STAT 466 HW 7

Brody Anderson

2024-03-01

Question 1

I read pages 85-120 of Chapter 4 from the textbook.

Question 2

a. Obtain 100,000 draws from the posterior (show histograms of the marginals for λ and β).

```
fluid <- c(5.45,16.46,15.70,10.39,6.71,3.77,7.42,6.89,9.45,5.89,7.39,5.61,16.55,
          12.63,8.18,10.44,6.03,13.96,5.19,10.96, 14.73,6.21,5.69,8.18,4.49,3.71,
          5.84,10.97,6.81,10.16, 4.34,9.81,4.30,8.91,10.07,5.85,4.95,7.30,4.81,
          8.44,6.56,9.40,11.29,12.04,1.24,3.45,11.28,6.64,5.74,6.79)

y <- fluid
#  GoF[i] <- pweib(y[i], v, lambda)


Fluid_model <- "model {
  for (i in 1:50) { y[i] ~ dweib(v, lambda);
  }

  v <- a
  lambda <- (1/b)^a

  a ~ dgamma(0.1, 0.1)
  b ~ dunif(0, 10)

  PPD ~ dweib(v, lambda)

}"

Fluid.sim <- jags( data = c('y'), parameters.to.save = c('v', 'lambda', 'PPD'), #'GoF' not incl.
  model.file = textConnection(Fluid_model),
  n.iter = 22000,
  n.burnin=2000,
  n.chains=5,
  n.thin=1
)
```

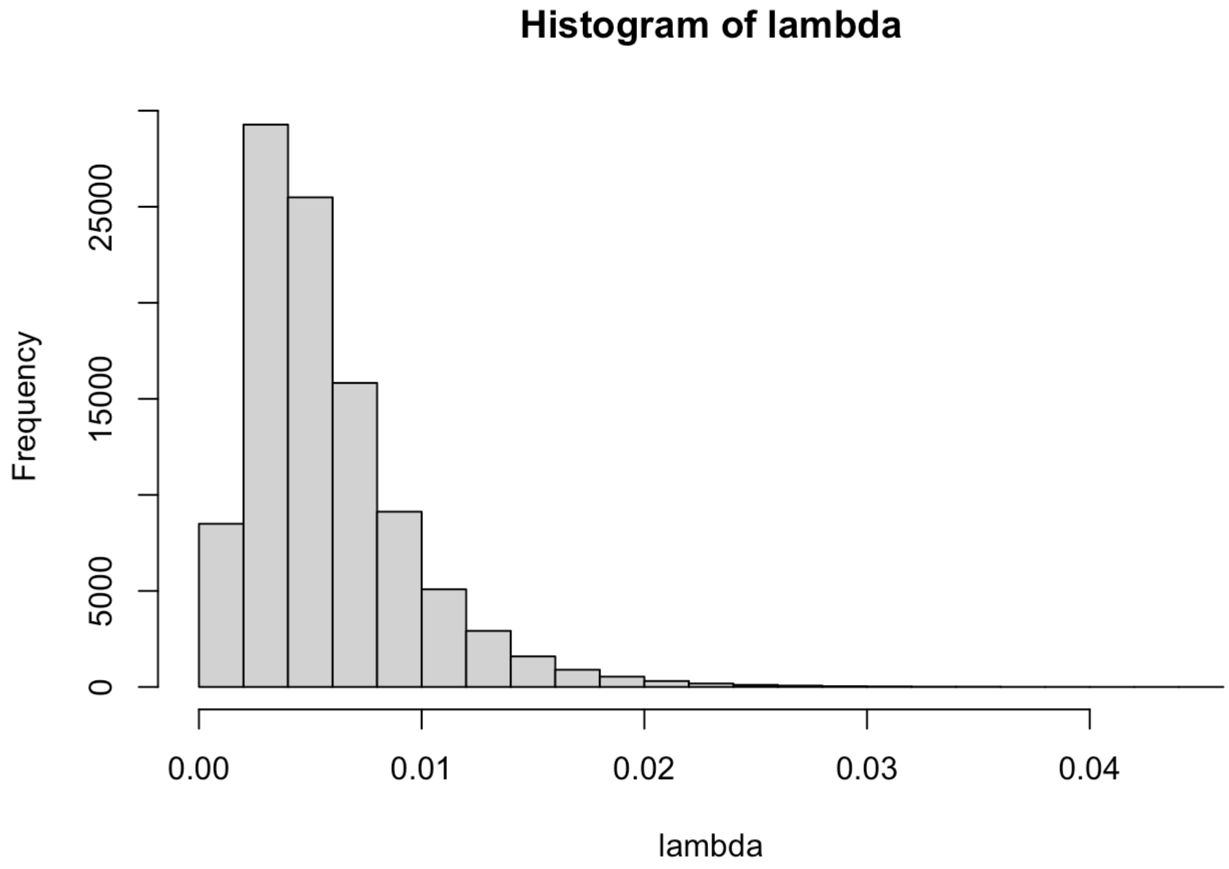
```
## module glm loaded
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 50
##    Unobserved stochastic nodes: 3
##    Total graph size: 59
##
## Initializing model
```

