)x^{50}dx = 0.979.$$

Chapter 3

3.1 Suppose that $X \sim Normal(0,1)$ and $Y = \exp(X)$.

(a) Use the change of variables technique to calculate the probability density function, mean, and variance of Y.

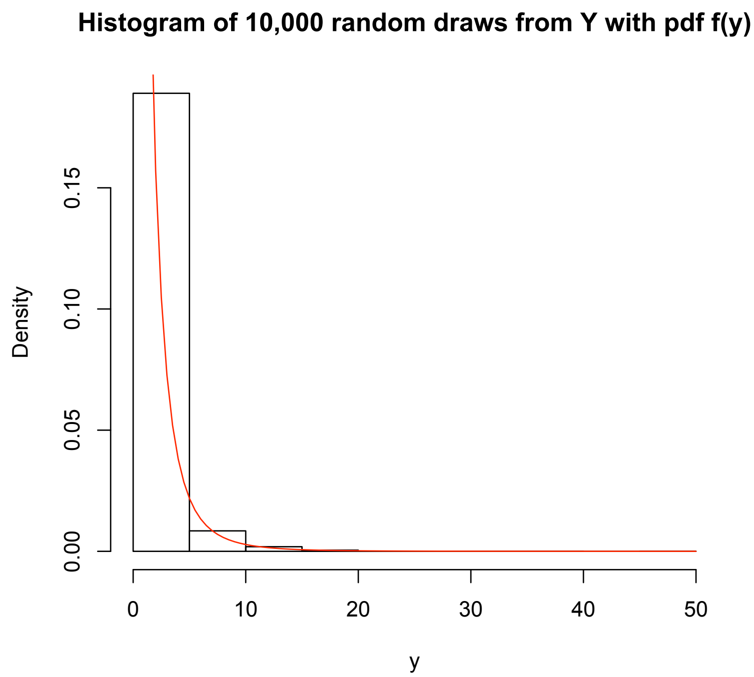
$X \sim$ Normal(0,1) $\quad f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$
$Y \sim e^X \quad g^{-1}(y) = \log(y) \quad \frac{d}{dy} g^{-1}(y) = 1/y$

$$\begin{aligned} f_Y(y) &= f_X(g^{-1}(y)) \left| \frac{d}{dy} g^{-1}(y) \right| \\ &= \frac{1}{\sqrt{2\pi} y} \exp\left[-\frac{1}{2} (\log(y))^2 \right], \quad y > 0 \end{aligned}$$

So $Y \sim LogNormal(0,1)$ with E(Y) $= e^{1/2} = 1.648$ and Var(Y) $= e^2 - e = 4.671$

(b) Draw a random sample $X_1, \ldots, X_{10{,}000}$ from a $Normal(0,1)$ distribution. Set $Y_i = \exp(X_i)$. Draw a histogram of the Y_i and overlay a plot of the probability density function of Y.

```
x <- rnorm(10000,0,1)
y <- exp(x)
f <- function(x){f <- 1/(x*sqrt(2*pi))*exp(-.5*(log(x))^2)}
hist(y, freq=F, main='Histogram of 10,000 random draws from Y
     with pdf f(y)')
curve(f, col=2, add=T)
```



(c) Estimate the probability density function, mean, and variance of Y using the random sample.

```
quantile(y,c(.025,.05,.5,.95,.975))
mean(y)
var(y)
```

			Quantiles				
Parameter	Mean	Variance	0.025	0.050	0.500	0.950	0.975
λ	1.66	4.64	0.14	0.19	0.99	5.17	7.04

3.2 Suppose we perform an experiment where the data have a $Poisson(\lambda)$ sampling density. We describe our uncertainty about λ using a *Gamma* prior density with parameters α and β. We also describe our uncertainty about α and β using independent *Gamma* prior densities.

(a) Simulate 50 observations from a Poisson distribution with parameter $\lambda = 5$.

```
x <- rpois(50,5)
```

(b) Choose diffuse prior densities for α and β.
Let $\alpha \sim Gamma(0.001, 0.001)$ and $\beta \sim Gamma(0.001, 0.001)$.

(c) Implement an MCMC algorithm to calculate posterior densities for λ, α, and β.

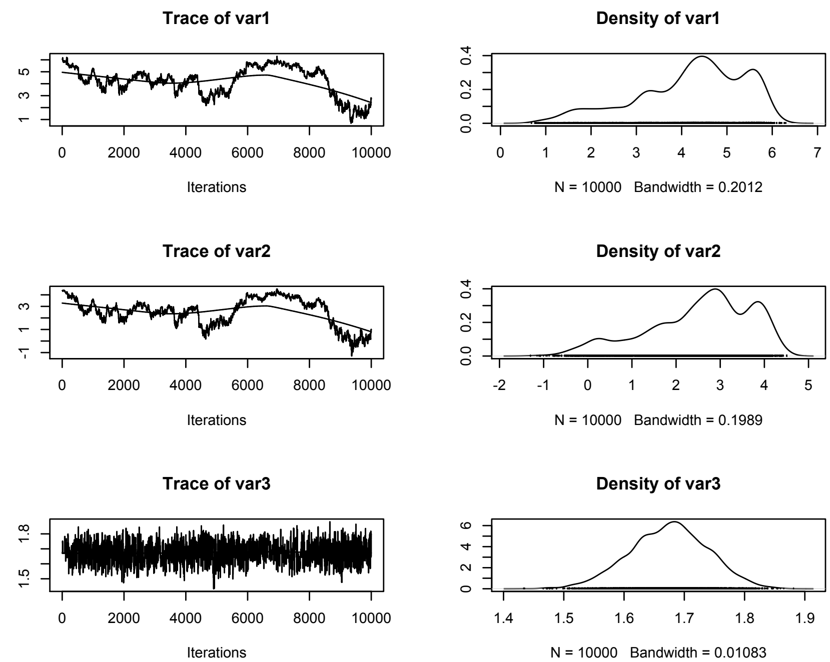
Evaluating the posterior distribution on the log scale helps to minimize numerical issues, as does transforming the parameters to real line. `logpost()` is a function that computes the log of the posterior. `logpost2()` is a function that computes the log transformed parameters.

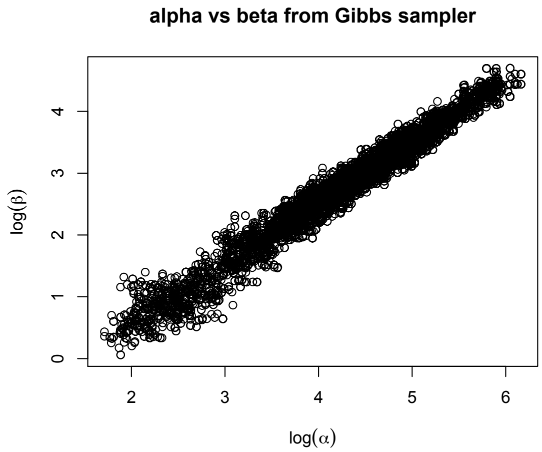
```
library(coda)
library(MASS)
library(LearnBayes)
library(MCMCpack)

logpost <- function(theta, data){
alpha = theta[1]
beta = theta[2]
lambda = theta[3]
n = length(data)
sumd = sum(data)
val = (sumd + alpha -1)*log(lambda) - lambda*(n+beta)
      + alpha*log(beta) - lgamma(alpha) + (.001-1)*log(alpha*beta)
      - .001*(alpha+beta)
return(val)
}

logpost2 <- function(theta, data){ #theta is the log of above theta
a = theta[1]
b = theta[2]
l = theta[3]
return(logpost(c(exp(a), exp(b), exp(l)), data) + a + b + l)
#remember the Jacobian!
}

mu.x <- mean(x)
theta <- c(1,1,mu.x); theta <- log(theta)
fit <- laplace(logpost2, theta, x)

sample <- gibbs(logpost2, fit$mode, 10000, c(.13,.13,.13), data=x)
sample$accept
plot(as.mcmc(sample$par))
plot(sample$par[,1], sample$par[,2], xlab=expression(alpha),
     ylab=expression(beta), main='alpha vs beta from Gibbs sampler')

sample2 <- rwmetrop(logpost2, list(var=fit$var, scale=2), fit$mode,
                    10000, x)
```

```
sample2$accept
plot(as.mcmc(sample2$par))
plot(sample2$par[,1], sample2$par[,2])
```

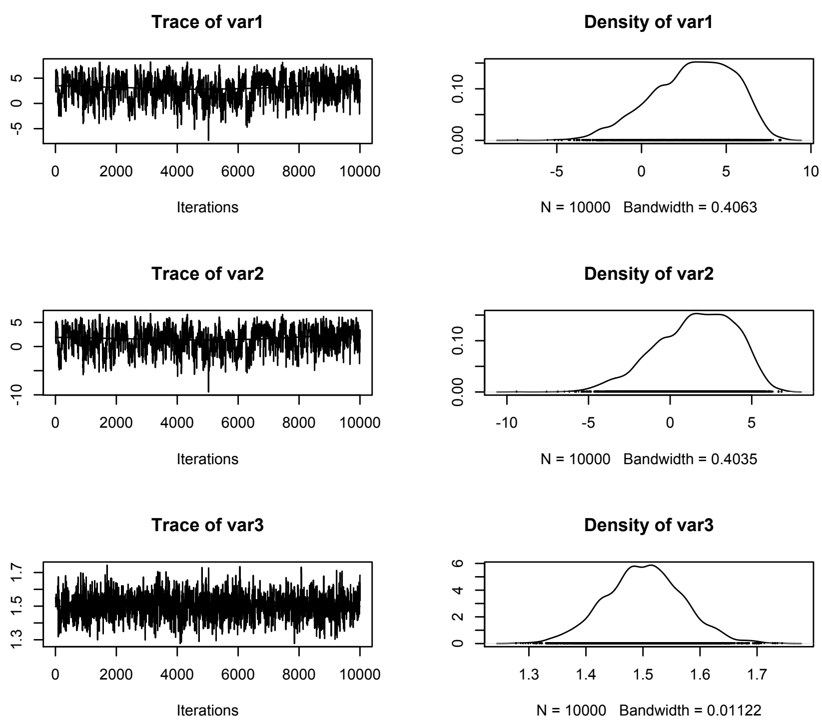
`gibbs()` and `rwmetrop()` are part of the LearnBayes package and `as.mcmc()` is in the MCMCpack package. `rwmetrop` is a random walk Metropolis algorithm, and `gibbs` is a Metropolis-within-Gibbs algorithm. If you would like addition reading and examples with these functions I recommend Jim Albert's *Bayesian Computation with R* (Springer).

The Gibbs sampler has problems mixing, probably due to the high correlation between α and β.





By sampling α and β together, the Metropolis algorithm mixes better than the Gibbs in this problem.



(d) Is $\lambda = 5$ contained in a 90% posterior credible interval for λ?

```
quantile(exp(sample2$par[,3]), c(.05,.95))
```

Yes. Remember that the above graphs are based on the log of the parameters. A 90% posterior credible interval for λ is (4.06, 5.05).

3.3 Consider again the fluid breakdown times introduced in Sect. 2.5. Two models were proposed for these data. The first incorporated a normal likelihood function and a noninformative prior density; the second, a normal likelihood function and a conjugate inverse-gamma/normal prior density. Now suppose that the properties of the manufacturing process were controlled when these samples of lubricant were produced so that it is known that the true mean of the sample values must lie between 6.0 and 7.4 (on the original measurement scale). No further information is available concerning the value of the variance parameter σ^2. Assume that the joint prior density for (μ, σ^2) is proportional to $1/\sigma^2$ whenever $\mu \in (\log(6.0), \log(7.4))$, and is 0 otherwise.

(a) Find an expression for a function that is proportional to the joint posterior density.

We can write the joint prior of μ and σ^2 with the use of an indicator function. i.e., $p(\mu, \sigma^2) \propto \frac{1}{\sigma^2} I[\log(6.0) < \mu < \log(7.4)]$. Then

$$p(\mu, \sigma^2 \,|\, \mathbf{y}) \propto (\sigma^2)^{-\frac{n}{2}-1} \exp\left[-\frac{1}{2\sigma^2} \sum_{i=1}^{n} (y_i - \mu)^2 \right] I[\log(6.0) < \mu < \log(7.4)].$$

(b) Describe a hybrid Gibbs/Metropolis-Hastings algorithm for sampling from the joint posterior density.

$Z \sim Normal(0, 1)$ and c_1 and c_2 are specified constants for μ and σ respectively.

0. Initialize $j = 0$, and starting vales $\mu^{(j)}$, $\sigma^{2(j)}$.

1. Generate μ^* from $\mu^{(j)} + c_1 Z$.

2. Compute $r = \frac{(\sigma^{2(j)})^{-\frac{n}{2}-1} \exp\left[-\frac{1}{2\sigma^{2(j)}} \sum_{i=1}^{n} (y_i - \mu^*)^2\right] I[\log(6.0) < \mu^* < \log(7.4)]}{(\sigma^{2(j)})^{-\frac{n}{2}-1} \exp\left[-\frac{1}{2\sigma^{2(j)}} \sum_{i=1}^{n} (y_i - \mu^{(j)})^2\right] I[\log(6.0) < \mu^{(j)} < \log(7.4)]}$.

3. Draw u from a $Uniform(0, 1)$ distribution.

4. If $u \leq r$, set $\mu^{(j+1)} = \mu^*$. Otherwise, set $\mu^{(j+1)} = \mu^{(j)}$.

5. Generate σ^{2*} from $\sigma^{2(j)} + c_2 Z$.

6. Compute $r = \frac{(\sigma^{2*})^{-\frac{n}{2}-1} \exp\left[-\frac{1}{2\sigma^{2*}} \sum_{i=1}^{n} (y_i - \mu^{(j)})^2\right] I[\log(6.0) < \mu^{(j)} < \log(7.4)]}{(\sigma^{2(j)})^{-\frac{n}{2}-1} \exp\left[-\frac{1}{2\sigma^{2(j)}} \sum_{i=1}^{n} (y_i - \mu^{(j)})^2\right] I[\log(6.0) < \mu^{(j)} < \log(7.4)]}$.

7. Draw u from a $Uniform(0, 1)$ distribution.

8. If $u \leq r$, set $\sigma^{2(j+1)} = \sigma^{2*}$. Otherwise, set $\sigma^{2(j+1)} = \sigma^{2(j)}$.

9. Increment j and return to 1.

3.4 Implement the algorithm in Fig. 3.1. Use batch means to compute the simulation error.

```
m = 10000
pi = array(0, dim=c(m,1))
pi0 = 0.5
pi1 = pi0
for (j in 1:m)
{
pi.star = runif(1,.1,.9)
r = (pi.star^3* (1-pi.star)^8)/(pi1^3*(1-pi1)^8)
u = runif(1) <= r
pi1 = pi.star*(u==1) + pi1*(u==0)
pi[j] = pi1
}
```

To calculate the simulation error via batch means:

```
# 1. Get the means of the batches
b <- 80
v <- m/b; v  #make sure v is a whole number
means <- array(0, dim=c(b,1))
for(i in 1:b){
    means[i,1] = mean(pi[(v*i - (v-1)):(v*i)])
}

# 2. Check that lag 1 autocorrelation of batch means is less than 0.1
library(coda)
autocorr(as.mcmc(means), 1)

# 3. Compute an estimate of the simulation error
mu <- mean(means)
stderror <- sd(means)/sqrt(b)
```

The simulation standard error is 0.0019. You can also get this using `summary(as.mcmc(pi))` from the coda package.

3.5 Implement the algorithm in Fig. 3.4. Calculate the autocorrelation for the chain.

```
data <- read.table('http://www.bayesianreliability.com/wp-content/
        uploads/2009/06/table23.txt', header=T)
```

```
y <- log(data[,1])
n = length(y)
m = 5000
par = array(0, dim=c(m,2))
par1 = c(0,0.5)
s1 = 0.5; s2 = 1
for(j in 1:m){
    mu.star = rnorm(n=1, mean=par1[1], sd=s1)
    r = exp(-sum((y-mu.star)^2)/(2*par1[2]))/
       exp(-sum((y-par1[1])^2)/(2*par1[2]))
    u = runif(1) <= r
    par1[1] = mu.star*(u==1) + par1[1]*(u==0)
    nu = rnorm(1,0,sd=s2)
    var.star = par1[2]*exp(nu)
    r = (var.star^(-n/2)*exp(-sum((y-par1[1])^2)/(2*var.star)))/
        (par1[2]^(-n/2)*exp(-sum((y-par1[1])^2)/(2*par1[2])))
    u = runif(1) <= r
    par1[2] = var.star*(u==1) + par1[2]*(u==0)
    par[j,] = par1
}

# Calculate the autocorrelation
autocorrelation <- function(par, lag){
    l = lag
    A = par[1:(m-l),]
    B = par[(1+l):m,]
    mu1 = mean(par[,1])
    mu2 = mean(par[,2])
    cor1 = (sum((A[,1]-mu1)*(B[,1]-mu1))/(m-l))/
           (sum((par[,1]-mu1)^2)/m)
    cor2 = (sum((A[,2]-mu2)*(B[,2]-mu2))/(m-l))/
           (sum((par[,2]-mu2)^2)/m)
    return(c(cor1, cor2))
}
autocorrelation(par, 1)
```

The autocorrelation for the chain (i.e. lag 1) is 0.8565168 and 0.7419024 for μ and σ^2. You can also get this using `autocorr.diag(as.mcmc(par))` from the coda package.

3.6 In the analysis of the launch vehicle success probabilities described in Example 3.4, the hyperparameters α and λ were assigned values of 5 and 1, respectively.

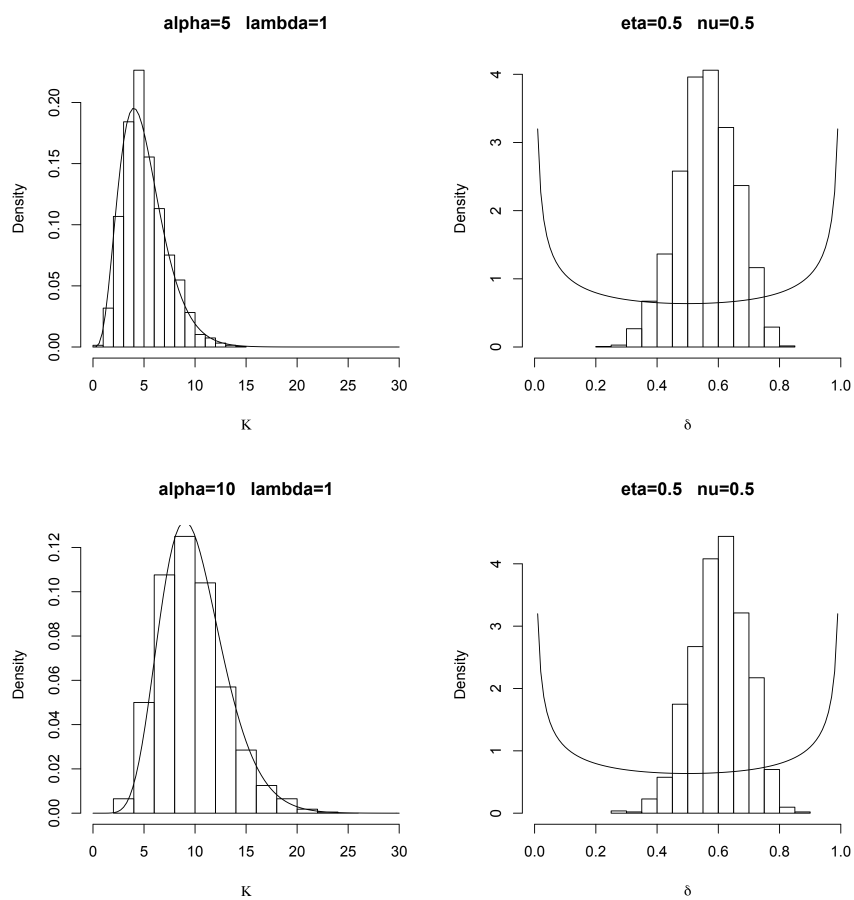
(a) Perform a sensitivity analysis for α and λ by varying their values over a suitable range.

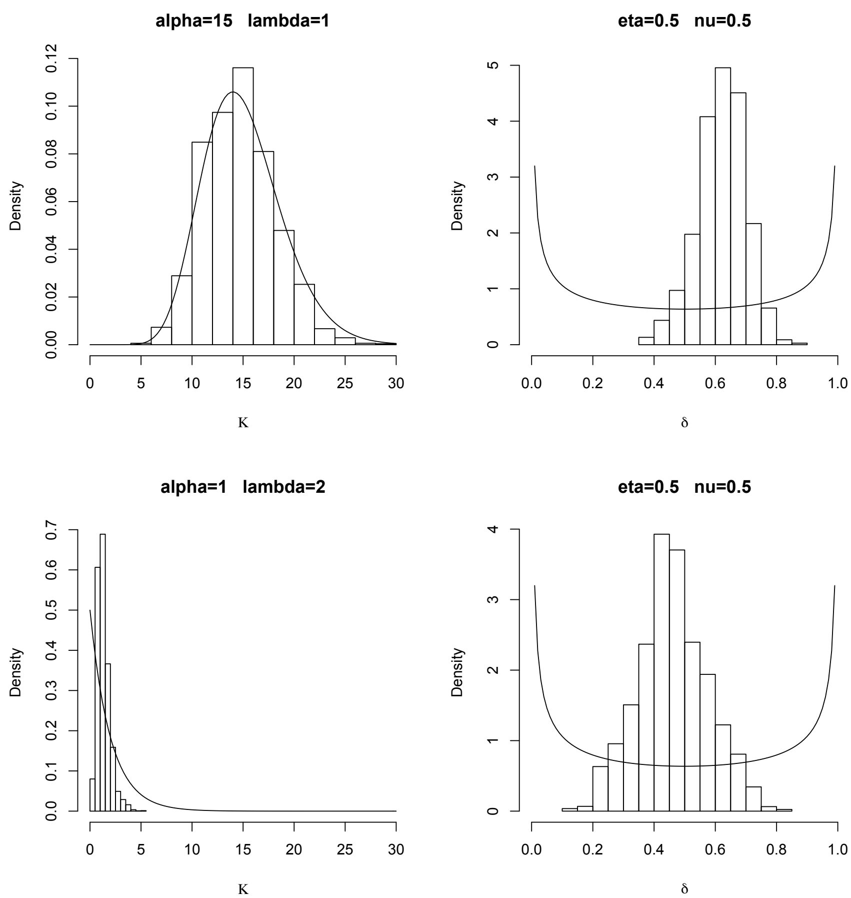
```
data <- read.table('http://www.bayesianreliability.com/wp-content/
    uploads/2009/06/table31.txt', header=T)
m <- data[,3]
y <- data[,2]
launch.mcmc <- function(m, y, size, alpha, lambda, eta, nu){
    K1 = alpha/lambda
    D1 = eta/(eta+nu)
    n = length(m)
    par1 = array(0, dim=c(size,2))
    par2 = array(0, dim=c(n,1))
    arateK = 0; arateD = 0
    # We use the log of the posterior for a more stable algorithm
    logpost = function(K,D){
        val=0
        for(i in 1:n){
            val = val + (y[i]+K*D-1)*log(par2[i]) +
                    (m[i]-y[i]+K-K*D-1)*log(1-par2[i])}
        val = val + n*lgamma(K) - n*lgamma(K*D) - n*lgamma(K-K*D) +
            (alpha-1)*log(K) - lambda*K + (eta-1)*log(D) + (nu-1)*log(1-D)
        return(val)
    }
    for (j in 1:size){
        for (i in 1:n){
            a = y[i] + K1*D1
            b = m[i] - y[i] + K1 - K1*D1
            par2[i] = rbeta(1,a,b)
        }
        z = rnorm(1)
        K.star = K1*exp(z)
        # use log(r) because we want to use the log of the posterior
        logr = logpost(K.star,D1) + log(K.star) - (logpost(K1,D1) + log(K1))
        u = runif(1) <= exp(logr)
        arateK = arateK + u
        K1 = K.star*(u==1) + K1*(u==0)
        c = mean(par2)
        D.star = rbeta(1, K1*c, K1*(1-c))
        logr = (logpost(K1,D.star) + (K1*c-1)*log(D1/D.star)) -
                (logpost(K1,D1) + (K1*(1-c)-1)*log((1-D.star)/(1-D1)))
        u = runif(1) <= exp(logr)
        arateD = arateD + u
        D1 = D.star*(u==1) + D1*(u==0)
        par1[j,] = c(K1,D1)
    }
    arate = c(arateK, arateD); arate = array(arate/size, dim=c(1,2))
```