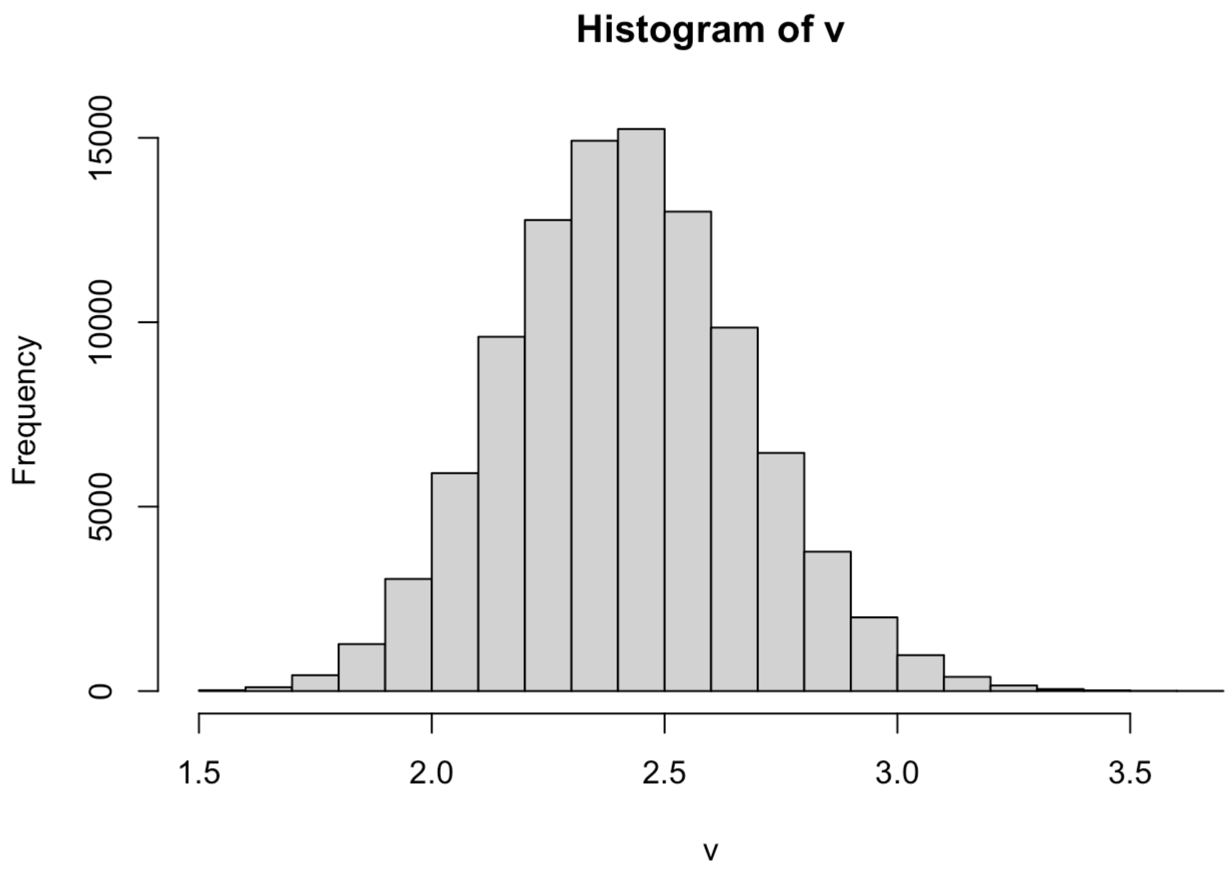
```
# 100,000 samples
# ((252000 - 2000)*4)/10 = 100,000 lambda^(-1/a)

head(Fluid.sim$BUGSoutput$sims.matrix)
```

```
##            PPD deviance      lambda        v
## [1,] 12.750757 264.7583 0.004049428 2.426964
## [2,] 15.758839 268.0232 0.013318647 1.956878
## [3,]  7.467797 264.6138 0.005532995 2.312509
## [4,]  8.617473 265.5266 0.004632394 2.506053
## [5,] 13.123542 264.5580 0.002376904 2.695870
## [6,]  2.429932 264.8302 0.004201003 2.529095
```

```
lambda <- Fluid.sim$BUGSoutput$sims.matrix[,3]
v <- Fluid.sim$BUGSoutput$sims.matrix[,4]
#beta <- (Fluid.sim$BUGSoutput$sims.matrix[,3])^(-(1/Fluid.sim$BUGSoutput$sims.matrix[,4])) #text beta

#Histograms
hist(lambda) # lambda
```




```
hist(v) # v or beta
```