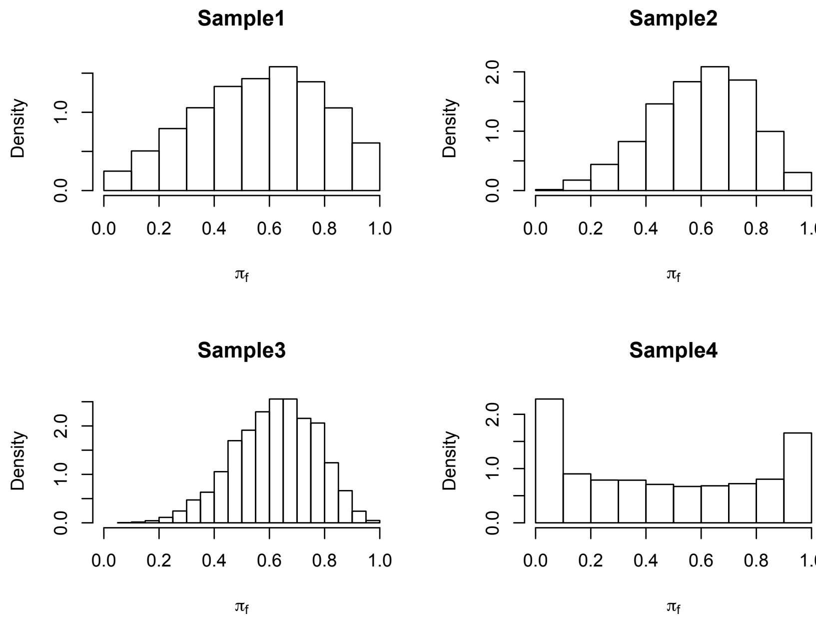
```
    colnames(arate) = c("Kappa", "Delta")
    pif <- array(0, dim=c(size,1))
    for(i in 1:size){
        Kj = par1[i,1]
        Dj = par1[i,2]
        pif[i] = rbeta(1, Kj*Dj, Kj*(1-Dj))
    }
par = cbind(par1, pif)
colnames(par) = c("Kappa", "Delta", "Pif")
ans = list(par = par, accept = arate)
return(ans)
}

sample1 <- launch.mcmc(m,y,size=10000, alpha=5, lambda=1, eta=.5, nu=.5)
sample1$par <- sample1$par[-c(1:50),]
mu <- array(apply(sample1$par, 2, mean), dim=c(3,1))
st.dev <- array(apply(sample1$par, 2, sd), dim=c(3,1))
quant <- rbind(quantile(sample1$par[,1], c(.025,.05,.5,.95,.975)),
    quantile(sample1$par[,2], c(.025, .05, .5, .95, .975)),
    quantile(sample1$par[,3], c(.025, .05, .5, .95, .975)))
summary = array(c(mu, st.dev, quant), dim=c(3,7))
colnames(summary) = c("Mean", "Std Dev", "2.5%",  "5%", "50%",
    "95%", "97.5%")
rownames(summary) = c("Kappa", "Delta", "Pif")
summary
par(mfrow=c(1,2))
hist(sample1$par[,1], freq=F, xlim=c(0,30), xlab=expression(Kappa),
    main='alpha=5   lambda=1')
curve(dgamma(x, shape=5, scale=1), add=T)
hist(sample1$par[,2], xlab=expression(delta), xlim=c(0,1), freq=F,
    main='eta=0.5   nu=0.5')
curve(dbeta(x, .5,.5), add=T)

sample2 <- launch.mcmc(m,y,size=5000, alpha=10, lambda=1, eta=.5, nu=.5)
sample2$par <- sample2$par[-c(1:50),]
mu <- array(apply(sample2$par, 2, mean), dim=c(3,1))
st.dev <- array(apply(sample2$par, 2, sd), dim=c(3,1))
quant <- rbind(quantile(sample2$par[,1], c(.025,.05,.5,.95,.975)),
    quantile(sample2$par[,2], c(.025, .05, .5, .95, .975)),
    quantile(sample2$par[,3], c(.025, .05, .5, .95, .975)))
summary = array(c(mu, st.dev, quant), dim=c(3,7))
colnames(summary) = c("Mean", "Std Dev", "2.5%",  "5%", "50%",
    "95%", "97.5%")
rownames(summary) = c("Kappa", "Delta", "Pif")
summary
par(mfrow=c(1,2))
```

```
hist(sample2$par[,1], freq=F, xlim = c(0,30), xlab=expression(Kappa),
    main='alpha=10   lambda=1')
curve(dgamma(x, shape=10, scale=1), add=T)
hist(sample2$par[,2], xlab=expression(delta), xlim=c(0,1), freq=F,
    main='eta=0.5   nu=0.5')
curve(dbeta(x, .5,.5), add=T)

sample3 <- launch.mcmc(m,y,size=5000, alpha=15, lambda=1, eta=.5, nu=.5)
sample3$par <- sample3$par[-c(1:50),]
mu <- array(apply(sample3$par, 2, mean), dim=c(3,1))
st.dev <- array(apply(sample3$par, 2, sd), dim=c(3,1))
quant <- rbind(quantile(sample3$par[,1], c(.025,.05,.5,.95,.975)),
    quantile(sample3$par[,2], c(.025, .05, .5, .95, .975)),
    quantile(sample3$par[,3], c(.025, .05, .5, .95, .975)))
summary = array(c(mu, st.dev, quant), dim=c(3,7))
colnames(summary) = c("Mean", "Std Dev", "2.5%",  "5%", "50%",
    "95%", "97.5%")
rownames(summary) = c("Kappa", "Delta", "Pif")
summary
par(mfrow=c(1,2))
hist(sample3$par[,1], freq=F, xlim = c(0,30), xlab=expression(Kappa),
    main='alpha=15   lambda=1')
curve(dgamma(x, shape=15, scale=1), add=T)
hist(sample3$par[,2], xlab=expression(delta), xlim=c(0,1), freq=F,
    main='eta=0.5   nu=0.5')
curve(dbeta(x, .5,.5), add=T)

sample4 <- launch.mcmc(m,y,size=5000, alpha=1, lambda=2, eta=.5, nu=.5)
sample4$par <- sample4$par[-c(1:50),]
mu <- array(apply(sample4$par, 2, mean), dim=c(3,1))
st.dev <- array(apply(sample4$par, 2, sd), dim=c(3,1))
quant <- rbind(quantile(sample4$par[,1], c(.025,.05,.5,.95,.975)),
    quantile(sample4$par[,2], c(.025, .05, .5, .95, .975)),
    quantile(sample4$par[,3], c(.025, .05, .5, .95, .975)))
summary = array(c(mu, st.dev, quant), dim=c(3,7))
colnames(summary) = c("Mean", "Std Dev", "2.5%",  "5%", "50%",
    "95%", "97.5%")
rownames(summary) = c("Kappa", "Delta", "Pif")
summary
par(mfrow=c(1,2))
hist(sample4$par[,1], freq=F, xlim = c(0,30), xlab=expression(Kappa),
    main='alpha=1   lambda=2')
curve(dgamma(x, shape=1, scale=2), add=T)
hist(sample4$par[,2], xlab=expression(delta), xlim=c(0,1), freq=F,
    main='eta=0.5   nu=0.5')
curve(dbeta(x, .5,.5), add=T)
```

```
par(mfrow=c(2,2))
hist(sample1$par[,3], freq=F, xlim=c(0,1), xlab=expression(pi[f]),
    main='Sample1')
hist(sample2$par[,3], freq=F, xlim=c(0,1), xlab=expression(pi[f]),
    main='Sample2')
hist(sample3$par[,3], freq=F, xlim=c(0,1), xlab=expression(pi[f]),
    main='Sample3')
hist(sample4$par[,3], freq=F, xlim=c(0,1), xlab=expression(pi[f]),
    main='Sample4')
```




alpha=15 lambda=1
eta=0.5 nu=0.5
alpha=1 lambda=2
eta=0.5 nu=0.5
Density
K
δ



				Quantiles	
	Parameter	Mean	Std Dev	0.05	0.95
sample1	$\kappa = 5$	5.12	2.12	2.25	9.04
	$\delta = 0.5$	0.56	0.09	0.40	0.71
	π_f	0.56	0.23	0.16	0.91
sample2	$\kappa = 10$	9.73	3.21	5.26	15.54
	$\delta = 0.5$	0.60	0.09	0.46	0.74
	π_f	0.60	0.18	0.28	0.87
sample3	$\kappa = 15$	14.62	3.49	9.40	20.82
	$\delta = 0.5$	0.62	0.08	0.48	0.74
	π_f	0.62	0.15	0.36	0.85
sample4	$\kappa = 0.5$	1.32	0.64	0.53	2.49
	$\delta = 0.5$	0.46	0.12	0.27	0.66
	π_f	0.46	0.35	0.00	0.99

An alternative to running an MCMC algorithm several times, as done above, is to do sampling importance resampling (SIR). For more information see Albert's *Bayesian Computation with R* section 5.10 (Springer). The idea is that we can take a weighted bootstrap sample (with replacement) from the current posterior distribution to get a sample from the new posterior distribution. The weights are computed by calculating $\frac{NewPosterior}{OldPosterior}$. Then convert the weights to probabilities by normalizing them to add to 1. Lastly, resample the sample from the OLD posterior using these probabilities to get a sample the NEW posterior.

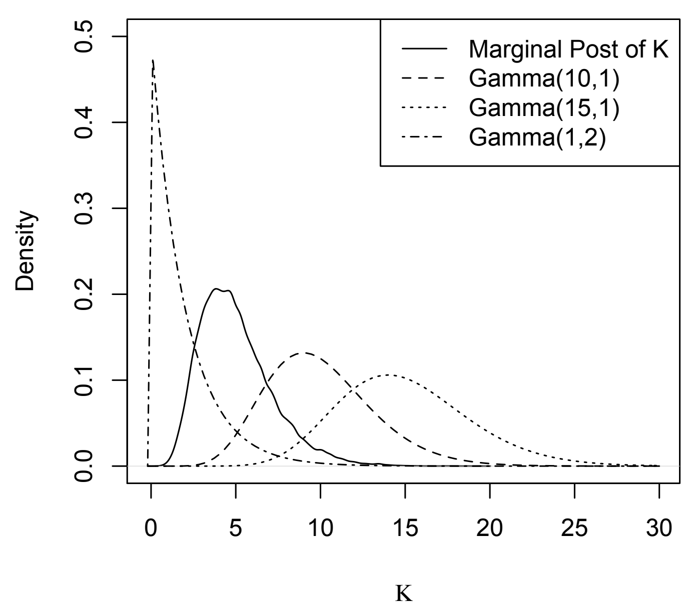
In this exercise, we are on the log scale. Since we are only interested in how changes to the prior on K affects the posterior distribution,

$$\log\left(\frac{NewPosterior}{OldPosterior}\right) = \log\left(\frac{p(K \mid \alpha,\lambda)_{NEW}}{p(K \mid \alpha,\lambda)_{OLD}}\right).$$

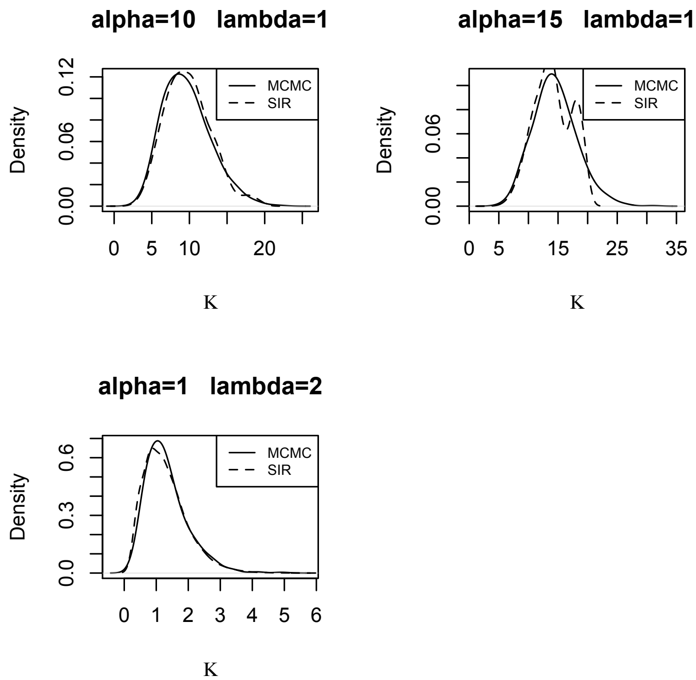
Then the weights to get an approximation to `sample2` above are

$$\begin{aligned}\log\left(\frac{p(K \mid \alpha,\lambda)_{NEW}}{p(K \mid \alpha,\lambda)_{OLD}}\right) &\propto \log\left(\frac{K^9 e^{-K}}{K^4 e^{-K}}\right) \\ &= 5\log K.\end{aligned}$$

For this particular example, SIR works well for getting an approximation to `sample2` and `sample4` above. However, for `sample3`, SIR gives a poor approximation. The graph below shows the marginal posterior that we will be sampling from, along with the three new priors on K that we used for samples 2 through 4 above. With SIR, be careful that you have enough points in the tails that will be heavily weighted in the resampling. To get a decent approximation to `sample2` I had to increase my MCMC sample size to 100,000 in order to get sufficient data in the upper tail. We can also see that for `sample3`, our samples for the marginal posterior do not extend nearly far enough to get a good approximation for that distribution.



The graphs below show how the three SIR approximations (resampled from the 100,000 size MCMC) compared to the three additional MCMC runs called `sample2, sample3` and `sample4` above.