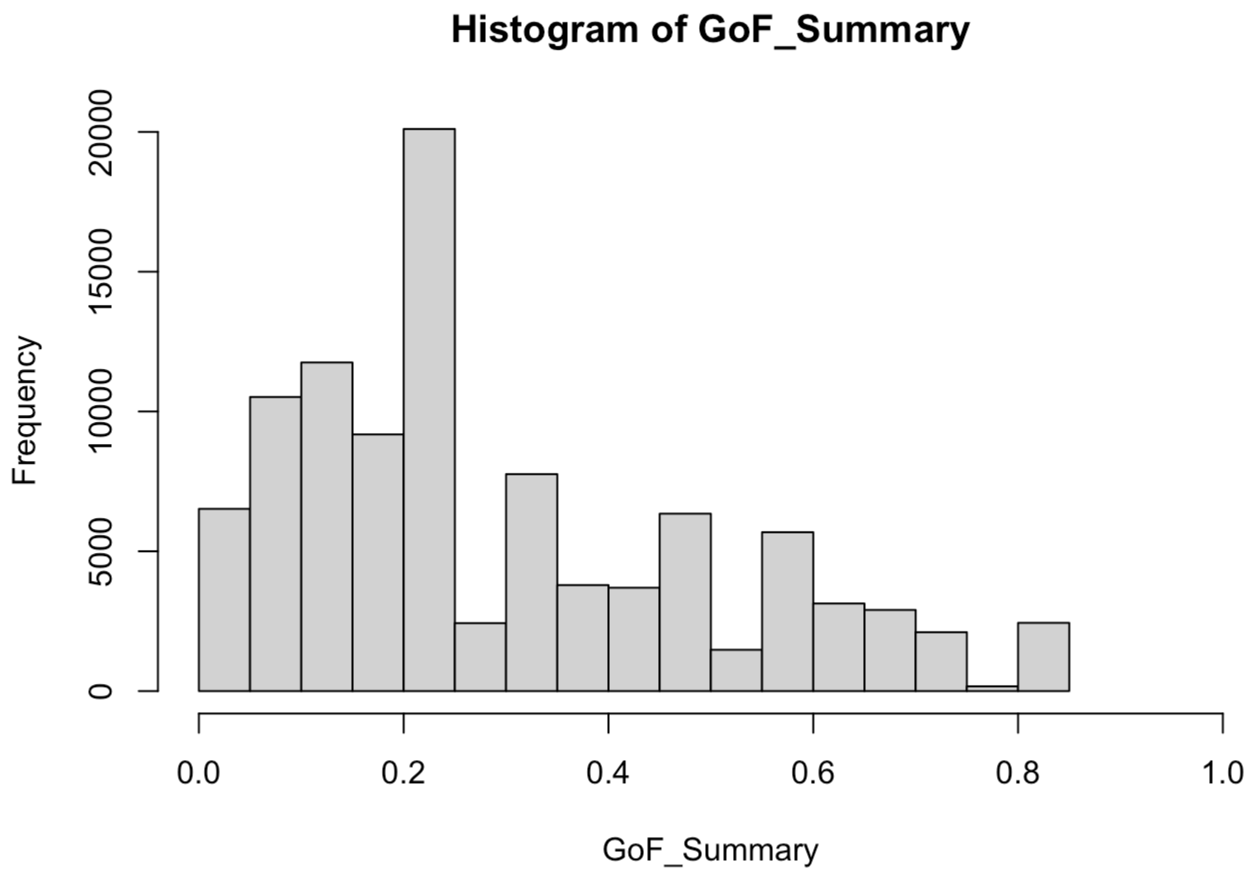


b. Using the Bayesian χ2 goodness-of-fit test determine if the model is an adequate representation of the data.

```
head(Fluid.sim$BUGSoutput$sims.matrix)
```

```
##            PPD deviance      lambda        v
## [1,] 12.750757 264.7583 0.004049428 2.426964
## [2,] 15.758839 268.0232 0.013318647 1.956878
## [3,]  7.467797 264.6138 0.005532995 2.312509
## [4,]  8.617473 265.5266 0.004632394 2.506053
## [5,] 13.123542 264.5580 0.002376904 2.695870
## [6,]  2.429932 264.8302 0.004201003 2.529095
```

```
chains <- as.mcmc(Fluid.sim$BUGSoutput$sims.matrix)
a <- as.mcmc(chains[,4])
b <- as.mcmc((chains[,3])^(-1/a))


GoF <- matrix(NA,ncol=length(fluid),nrow=length(chains[,3]))
for (i in 1:length(chains[,3])) {
  GoF[i,] <- pweibull(fluid, a[i], b[i])
}

# Function requires fitted quantiles and returns a p-value
GoF_Test <- function(fitted_quantiles) {
  n <- length(fitted_quantiles)
  K <- round((n)^(0.4))
  mK <- table(cut(fitted_quantiles,(0:K)/K))
  np <- n/K
  RB <- sum(((mK-np)^2)/np)
  return(1-pchisq(RB,K-1))
}

# Calculating the p-values for each posterior model
GoF_Summary <- apply(GoF,1,GoF_Test)

# Histogram of posterior model p-values
hist(GoF_Summary,xlim=c(0,1))
```




```
# Percent of posterior models with p-value less than 0.05
mean(GoF_Summary < 0.05)
```

```
## [1] 0.06518
```

The model is not a bad representation of the data. We have a roughly 6-ish% of posterior models with a p-value less than 0.05. We would like to see roughly 5%, but 6% is pretty good. I'd say the model is good.