Here is the code used for SIR

```
# Sampling Importance Resampling (SIR)
sample1 <- launch.mcmc(m,y,size=100000, alpha=5, lambda=1, eta=.5, nu=.5)
sample1$par <- sample1$par[-c(1:50),]
lw <- 5*log(sample1$par[,1])
lw <- lw - max(lw)   #so we don't exponentiate anything too large or small
wt <- exp(lw)/sum(exp(lw))  #normalizing
ind <- sample(1:99950, replace=T, prob=wt)  #easier to sample the
s2 <- sample1$par[ind,]                     #indices of sample1

lw <- 10*log(sample1$par[,1])
lw <- lw - max(lw)
wt <- exp(lw)/sum(exp(lw))
ind <- sample(1:99950, replace=T, prob=wt)
s3 <- sample1$par[ind,]

lw <- -4*log(sample1$par[,1]) - sample1$par[,1]
lw <- lw - max(lw)
wt <- exp(lw)/sum(exp(lw))
ind <- sample(1:99950, replace=T, prob=wt)
s4 <- sample1$par[ind,]
```

(b) Report how changes in the values assumed for λ and α impact the posterior means of other model parameters.

The choice of K has a small effect on the marginal posterior distribution of δ. As we increase K, the mean of the marginal posterior of δ increases slightly. The posterior predictive mean for π_f is roughly equal to the posterior mean of δ, so our choice of K has the same effect on the posterior mean of π_f as that of δ.

3.7 Derive the conditional densities described in Example 3.4 for the random effects model.

The joint posterior distribution is

$$p(\mu, \sigma^2, \kappa, \beta \,|\, \mathbf{y}) \propto \kappa^{-9.5}(\sigma^2)^{-31}\exp\left[-\frac{0.25}{\kappa} - \frac{1}{2\sigma^2\kappa}\sum_{j=1}^{10}\beta_j^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{5}\sum_{j=1}^{10}(y_{ij} - \beta_j - \mu)^2\right].$$

So the full conditional densities are

$$\begin{aligned}
p(\mu \,|\, \beta, \sigma^2, \kappa, \mathbf{y}) &\propto \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{5}\sum_{j=1}^{10}(y_{ij} - \beta_j - \mu)^2\right] \\
&\propto \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{5}\sum_{j=1}^{10}(\mu - (y_{ij} - \beta_j))^2\right] \\
&\propto \exp\left[-\frac{1}{2\sigma^2}\left(50\mu^2 - 2\mu\sum_{i=1}^{5}\sum_{j=1}^{10}(y_{ij} - \beta_j)\right)\right] \\
&\propto \exp\left[-\frac{1}{2(\sigma^2/50)}\left(\mu - \frac{1}{50}\sum_{i=1}^{5}\sum_{j=1}^{10}(y_{ij} - \beta_j)\right)^2\right] \\
&\propto \frac{1}{\sqrt{2\pi(\sigma^2/50)}}\exp\left[-\frac{1}{2(\sigma^2/50)}\left(\mu - \frac{1}{50}\sum_{i=1}^{5}\sum_{j=1}^{10}(y_{ij} - \beta_j)\right)^2\right] \\
&\sim \mathit{Normal}\left(\frac{1}{50}\sum_{i=1}^{5}\sum_{j=1}^{10}(y_{ij} - \beta_j), \frac{\sigma^2}{50}\right)
\end{aligned}$$

$$
\begin{aligned}
p(\beta_j \,|\, \beta_{\mathbf{i}\neq\mathbf{j}}, \mu, \sigma^2, \kappa, \mathbf{y}) &\propto \exp\left[-\frac{1}{2\sigma^2\kappa}\sum_{j=1}^{10}\beta_j^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{5}(y_{ij}-\beta_j-\mu)^2\right] \\
&\propto \exp\left[-\frac{1}{2\sigma^2}\left(\frac{1}{\kappa}\beta_j^2 + \sum_{i=1}^{5}(\beta_j-(y_{ij}-\mu))^2\right)\right] \\
&\propto \exp\left[-\frac{1}{2\sigma^2}\left(\frac{1}{\kappa}\beta_j^2 + 5\beta_j^2 - 2\beta_j\sum_{i=1}^{5}(y_{ij}-\mu) + \sum_{i=1}^{5}(y_{ij}-\mu)^2\right)\right] \\
&\propto \exp\left[-\frac{5+1/\kappa}{2\sigma^2}\left(\beta_j - \frac{\sum_{i=1}^{5}(y_{ij}-\mu)}{5+1/\kappa}\right)^2\right] \\
&\propto \frac{1}{\sqrt{2\pi(\frac{1}{5/\sigma^2+1/(\kappa\sigma^2)})}}\exp\left[-\frac{\frac{5}{\sigma^2}+\frac{1}{\kappa\sigma^2}}{2\sigma^2}\left(\beta_j - \frac{\sum_{i=1}^{5}(y_{ij}-\mu)/\sigma^2}{\frac{5}{\sigma^2}+\frac{1}{\kappa\sigma^2}}\right)^2\right] \\
&\sim Normal\left(\frac{\sum_{i=1}^{5}(y_{ij}-\mu)/\sigma^2}{5/\sigma^2+1/(\kappa\sigma^2)}, \frac{1}{5/\sigma^2+1/(\kappa\sigma^2)}\right)
\end{aligned}
$$

$$
\begin{aligned}
&p(\sigma^2 \quad | \quad \beta, \mu, \kappa, \mathbf{y}) \\
&\propto \quad (\sigma^2)^{-31}\exp\left[-\frac{1}{2\sigma^2\kappa}\sum_{j=1}^{10}\beta_j^2 - \frac{1}{2\sigma^2}\sum_{i=1}^{5}\sum_{j=1}^{10}(y_{ij}-\beta_j-\mu)^2\right] \\
&\propto \quad \frac{\left[\frac{1}{2}\sum_{i=1}^{5}\sum_{j=1}^{10}(y_{ij}-\beta_j-\mu)^2 + \frac{1}{2\sigma^2}\sum_{j=1}^{10}\frac{\beta_j^2}{\kappa}\right]^{30}}{\Gamma(30)}(\sigma^2)^{-(30+1)} \\
&\qquad \times\exp\left[-\frac{1}{\sigma^2}\left(\frac{1}{2}\sum_{i=1}^{5}\sum_{j=1}^{10}(y_{ij}-\beta_j-\mu)^2 + \frac{1}{2}\sum_{j=1}^{10}\frac{\beta_j^2}{\kappa}\right)\right] \\
&\sim \quad InverseGamma\left(30,\ \frac{1}{2}\sum_{i=1}^{5}\sum_{j=1}^{10}(y_{ij}-\beta_j-\mu)^2 + \frac{1}{2}\sum_{j=1}^{10}\frac{\beta_j^2}{\kappa}\right)
\end{aligned}
$$

$$
\begin{aligned}
p(\kappa \,|\, \beta, \mu, \sigma^2, \mathbf{y}) &\propto \kappa^{-(8.5+1)} \exp\left[-\frac{0.25}{\kappa} - \frac{0.5}{\sigma^2 \kappa} \sum_{j=1}^{10} \beta_j^2 \right] \\
&\propto \kappa^{-(8.5+1)} \exp\left[-\frac{1}{\kappa} \left(0.25 + 0.5 \sum_{j=1}^{10} \beta_j^2 / \sigma^2 \right) \right] \\
&\sim InverseGamma\left(8.5, 0.25 + 0.5 \sum_{j=1}^{10} \beta_j^2 / \sigma^2 \right)
\end{aligned}
$$

Chapter 4

4.1 The $Beta(293, 0.5)$ prior for π yields a posterior with a median of 0.999 and a 95% credible interval of (0.990, 1.000), whose length is 0.010. The uniform prior (i.e., $Beta(1, 1)$) yields a posterior with a median of 0.997 and a 95% credible interval of (0.983, 1.000), whose length is 0.017.

4.3 The likelihood is $\prod_{i=1}^{n}(\lambda t_i)^{y_i}\exp(-\lambda t_i)/y_i! \propto \lambda^{\sum_{i=1}^{n} y_i}\exp(-\sum_{i=1}^{n} t_i\lambda)$ and the prior distribution is proportional to $\lambda^{\alpha-1}\exp(-\beta\lambda)$. Using Bayes' Theorem, the posterior distribution is proportional to $\lambda^{\alpha+\sum_{i=1}^{n} y_i - 1}\exp[-(\beta+\sum_{i=1}^{n} t_i)\lambda]$ so that $\lambda|\vec{y},\vec{t} \sim Gamma(\alpha+\sum_{i=1}^{n} y_i, \beta+\sum_{i=1}^{n} t_i)$.

4.4 The $Gamma(5, 1)$ prior for λ yields a posterior with a mean of 2.85 and a 95% credible interval of (2.40, 3.05), whose length is 0.65. The $Uniform(0, 20)$ prior yields a posterior with a mean of 3.00 and a 95% credible interval of (2.52, 3.52), whose length is 1.00.

4.12 A 95% credible interval for μ is (2.65, 6.29) with a posterior median of 3.81. A 95% credible interval for λ is (1.01, 2.27) with a posterior median of 1.57. Using $K = 5$ equal probability bins, we find that 0.7% of the R^B test statistics exceed the 0.95 quantile of the $ChiSquared(4)$ reference distribution, which suggests no lack of fit. See the plot of the posterior median of reliability with 90% credible interval over the first 5 hours in Figure 1.

4.13 Using $K = 5$ equal probability bins, we find that 4.9% of the R^B test statistics exceed the 0.95 quantile of the $ChiSquared(4)$ reference distribution, which suggests no lack of fit. The DIC for the hierarchical model is 245.446 and the DIC for the constant failure probability model is 285.322. Based on DIC, we prefer the hierarchical model. We provide the WinBUGS code for the hierarchical in Table **??**.

Table 1: WinBUGS code for exercise 4.13

```
########################################################
# Exercise 4.13
# EDG Hierarchical Model
# for success probabilities pi[]
# x[] number of failures in n[] trials

model
{
for( i in 1 : N ) {
z[i]<-ind[i]
y[i]<-n[i]-x[i]
y[i] ~ dbin(pi[i],n[i])
pi[i] ~dbeta(delta,gamma)I(.0001,.9999)
}

#use InverseGamma
delta<-1/rdelta
gamma<-1/rgamma
rdelta ~ dgamma(0.1,0.1)I(.001,1000)
rgamma ~ dgamma(0.1,0.1)I(.001,1000)

}

Data

list(N = 63)

Inits

list(
rdelta=.003,rgamma=.5,
pi=c(
.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,
.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,
.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,
.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,
.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,
.5,.5,.5,.5,.5,.5,.5,.5,.5,.5,
.5,.5,.5)
)
########################################################
```

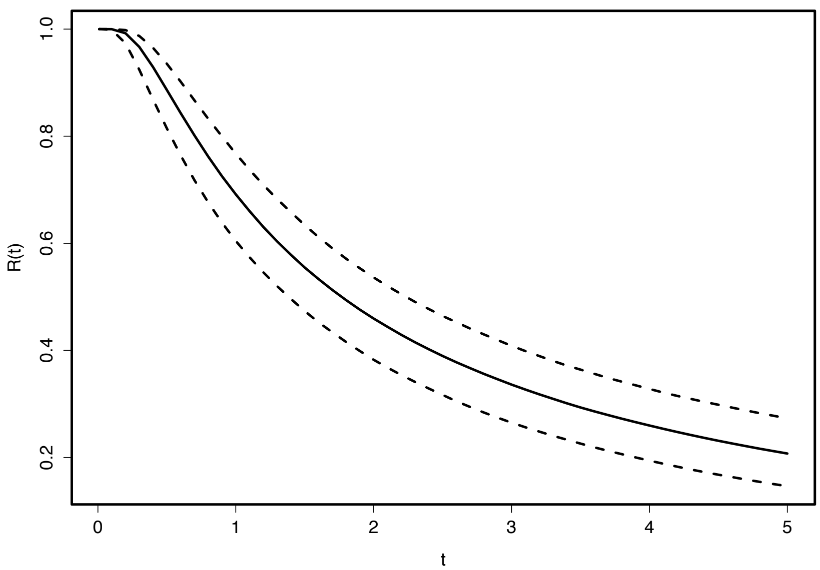


Figure 1: Exercise 4.12 median posterior of reliability and 95% credible intervals versus time t in hours.

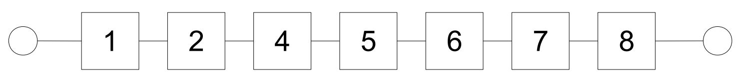
Chapter 5

5.1 Draw a reliability block diagram describing how to successfully perform an everyday task.

Consider the task of brushing your teeth. The following is a list of possible components for the block diagram:

1. get toothbrush
2. put toothpaste on toothbrush
3. put water on toothbrush
4. brush teeth
5. brush tongue
6. spit out toothpaste
7. rinse mouth
8. rinse toothbrush

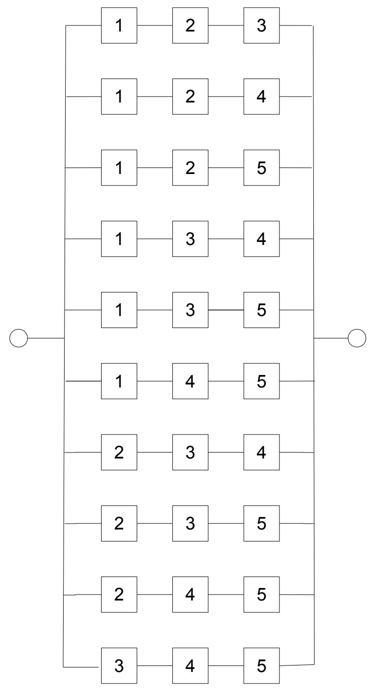
Below is the reliability block diagram.