c. Report a posterior point estimate of the MTTF and also a 95% probability interval.

```
fluid.mttf <- b*gamma(1+(1/a))
fluid.point.estimate <- mean(fluid.mttf)
fluid.prob.interval <- quantile(fluid.mttf, c(0.025,0.975))
```

With .95 probability, the mean time to failure for lubricating fluid is between 7.1695787 and 8.8057276 thousands of hours, with our best guess at 8.0778155 thousand hours.

d. What is the posterior probability that the process has an increasing hazard function?

```
mean(b > 1)
```

```
## [1] 1
```

```
#summary(b)
#hist(b)
```

b is always greater than 1, which means the posterior probability that the process has an increasing hazard function is 1.

e. Using the posterior predictive distribution estimate the probability that a fluid breakdown time will be less than 4 thousand hours.

```
results <- table(Fluid.sim$BUGSoutput$sims.matrix[,1][Fluid.sim$BUGSoutput$sims.matrix[,1] < 4])
est <- sum(results) / length(Fluid.sim$BUGSoutput$sims.matrix[,1])

#mean(Fluid.sim$BUGSoutput$sims.matrix[,1] < 4)
```

probability less than 4 thousand hours: 0.13101

Question 3

Reanalyze the plant 63 EDG demand data in Example 4.1 using a uniform prior distribution for the probability of successful start π. Compare the results with those obtained in Example 4.1 in terms of the point estimate, credible interval, and length of the credible interval.

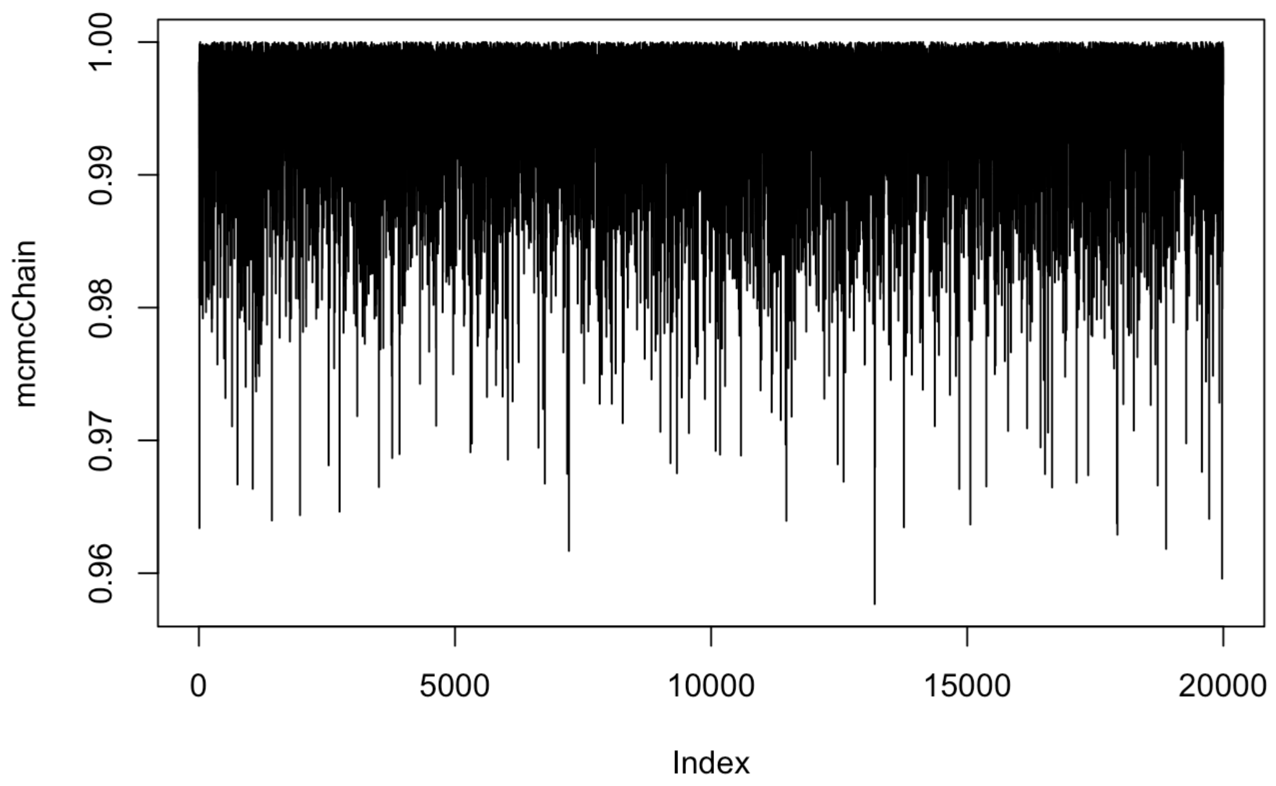
```
q <- c(rep(1, 212))

EDG.model <- "model {
  for(i in 1:212){
    q[i] ~ dbin(pi, 1);
  }
  pi ~ dbeta(1,1)

}
"

EDG.sim <- jags(
  data=c('q'),
  parameters.to.save=c('pi'),
  model.file=textConnection(EDG.model),
  n.iter=12000,
  n.burnin=2000,
  n.chains=2,
  n.thin=1
)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 212
##    Unobserved stochastic nodes: 1
##    Total graph size: 214
##
## Initializing model
```