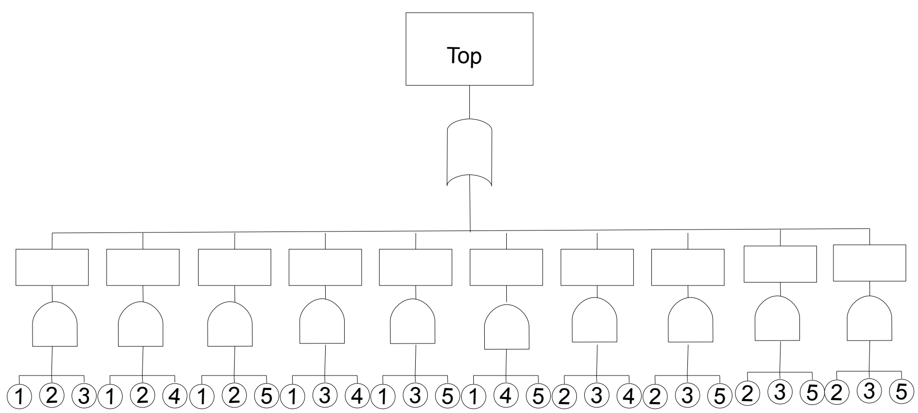


This a series system. Notice that component 3 is not essential for the cleaning of one's teeth, so it can be left out of the diagram.

For additional reading om the diagrams discussed in this chapter I recommend *System Reliability Theory* by Rausand and Høyland.

5.2 Draw the reliability block diagram and fault tree corresponding to a 3-of-5 system.





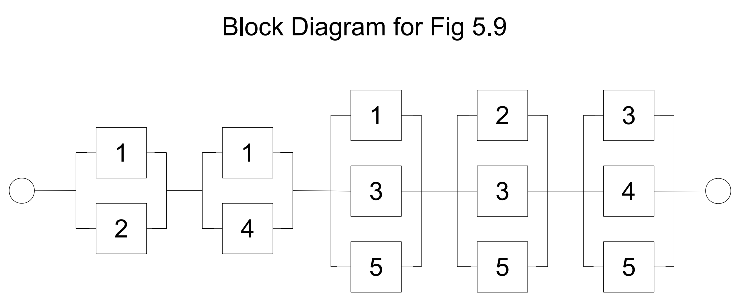
5.3 Determine the structure function for a 3-of-5 system.

The structure for a k-of-n system is given by equation 5.1.

$$\begin{aligned}
\phi(\mathbf{x}) &= x_1x_2x_3(1-x_4)(1-x_5) + x_1x_2x_4(1-x_3)(1-x_5) \\
&\quad + x_1x_2x_5(1-x_3)(1-x_4) + x_1x_3x_4(1-x_2)(1-x_5) \\
&\quad + x_1x_3x_5(1-x_2)(1-x_4) + x_1x_4x_5(1-x_2)(1-x_3) \\
&\quad + x_2x_3x_4(1-x_1)(1-x_5) + x_2x_3x_5(1-x_1)(1-x_4) \\
&\quad + x_2x_4x_5(1-x_1)(1-x_3) + x_3x_4x_5(1-x_1)(1-x_2) \\
&\quad + x_1x_2x_3x_4(1-x_5) + x_1x_2x_3x_5(1-x_4) \\
&\quad + x_1x_2x_4x_5(1-x_3) + x_1x_3x_4x_5(1-x_2) \\
&\quad + x_2x_3x_4x_5(1-x_1) + x_1x_2x_3x_4x_5 \\
&= x_1x_2x_3 + x_1x_2x_4 + x_1x_2x_5 + x_1x_3x_4 + x_1x_3x_5 + x_1x_4x_5 \\
&\quad + x_2x_3x_4 + x_2x_3x_5 + x_3x_4x_5 - 3x_1x_2x_3x_4 - 3x_1x_2x_3x_5 \\
&\quad - 3x_1x_2x_4x_5 - 3x_1x_3x_4x_5 - 3x_2x_3x_4x_5 + 6x_1x_2x_3x_4x_5
\end{aligned}$$

5.4 Draw the reliability block diagram corresponding to Fig. 5.9.

Using the 5 minimal cut sets we might draw the block diagram as



5.5 Determine the minimal path sets and minimal cut sets for IE6 in Fig. 5.9. Calculate the structure function for IE6.

The minimal cut sets are {BE2, BE3, BE5}, {BE3, BE4, BE5}. The minimal path sets are {BE2, BE4}, {BE3}, {BE5}. To determine the structure for IE6 we can use either equations 5.3 or 5.4. Using equation 5.4 with the 3 minimal path sets we get

$$\begin{aligned}
\phi(\mathbf{x}) &= 1-(1-x_2x_4)(1-x_3)(1-x_5) \\
&= x_3 + x_5 + x_2x_4 - x_3x_5 - x_2x_3x_4 - x_2x_4x_5 + x_2x_3x_4x_5
\end{aligned}$$

5.6 Define the *structural importance* of component i in a coherent system of n components as

$$I_\phi(i) = \frac{1}{2^{n-1}} \sum_{\mathbf{x}\,|\,x_i=1} [\phi(1_i,\mathbf{x}) - \phi(0_i,\mathbf{x})].$$

The sum is over the 2^{n-1} vectors for which $x_i = 1$. Calculate the structural importance of each component in Fig. 5.5.

For component 1

$(\cdot, x_2, x_3)$	$\phi(1, x_2, x_3) - \phi(0, x_2, x_3)$
$(\cdot 00)$	0
$(\cdot 01)$	1
$(\cdot 10)$	1
$(\cdot 11)$	1

$$I_\phi(1) = \frac{3}{2^{3-1}} = \frac{3}{4}$$

For component 2

$(x_1, \cdot, x_3)$	$\phi(x_1, 1, x_3) - \phi(x_1, 0, x_3)$
$(0 \cdot 0)$	0
$(0 \cdot 1)$	0
$(1 \cdot 0)$	1
$(1 \cdot 1)$	0

$$I_\phi(2) = \frac{1}{4}$$

For component 3

$(x_1, x_2, \cdot)$	$\phi(x_1, x_2, 1) - \phi(x_1, x_2, 0)$
$(00 \cdot)$	0
$(01 \cdot)$	0
$(10 \cdot)$	1
$(11 \cdot)$	0

$$I_\phi(3) = \frac{1}{4}$$

5.7 Derive Eq. 5.8 from Eq. 5.1 by assuming that each component has reliability $R_i(t) = R(t)$.

Beginning with equation (5.1),

$$P(\phi(x) = 1) = P(\sum_j (\prod_{i \in A_j} x_i)[\prod_{i \in A_j^c} (1 - x_j)] = 1)$$

We want to choose the subset A_j that is a minimum path set (i.e. $\phi(x) = 1$ for the elements in A_j). Therefore, we want at least k elements of A_j to be 1. Let s be the number of elements in A_j equal to 1. Therefore,

$$P((\prod_{i \in A_j} x_i)[\prod_{i \in A_j^c} (1 - x_j)] = 1) = P(s \geq k) = \sum_{s=k}^{n} \binom{n}{s} R(t)^s (1 - R(t))^{n-s} = \cdots$$

$$\cdots = 1 - \sum_{s=0}^{k-1} \binom{n}{s} R(t)^s (1 - R(t))^{n-s}$$

5.8 Calculate the hazard function for a series system with n components when each component lifetime has a Weibull distribution.

Let $C_i \sim Weibull(\lambda_i, \beta_i)$. By definition, the hazard function is $h_s(t) = \frac{f_s(t)}{R_s(t)}$. Using example 5.6 and $R_s = \prod_{i=1}^{n} R_i$, the hazard function is $h_s(t) = \sum_{i=1}^{n} \lambda_i \beta_i t_i^{\beta_i - 1}$

5.9 Show that the mean time to failure (MTTF) for a standby system with perfect switching is equal to the sum of the MTTFs for each component:

$$MTTF_S = \sum_{i=1}^{n} MTTF_i.$$

$MTTF_s = E[T_s] = E[T_1 + T_2 + \cdots + T_n] = E[T_1] + E[T_2] + \cdots + E[T_n] = \sum_{i=1}^{n} MTTF_i$

5.10 Suppose that each of the n components of a standby system with perfect switching has an $Exponential(\lambda)$ distribution. Show that the lifetime of the system has a $Gamma(n, \lambda)$ distribution.

$T_i \sim \text{Exponential}(\lambda) = \text{Gamma}(1, \lambda)$. Let T_s denote the systems lifetime. Then $T_s = \sum_{i=1}^{n} T_i$. Therefore, since T_s is the sum of independent $\text{Gamma}(1, \lambda)$ random variables and using the result for gamma random variables in section B of the appendix, we have $T_s \sim \text{Gamma}(n, \lambda)$.

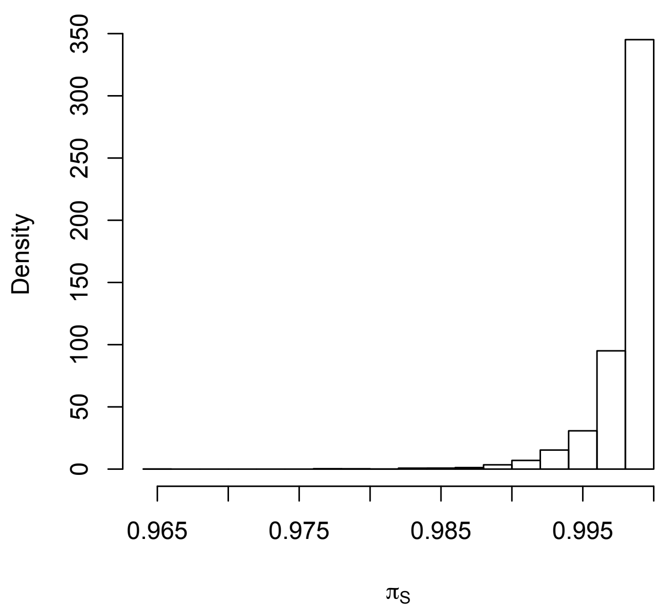
5.11 Reanalyze the data from Table 5.3 assuming that the prior distribution for the reliability of each component is $[\Gamma(1/3)]^{-1}(-\log(\pi_i))^{-\frac{2}{3}}$.

The posterior now becomes

$$p(\pi_1, \pi_2, \pi_3 \quad | \quad \mathbf{x}) \propto \pi_1^8 (1 - \pi_1)^2 \pi_2^7 (1 - \pi_2)^2 \pi_3^3 (1 - \pi_3)(\pi_1 \pi_2 \pi_3)^{10} (1 - \pi_1 \pi_2 \pi_3)^2$$
$$[-\log(\pi_1)]^{-\frac{2}{3}} [-\log(\pi_2)]^{-\frac{2}{3}} [-\log(\pi_3)]^{-\frac{2}{3}}$$

Parameter	Mean	St.Dev	Quantiles 0.025	0.050	0.500	0.950	0.975
π_1	0.868	0.074	0.699	0.732	0.878	0.965	0.976
π_2	0.861	0.077	0.690	0.720	0.871	0.968	0.978
π_3	0.887	0.082	0.690	0.733	0.904	0.987	0.992
π_S	0.998	0.002	0.992	0.994	0.999	1.000	1.000

The following is a histogram of the posterior distribution on π_s:




```
mh <- function(theta, size, data){
    pi1 = theta[1]
    pi2 = theta[2]
    pi3 = theta[3]
    s = data[,2]
    f = data[,3]
    par = array(0, dim=c(size, 4))
    arate1 = 0; arate2 = 0; arate3 = 0
    post = function(theta){
        p1 = theta[1]; p2 = theta[2]; p3 = theta[3]
        ps = theta[1]*theta[2]*theta[3]
        val = p1^s[1] * (1-p1)^f[1] * p2^s[2] * (1-p2)^f[2] * p3^s[3] *
             (1-p3)^f[3] * ps^s[4] * (1-ps)^f[4] * (-log(p1))^(-2/3) *
```

```
            (-log(p2))^(-2/3) * (-log(p3))^(-2/3)
        return(val)
    }
    for(i in 1:size){
        pi1.star = runif(1)
        r = post(c(pi1.star, pi2, pi3)) / post(c(pi1, pi2, pi3))
        u = runif(1) <= r
        arate1 = arate1 + u
        pi1 = pi1.star*(u==1) + pi1*(u==0)
        pi2.star = runif(1)
        r = post(c(pi1, pi2.star, pi3)) / post(c(pi1, pi2, pi3))
        u = runif(1) <= r
        arate2 = arate2 + u
        pi2 = pi2.star*(u==1) + pi2*(u==0)
        pi3.star = runif(1)
        r = post(c(pi1, pi2, pi3.star)) / post(c(pi1, pi2, pi3))
        u = runif(1) <= r
        arate3 = arate3 + u
        pi3 = pi3.star*(u==1) + pi3*(u==0)
        pis = 1 - (1-pi1)*(1-pi2)*(1-pi3)
        par[i,] = c(pi1, pi2, pi3, pis)
    }
    arate = c(arate1, arate2, arate3); arate = arate/size
    list = list(par = par, accept = arate)
    return(list)
}

start <- data[,2]/data[,4]
sample <- mh(start[1:3], 10000, data)
plot(as.mcmc(sample$par))

#get rid of burn-in samples - calculate summary statistics
sample$par <- sample$par[-c(1:100),]
mu <- array(apply(sample$par, 2, mean), dim=c(4,1))
st.dev <- array(apply(sample$par, 2, sd), dim=c(4,1))
quant = rbind(quantile(sample$par[,1], c(.025, .05,.5,.95,.975)),
quantile(sample$par[,2], c(.025, .05,.5,.95,.975)),
quantile(sample$par[,3], c(.025, .05,.5,.95,.975)),
quantile(sample$par[,4], c(.025, .05,.5,.95,.975)))
summary = array(c(mu, st.dev, quant), dim=c(4,7))
colnames(summary) <- c("Mean", "Std Dev", "2.5%",  "5%", "50%",
"95%", "97.5%")
rownames(summary) <- c("pi1", "pi2", "pi3", "piS")
summary
hist(sample$par[,4], freq=F, xlab=expression(pi[S]), main="")
```

5.12 There are a variety of different measures of the reliability importance of a component (Rausand and Høyland, 2003). Birnbaum's measure of importance of the ith component at time t is

$$I_B(i \mid t) = \frac{\mathrm{d}R_S(t)}{\mathrm{d}\pi_i(t)}.$$

Birnbaum's measure is the partial derivative of the system reliability with respect to each component reliability $\pi_i(t)$. A larger value of $I_B(i \mid t)$ means that a small change in the reliability of the ith component results in a comparatively large change in the system reliability. Show that in a series system, Birnbaum's measure selects the component with the lowest reliability as the most important one.

The three Birnbaum's measures are: $I_{B1} = \pi_2\pi_3$, $I_{B2} = \pi_1\pi_3$, and $I_{B3} = \pi_1\pi_2$. Without loss of generality, suppose $\pi_1 < \pi_2 < \pi_3$. Based on the description of the measure in the exercise, we are looking for the largest value, which should correspond to π_1. Therefore, by comparing the different measures: $I_{B1} = \pi_2\pi_3 > \pi_1\pi_3 = I_{B2}$ if and only if $\pi_2 > \pi_1$. Which is true by our assumption. Also, $I_{B1} = \pi_2\pi_3 > \pi_1\pi_2 = I_{B3}$ if and only if $\pi_3 > \pi_1$. Which is again true by our assumption. Therefore, I_{B1} is the largest value and the procedure selected the most important component. This result still holds if $\pi_1 \leq \pi_2 < \pi_3$. It is trivial for the case that $\pi_1 = \pi_2 = \pi_3$.

5.13 Show how to calculate the posterior distribution for π_1, π_2, and π_3 using the data in Table 5.1 using simulation and the Metropolis-Hastings algorithm.

R code for a Metropolis-Hastings algorithm:

```
mh <- function(theta, size, data){
    pi1 = theta[1]
    pi2 = theta[2]
    pi3 = theta[3]
    s = data[,2]
    f = data[,3]
    par = array(0, dim=c(size, 3))
    arate1 = 0; arate2 = 0; arate3 = 0
    post = function(theta){
        p1 = theta[1]; p2 = theta[2]; p3 = theta[3]
        val = p1^s[1]*(1-p1)^f[1]*p2^s[2]*(1-p2)^f[2]*p3^s[3]*(1-p3)^f[3]
        return(val)
    }
    for(i in 1:size){
        pi1.star = runif(1)
        r = post(c(pi1.star, pi2, pi3)) / post(c(pi1, pi2, pi3))
```

```
        u = runif(1) <= r
        arate1 = arate1 + u
        pi1 = pi1.star*(u==1) + pi1*(u==0)
        pi2.star = runif(1)
        r = post(c(pi1, pi2.star, pi3)) / post(c(pi1, pi2, pi3))
        u = runif(1) <= r
        arate2 = arate2 + u
        pi2 = pi2.star*(u==1) + pi2*(u==0)
        pi3.star = runif(1)
        r = post(c(pi1, pi2, pi3.star)) / post(c(pi1, pi2, pi3))
        u = runif(1) <= r
        arate3 = arate3 + u
        pi3 = pi3.star*(u==1) + pi3*(u==0)
        par[i,] = c(pi1, pi2, pi3)
    }
    arate = c(arate1, arate2, arate3); arate = arate/size
    list = list(par = par, accept = arate)
    return(list)
}

start <- data[,2]/data[,4]
sample <- mh(start, 10000, data)
#get rid of burn-in and calculate summary statistics
plot(as.mcmc(sample$par))
sample$par <- sample$par[-c(1:50),]
mu <- array(apply(sample$par, 2, mean), dim=c(4,1))
st.dev <- array(apply(sample$par, 2, sd), dim=c(4,1))
quant = rbind(quantile(sample$par[,1], c(.025, .05,.5,.95,.975)),
    quantile(sample$par[,2], c(.025, .05,.5,.95,.975)),
    quantile(sample$par[,3], c(.025, .05,.5,.95,.975)),
    quantile(sample$par[,4], c(.025, .05,.5,.95,.975)))
summary = array(c(mu, st.dev, quant), dim=c(4,7))
colnames(summary) <- c("Mean", "Std Dev", "2.5%",  "5%", "50%", "95%", "97.5%")
rownames(summary) <- c("pi1", "pi2", "pi3", "piS")
summary
hist(sample$par[,4], freq=F, xlab=expression(pi[S]),
    main="Marginal Posterior Distribution from M-H")
```

The posterior distributions can be found in Table 5.2 and Fig. 5.15.

R code for a simulation:

```
sim <- function(size){
    pi1 = rbeta(size, 9,3)
    pi2 = rbeta(size, 8,3)
    pi3 = rbeta(size, 4,2)
    pis = pi1*pi2*pi3
```

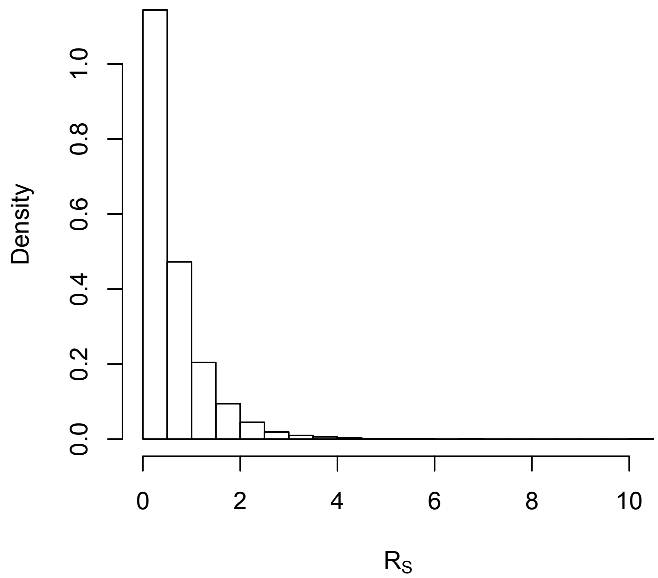
```
    pi = array(c(pi1,pi2,pi3, pis), dim=c(size,4))
    mu <- array(apply(pi, 2, mean), dim=c(4,1))
    st.dev <- array(apply(pi, 2, sd), dim=c(4,1))
    quant = rbind(quantile(pi[,1], c(.025, .05,.5,.95,.975)),
        quantile(pi[,2], c(.025, .05,.5,.95,.975)),
        quantile(pi[,3], c(.025, .05,.5,.95,.975)),
        quantile(pi[,4], c(.025, .05,.5,.95,.975)))
    summary = array(c(Mean=mu, Std.Dev=st.dev, quant), dim=c(4,7))
    colnames(summary) <- c("Mean", "Std Dev", "2.5%",  "5%", "50%",
        "95%", "97.5%")
    rownames(summary) <- c("pi1", "pi2", "pi3", "piS")
    list = list(pi = pi, summary = summary)
}
simulation <- sim(10000)
simulation$summary
hist(simulation$pi[,4], freq=F, xlab=expression(pi[S]),
    main="Marginal Distribution from Simulation")
```

			Quantiles				
Parameter	Mean	St.Dev	0.025	0.050	0.500	0.950	0.975
π_1	0.750	0.119	0.494	0.536	0.762	0.920	0.941
π_2	0.726	0.128	0.448	0.495	0.738	0.915	0.936
π_3	0.673	0.177	0.289	0.347	0.696	0.925	0.947
π_S	0.366	0.133	0.132	0.161	0.359	0.600	0.645

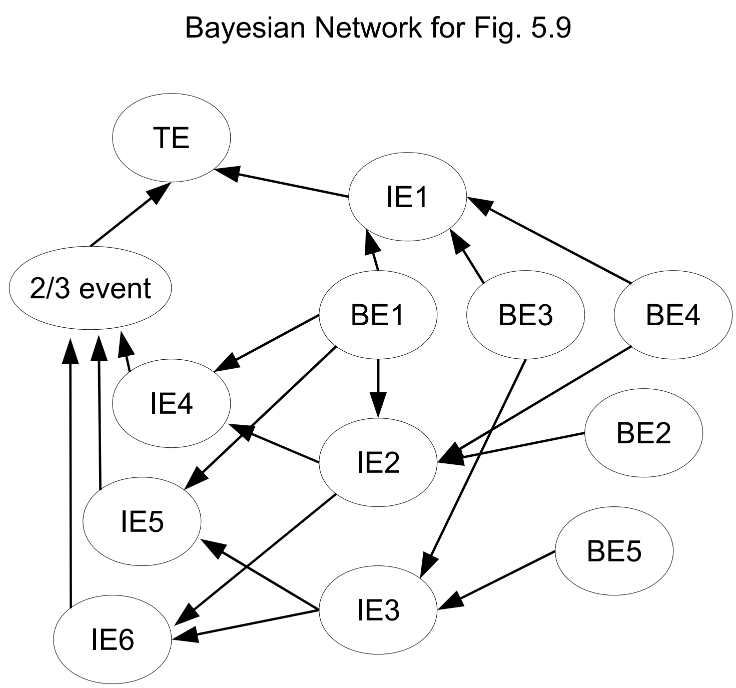
5.14 Assume a two-component series system. One component has an *Exponential*(3) prior distribution; the other has a $Weibull(5, 2)$ prior distribution. Using simulation, determine the probability density function of the prior distribution for the system.

```
r1 = rexp(10000, 3)
r2 = rweibull(10000, 5,2)
rs = r1*r2
hist(rs, freq=F, xlab = expression(R[S]), main="")
mean(rs); sd(rs)
quantile(rs, c(.025, .05,.5,.95,.975))
```

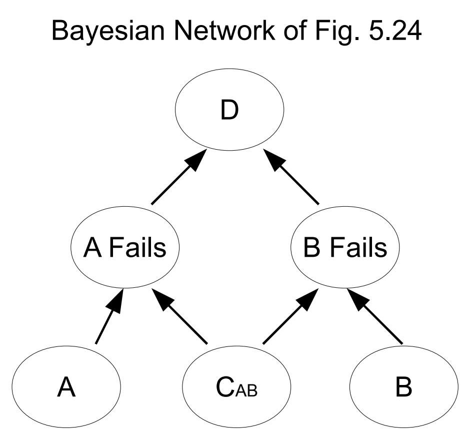
			Quantiles				
Parameter	Mean	St.Dev	0.025	0.050	0.500	0.950	0.975
R_S	0.609	0.652	0.015	0.030	0.405	1.88	2.340



5.15 Translate the fault tree in Fig. 5.9 into a BN.



5.16 Translate the fault tree in Fig. 5.24 into a BN. Write down the conditional probabilities specified by the fault tree.



$$\mathbf{P}(B_F = 0 \,|\, C_{AB} = 0, B = 0) = 1 \qquad \mathbf{P}(A_F = 0 \,|\, C_{AB} = 0, A = 0) = 1$$
$$\mathbf{P}(B_F = 0 \,|\, C_{AB} = 1, B = 0) = 1 \qquad \mathbf{P}(A_F = 0 \,|\, C_{AB} = 1, A = 0) = 1$$
$$\mathbf{P}(B_F = 0 \,|\, C_{AB} = 0, B = 1) = 1 \qquad \mathbf{P}(A_F = 0 \,|\, C_{AB} = 0, A = 1) = 1$$
$$\mathbf{P}(B_F = 0 \,|\, C_{AB} = 1, B = 1) = 0 \qquad \mathbf{P}(A_F = 0 \,|\, C_{AB} = 1, A = 1) = 0$$

$$\mathbf{P}(D = 0 \,|\, A_F = 0, B_F = 0) = 1$$
$$\mathbf{P}(D = 0 \,|\, A_F = 1, B_F = 0) = 0$$
$$\mathbf{P}(D = 0 \,|\, A_F = 0, B_F = 1) = 0$$
$$\mathbf{P}(D = 0 \,|\, A_F = 1, B_F = 1) = 0$$

5.17 Suppose that the data in Table 5.3 come from a three-component parallel system. Using independent $Uniform(0, 1)$ prior distributions for the reliability of each component, calculate the posterior distributions for the reliability of each component and the system.

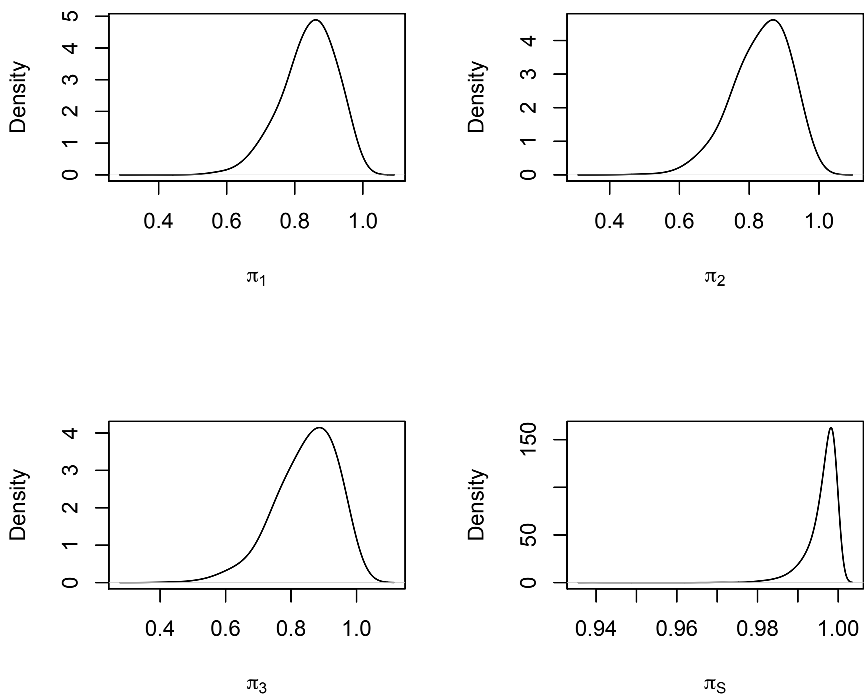
The formula for the reliability of the system in a parallel system is given on page 136. For the three component system in Table 5.3, we have

$$\pi_S = 1 - (1 - \pi_1)(1 - \pi_2)(1 - \pi_3)$$

The MCMC algorithm is similar to the one used in problems 11 and 13. Only the posterior function needs to be adjusted.

			Quantiles				
Parameter	Mean	St.Dev	0.025	0.050	0.500	0.950	0.975
π_1	0.841	0.078	0.669	0.696	0.850	0.953	0.962
π_2	0.834	0.081	0.655	0.684	0.845	0.948	0.959
π_3	0.845	0.092	0.624	0.676	0.858	0.968	0.978
π_S	0.996	0.004	0.986	0.989	0.997	1.000	1.000

Notice the difference between the posterior distribution for the reliability of this parallel system and the series system from the same data shown in Table 5.4.
The Kernel density estimates of the posterior distributions of the components and the system are shown below.



5.18 Suppose that we have a three-component system like that in Example 5.1, and suppose that each component has an $Exponential(\lambda)$ lifetime. Write an expression for the probability density function of the lifetime of the system.

$$f_i(t\,|\,\lambda) = \lambda e^{-\lambda t} \qquad F_i(t\,|\,\lambda) = 1 - e^{-\lambda t}$$

The reliability of the system can be derived combining equations 5.5 and 5.7.

$$\begin{aligned}
R_S(t) &= R_1(1-(1-R_2)(1-R_3)) \\
&= R_1R_2 + R_2R_3 - R_1R_2R_3 \\
&= 2R^2 - R^3 \qquad\qquad \text{(since } R_1 = R_2 = R_3\text{)} \\
1 - F_s(t) &= 2(1-F(t))^2 - (1-F(t))^3 \\
\frac{d}{dt}(1 - F_s(t)) &= \frac{d}{dt}\left[1 - F(t) - F^2(t) + F^3(t)\right] \\
f_s(t) &= f(t) + 2f(t)F(t) - 3f(t)F^2(t) \\
&= \lambda e^{-\lambda t} + 2\lambda e^{-\lambda t}(1-e^{-\lambda t}) - 3\lambda e^{-\lambda t}(1-e^{-\lambda t})^2 \\
&= 4\lambda e^{-2\lambda t} - 3\lambda e^{-3\lambda t}
\end{aligned}$$

5.19 Reanalyze the BN in Fig. 5.22 with data from Tables 5.8 and 5.9 assuming that we have also observed 20 observations with $C_1 = 0, C_2 = 1, C_3 = 1$ that resulted in 6 system successes and 14 system failures.

We this information we can add $\pi_{FSS}^6(1-\pi_{FSS})^{14}$ to the likelihood and the posterior becomes

$$
\begin{aligned}
p(\pi_1, \pi_2, \pi_3 \quad | \quad & \mathbf{x}) \propto \pi_1^8(1-\pi_1)^2\pi_2^7(1-\pi_2)^2\pi_3^3(1-\pi_3)\pi_{FSS}^6(1-\pi_{FSS})^{14}\pi_S^{10}(1-\pi_S)^2 \\
& [-\log(\pi_1)]^{-\frac{2}{3}}[-\log(\pi_2)]^{-\frac{2}{3}}[-\log(\pi_3)]^{-\frac{2}{3}} \\
& I[\pi_{FSS} \in (0.35, 0.85)]
\end{aligned}
$$

Using the Metropolis-Hastings algorithm given below we obtain

			Quantiles				
Parameter	Mean	St.Dev	0.025	0.050	0.500	0.950	0.975
π_1	0.82	0.10	0.59	0.63	0.83	0.95	0.97
π_2	0.78	0.12	0.51	0.56	0.80	0.95	0.96
π_3	0.78	0.16	0.41	0.47	0.80	0.97	0.98
π_{FSS}	0.42	0.06	0.35	0.35	0.41	0.54	0.57
π_S	0.80	0.05	0.68	0.71	0.81	0.88	0.89

With this new information the 95% credible interval for π_{FSS} has narrowed from (.36, 0.84) in the example in the text to (0.35, 0.57).

```
mh <- function(theta, size, data){
    pi1 = theta[1]
    pi2 = theta[2]
    pi3 = theta[3]
    pifss = theta[4]
    s = data[,2]
    f = data[,3]
    par = array(0, dim=c(size, 5))
    arate1 = 0; arate2 = 0; arate3 = 0; arate4 = 0
    post = function(theta){
        p1 = theta[1]; p2 = theta[2]; p3 = theta[3]; pFSS = theta[4]
        ps = 0.95*p1*p2*p3 + 0.8*p1*p2*(1-p3) + 0.85*p1*(1-p2)*p3 +
            0.5*p1*(1-p2)*(1-p3) + pFSS*(1-p1)*p2*p3 + 0.4*(1-p1)*
            p2*(1-p3) + 0.55*(1-p1)*(1-p2)*p3 + 0.05*(1-p1)*(1-p2)*
            (1-p3)
        val = p1^s[1] * (1-p1)^f[1] * p2^s[2] * (1-p2)^f[2] * p3^s[3] *
            (1-p3)^f[3] * pFSS^6 * (1-pFSS)^14 * ps^s[4] *
            (1-ps)^f[4] * (-log(p1))^(-2/3) * (-log(p2))^(-2/3) *
            (-log(p3))^(-2/3)
        return(val)
    }
```

```
    for(i in 1:size){
        pi1.star = runif(1)
        r = post(c(pi1.star, pi2, pi3, pifss)) /
            post(c(pi1, pi2, pi3, pifss))
        u = runif(1) <= r
        arate1 = arate1 + u
        pi1 = pi1.star*(u==1) + pi1*(u==0)
        pi2.star = runif(1)
        r = post(c(pi1, pi2.star, pi3, pifss)) /
            post(c(pi1, pi2, pi3, pifss))
        u = runif(1) <= r
        arate2 = arate2 + u
        pi2 = pi2.star*(u==1) + pi2*(u==0)
        pi3.star = runif(1)
        r = post(c(pi1, pi2, pi3.star, pifss)) /
            post(c(pi1, pi2, pi3, pifss))
        u = runif(1) <= r
        arate3 = arate3 + u
        pi3 = pi3.star*(u==1) + pi3*(u==0)
        pifss.star = runif(1,.35,.85)
        r = post(c(pi1, pi2, pi3, pifss.star)) /
            post(c(pi1, pi2, pi3, pifss))
        u = runif(1) <= r
        arate4 = arate4 + u
        pifss = pifss.star*(u==1) + pifss*(u==0)
        pis = 0.95*pi1*pi2*pi3 + 0.8*pi1*pi2*(1-pi3) + 0.85*pi1*
            (1-pi2)*pi3 + 0.5*pi1*(1-pi2)*(1-pi3) + pifss*(1-pi1)*
            pi2*pi3 + 0.4*(1-pi1)*pi2*(1-pi3) + 0.55*(1-pi1)*(1-pi2)*
            pi3 + 0.05*(1-pi1)*(1-pi2)*(1-pi3)
        par[i,] = c(pi1, pi2, pi3, pifss, pis)
    }
    arate = array(c(arate1, arate2, arate3, arate4), dim=c(1,4))
    colnames(arate) = c("pi1", "pi2", "pi3", "piFSS")
    arate = arate/size
    return(list(par = par, accept = arate))
}
start <- c(data[1:3,2]/data[1:3,4], 6/14)
sample <- mh(start, 10000, data)
```

5.20 In Example 5.7, determine the probability that the item fails because of risk 1.

The probability that the item fails because of risk 1 is given by

$$\begin{aligned}
\mathbf{P}(T_1 < T_2) &= \int_0^\infty \mathbf{P}(T_2 > t \,|\, T_1 = t) f_{T_1} dt \\
&= \int_0^\infty e^{-\lambda_2 t} \lambda_1 e^{-\lambda_1 t} dt \\
&= \lambda_1 \int_0^\infty e^{-t(\lambda_1+\lambda_2)} dt \\
&= \frac{\lambda_1}{\lambda_1 + \lambda_2}
\end{aligned}$$

Solutions to Selected Chapter 6 Exercises

6.1 From $\Lambda(T_i) - \Lambda(T_{i-1}) \sim \mathit{Exponential}(1)$, where $\Lambda(T_i) = \lambda T_i$ for the exponential renewal process, we obtain $\Lambda(T_i) - \Lambda(T_{i-1}) = \lambda T_i - \lambda T_{i-1} = \lambda(T_i - T_{i-1}) \sim \mathit{Exponential}(1)$. Consequently, $T_i - T_{i-1} \sim \frac{1}{\lambda}\mathit{Exponential}(1) = \mathit{Exponential}(\lambda)$.

6.2 For failure times $t_1, t_2, \ldots, t_n$, $T_1, T_2 - T_1, \ldots, T_n - T_{n-1}$ are i.i.d. $\mathit{Gamma}(\alpha, \lambda)$. The corresponding likelihood is proportional to

$$[\lambda^\alpha/\Gamma(\alpha)]^n \left(\prod_{i=1}^{n}(t_i - t_{i-1})^{\alpha-1}\right)\exp(-\lambda t_n)\,,$$

where $t_0 = 0$. Under Type-I censoring at t_c, the corresponding likelihood is proportional to

$$[\lambda^\alpha/\Gamma(\alpha)]^n \left(\prod_{i=1}^{n}(t_i - t_{i-1})^{\alpha-1}\right)\exp(-\lambda t_n)\int_{t_c - t_n}^{\infty} f(t|\alpha,\lambda)\mathrm{d}t\,,$$

where $f(t|\alpha, \lambda)$ is the $\mathit{Gamma}(\alpha, \lambda)$ probability density function. Type-II censoring is not relevant for a single repairable system.

6.4 A 95% credible interval for κ is (0.473, 1.120), so that the data suggest that there is no need to use the MPLP over the PLP, where κ equals 1. Accounting for the censored observation, $26^{0.4} \approx 4$. Using $K = 4$ equal probability bins, we find that 1.3% of the R^B test statistics exceed the 0.95 quantile of the $\mathit{ChiSquared}(3)$ reference distribution, which suggests no lack of fit.

6.8 If the last failure occurred at t^*, we have $\Lambda(T + t^*) - \Lambda(t^*) \sim \mathit{Gamma}(\kappa, 1)$ so that $R_{t^*}(t) = \mathbf{P}(T > t)$, but

$$T > t \equiv \Lambda^{-1}\left(\Lambda(t*) + X\right) - t^* > t \equiv X > \Lambda(t + t^*) - \Lambda(t^*)\,,$$

where $X \sim \mathit{Gamma}(\kappa, 1)$. Consequently,

$$R_{t^*}(t) = \int_{\Lambda(t+t^*)-\Lambda(t^*)}^{\infty} f(x|\kappa, 1)\mathrm{d}t\,,$$

where $f(x|\kappa, 1)$ is the $\mathit{Gamma}(\kappa, 1)$ probability density function.

6.20 Assuming that uptime $U \sim \mathit{Gamma}(\alpha_U, \lambda_U)$ and downtime $D \sim \mathit{Gamma}(\alpha_D, \lambda_D)$, then the long-run availability $A = \frac{E(U)}{E(U)+E(D)}$ is

$$\frac{\alpha_U/\lambda_U}{\alpha_U/\lambda_U + \alpha_D/\lambda_D}\,.$$

Solutions to Selected Chapter 7 Exercises

7.2 We fit a logistic regression model in heating time and soaking time. We use the actual heating and soaking times (centered by their respective averages) as covariates in a full second order model, i.e., X_1, X_2, X_1X_2, X_1^2, and X_2^2, where X_1 and X_2 are the centered heating and soaking times, respectively. The regression coefficient of X_2 has a high posterior probability (0.992) of being different from zero, i.e., has an impact. The coefficients for X_2^2, X_1^2, and X_1X_2 are less important with posterior probabilities of 0.938, 0.920, and 0.908, respectively, of being different from zero. See the following reference for more details: B.P. Weaver and M.S. Hamada (2008), "A Bayesian Approach to the Analysis of Industrial Experiments: An Illustration with Binomial Count Data," *Quality Engineering*, 20, 269–280.

7.3 Using $K = 4 \approx 63^{0.4}$ equal probability bins, we find that 4.3% of the R^B test statistics exceed the 0.95 quantile of the $ChiSquared(3)$ reference distribution, which suggests no lack of fit.

7.14 a) We fit the model

$$Y_i \sim Poisson(\lambda_i t_i)\,,$$

where

$$\begin{aligned}\log(\lambda_i) &= \beta_1 + \beta_2 SYS2_i + \beta_3 SYS3_i + \beta_4 SYS4_i + \beta_5 SYS5_i \\ &\quad + \beta_6 OTY2_i + \beta_7 OTY3_i + \beta_8 OTY4_i \\ &\quad + \beta_9 VTY2_i + \beta_{10} VTY3_i + \beta_{11} VTY4_i + \beta_{12} VTY5_i + \beta_{13} VTY6_i \\ &\quad + \beta_{14} SIZ2_i + \beta_{15} SIZ3_i + \beta_{16} OPM2_i\,.\end{aligned}$$

The covariates in the Poisson regression model above are dummy variables, e.g., $SYS2_i = 1$ if $SYS = 2$, etc. The summaries of the posterior distributions for the βs are given in the table below.

b) A 90% credible upper bound on the predicted number of failures in the next 10 years for a normally closed (OPM=1) 2- 10-inch (SIZ=2) air-driven (OTY=1) globe valve (VTY=5) in a power conversion system (SYS=3) is 99.

Parameter	Mean	Std Dev	Quantiles		
			0.025	0.500	0.975
β_1	-10.000	0.781	-11.740	-9.971	-8.618
β_2	0.9023	0.5226	-0.1079	0.8903	1.9800
β_3	1.030	0.490	0.121	1.012	2.065
β_4	1.214	0.551	0.161	1.204	2.343
β_5	0.317	0.572	-0.766	0.311	1.469
β_6	0.5966	0.5961	-0.6806	0.6254	1.6900
β_7	-1.213	0.248	-1.700	-1.211	-0.719
β_8	-2.560	0.497	-3.610	-2.529	-1.672
β_9	0.2025	0.7742	-1.3010	0.1882	1.7550
β_{10}	0.6059	0.7904	-0.9477	0.6105	2.1500
β_{11}	3.068	0.592	2.040	3.026	4.370
β_{12}	1.893	0.602	0.825	1.846	3.203
β_{13}	0.833	0.992	-1.246	0.875	2.677
β_{14}	-0.0039	0.2792	-0.5266	-0.0069	0.5741
β_{15}	1.625	0.316	1.021	1.618	2.262
β_{16}	-0.2065	0.1896	-0.5760	-0.2063	0.1617

7.17 For $X \sim Gamma(\alpha, \lambda)$, we can use $Y = \lambda X$ to remove λ as seen by $Y \sim Gamma(\alpha, 1)$. However, we cannot transform Y to remove α. Consequently, a Cox-Snell residual does not exist for the gamma distribution. The deviance residual for the ith observation y_i has the form:

$$\text{sign}(y_i - \mu_i)\sqrt{2[-\log(y_i - \mu_i) + (y_i - \mu_i)/\mu_i]}\,,$$

where $\mu_i = \alpha_i/\lambda_i^2$. See McCullagh and Nelder (1989) for more details.

7.24 Based on the predictive distribution summaries displayed in the table below, PCB type copper-tin-lead at 20°C is the recommended factor-level combination with the longest lifetime distribution.

PCB Type	Temp (° C)	Quantiles		
		0.025	0.500	0.975
copper-nickel-tin	20	251.2	700.2	1233.2
	60	55.0	157.9	278.5
	100	51.6	143.9	253.3
copper-nickel-gold	20	570.5	1636.9	2869.4
	60	89.8	533.9	929.1
	100	142.3	403.6	708.6
copper-tin-lead	20	587.4	1702.2	3031.9
	60	168.1	469.5	844.3
	100	130.5	372.8	646.8

Solutions to Selected Chapter 8 Exercises

8.1 We use the model $y_{ij} = -5 + \beta_i x_j + \epsilon$, where $\beta_i \sim Normal(\mu_\beta, \sigma_\beta^2)$ and $\epsilon \sim Norma(0, \sigma^2)$, for the jth observation of the ith LED. See the table below for posterior summaries of the model parameters. Using $K = 5$ equal probability bins, we find that 33.6% of the R^B test statistics exceed the 0.95 quantile of the $ChiSquared(4)$ reference distribution, which suggests some lack of fit. Using a threshold of -2.0 at 300 hours, a 90% credible interval for reliability is (0.3343, 0.7514) with a posterior median of 0.5505.

			Quantiles		
Parameter	Mean	Std Dev	0.025	0.500	0.975
β_1	0.004246	5.482E-4	0.003169	0.004245	0.005317
β_2	0.007738	5.528E-4	0.006640	0.007746	0.008833
β_3	0.007844	5.571E-4	0.006747	0.007838	0.008937
β_4	0.007461	5.543E-4	0.006385	0.007464	0.008572
β_5	0.009606	5.523E-4	0.008527	0.009608	0.010710
β_6	0.006429	5.500E-4	0.005347	0.006427	0.007494
β_7	0.007638	5.613E-4	0.006531	0.007635	0.008756
β_8	0.008719	5.525E-4	0.007629	0.008722	0.009819
β_9	0.011470	5.591E-4	0.010380	0.011470	0.012580
μ_β	0.007862	0.006230	-0.004482	0.007914	0.020440
σ	0.203600	0.024980	3.036E-4	0.200900	0.260200
σ_β	0.017710	0.005276	0.010830	0.016750	0.030620

8.4 Using $K = 9$ equal probability bins, we find that 32.5% of the R^B test statistics exceed the 0.95 quantile of the $ChiSquared(8)$ reference distribution, which shows lack of fit, as compared with the original model in Example 8.2, whose assessment is given below as the solution to Exercise 8.16. Consequently, we prefer the original model and do not use the alternative model to calculate $R(t)$ and $t_{0.1}$.

8.14 Using $K = 22$ equal probability bins, we find that 3.8% of the R^B test statistics exceed the 0.95 quantile of the $ChiSquared(21)$ reference distribution, which suggests no lack of fit.

8.16 Here, we consider only the fit of the model in Example 8.2. Using $K = 9$ equal probability bins, we find that 12.9% of the R^B test statistics exceed the 0.95 quantile of the $ChiSquared(8)$ reference distribution, which suggests no lack of fit.

8.17 It is doubtful that there is a closed form expression for the destructive measurement $z_i = D_i(t)$ with $D_i(t)$ defined in Eq. 8.21 when it is measured with a $Normal(0, \sigma^2)$ measurement error. Instead, we derive Eq. 8.23, the probability density function of z_i without measurement error as follows. We have $z = D(t) = \beta_0 - \beta_1(1/x)t$, where $x \sim Lognormal(\mu, \sigma^2)$, where the probability density function of x is

$$f(x \mid \mu, \sigma^2) \quad = \quad \frac{1}{x\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}(\log(x) - \mu)^2\right].$$

We obtain the probability desntiy function of z by a change of variables. Because x can be expressed in terms of z by $x = \frac{\beta_1 t}{\beta_0 - z}$, we see that $\mathrm{d}x = \frac{\beta_1 t}{(\beta_0 - z)^2}\mathrm{d}z$. Consequently,

$$f(z \mid \mu, \sigma^2, \beta_0, \beta_1, t) \quad = \quad \frac{(\beta_1 t)}{(\beta_0 - z)^2} \frac{1}{\frac{\beta_1 t}{\beta_0 - z}\sqrt{2\pi\sigma^2}} \exp\left[-\frac{1}{2\sigma^2}\left(\log\left(\frac{\beta_1 t}{\beta_0 - z}\right) - \mu\right)^2\right].$$

which simplifies to Eq. 8.23.

Selected Solutions to Chapter 9 Exercises

9.1 Under a $Beta(1,1)$ prior, the 0.90 posterior quantile of the 95% credible interval length for a sample size of 377 is 0.075. The 0.90 posterior quantile of the 95% credible interval length for a sample size of 3000 is 0.036, so that we use $n_{max} = 3000$. A bisection search finds that for a sample size of 2695, the 0.90 posterior quantile of the 95% credible interval length does not exceed 0.0375. That is $n_{optimal} = 2695$.

9.2 For (a) 5 out of 10 successes, $n_{optimal} = 1531$; (b) 50 out of 100 successes, $n_{optimal} = 1462$; (c) 9 out of 10 successes, $n_{optimal} = 1219$; and (d) 90 out of 100 successes, $n_{optimal} = 643$.

9.3 Under $X \sim Poisson(\lambda t)$, with a $Gamma(1,1)$ prior for λ, $\alpha = 0.95$, $\gamma = 0.90$, $L_{target} = 0.05$, and $t_{max} = 100000$, we find using a bisection search that $t_{optimal} = 14339$. We can carry out the testing by allocating the total testing time of 14339 across multiple units that can be tested simultaneously, assuming that all the test units have a common λ.

9.4 Consider the data collection planning example, which focuses on reliability at time $t = 24$ months. Assuming that the lifetimes have a $LogNormal(\mu, \sigma^2)$ distribution, we use the following prior distributions: $\mu \sim Normal(4, 0.1)$ and $\sigma^2 \sim InverseGamma(20, 10)$. Letting $\gamma = 0.90$, $\alpha = 0.95$, and $L_{target} = 0.1$, we find that a sample size $n_{max} = 500$ meets the stated requirement, i.e., the probability of the $\alpha \times 100\%$ credible interval length of $R(24)$ not exceeding L_{target} is at least γ. A bisection search yields $n_{optimal} = 83$. The example in the chapter used prior distributions, which yielded a reliability prior distribution at 24 months with a median of 0.50 and a 0.95 probability interval of (0.00, 1.00). The prior distributions used in this exercise yielded a reliability prior distribution at 24 months with a median of 0.88 and a 0.95 probability interval of (0.79, 0.94).

9.7 We use the following less diffuse priors distributions:

$$\begin{aligned}
&\beta_0 \sim Normal(-7.3, (0.15/5)^2), \\
&\beta_1 \sim Normal(7.5, (0.15/5)^2), \text{ and} \\
&\sigma^2 \sim InverseGamma(100 \times 5, 0.11^2 \times 100 \times 5).
\end{aligned}$$

For $\gamma = 0.9$, $\alpha = 0.95$, $L_{target} = 0.1$, and reliability at time $t = 10{,}500$ days, we check to make sure that the planning criterion for $n_i = 400, i = 1, 2, 3$ and $v_2 = 0.5$ meets the requirement. We use a GA that minimizes the total sample size, $n_1 + n_2 + n_3$, where instead of discarding (n_1, n_2, n_3) cases, which have a planning criterion ρ (i.e., the probability that the $\alpha \times 100\%$ credible interval length does not exceed L_{target}) that does not exceed γ, we penalize these cases by minimizing:

$$n_1 + n_2 + n_3 + [100(\gamma - \rho)/0.01]I(\rho < \gamma) + [25(\gamma - \rho)/0.01]I(\rho > \gamma).$$

A GA found the nearly optimal solution $v_{2,optimal} = 0.496$ and $\mathbf{n}_{optimal} = (n_1, n_2, n_3)$, where $n_1 = 145$, $n_2 = 50$, and $n_3 = 189$, whose penalized criterion is 384.