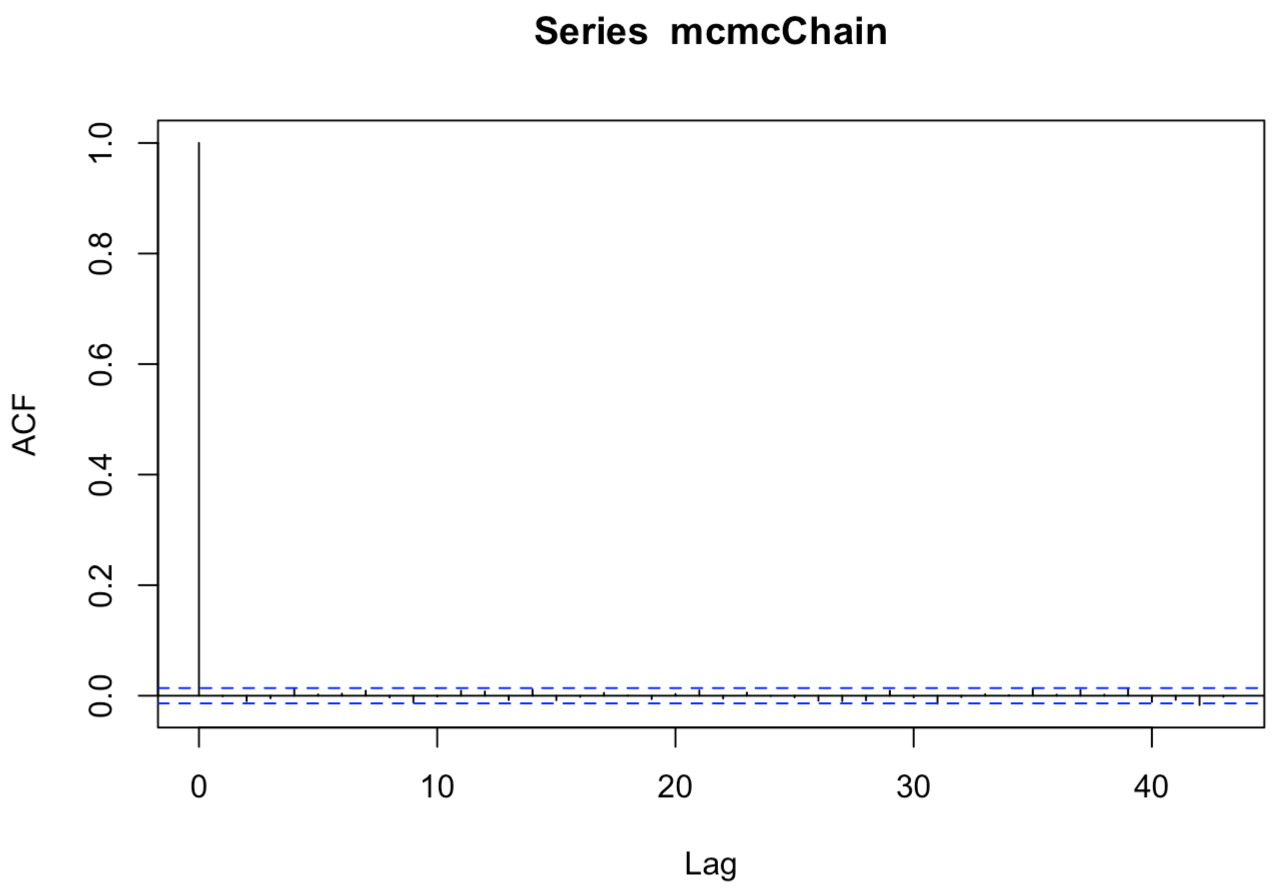
```
head(EDG.sim$BUGSoutput$sims.matrix)
```

```
##       deviance        pi
## [1,] 0.7695174 0.9981867
## [2,] 1.6053306 0.9962210
## [3,] 0.6516192 0.9984643
## [4,] 3.3930497 0.9920295
## [5,] 1.7318604 0.9959238
## [6,] 8.2576100 0.9807129
```

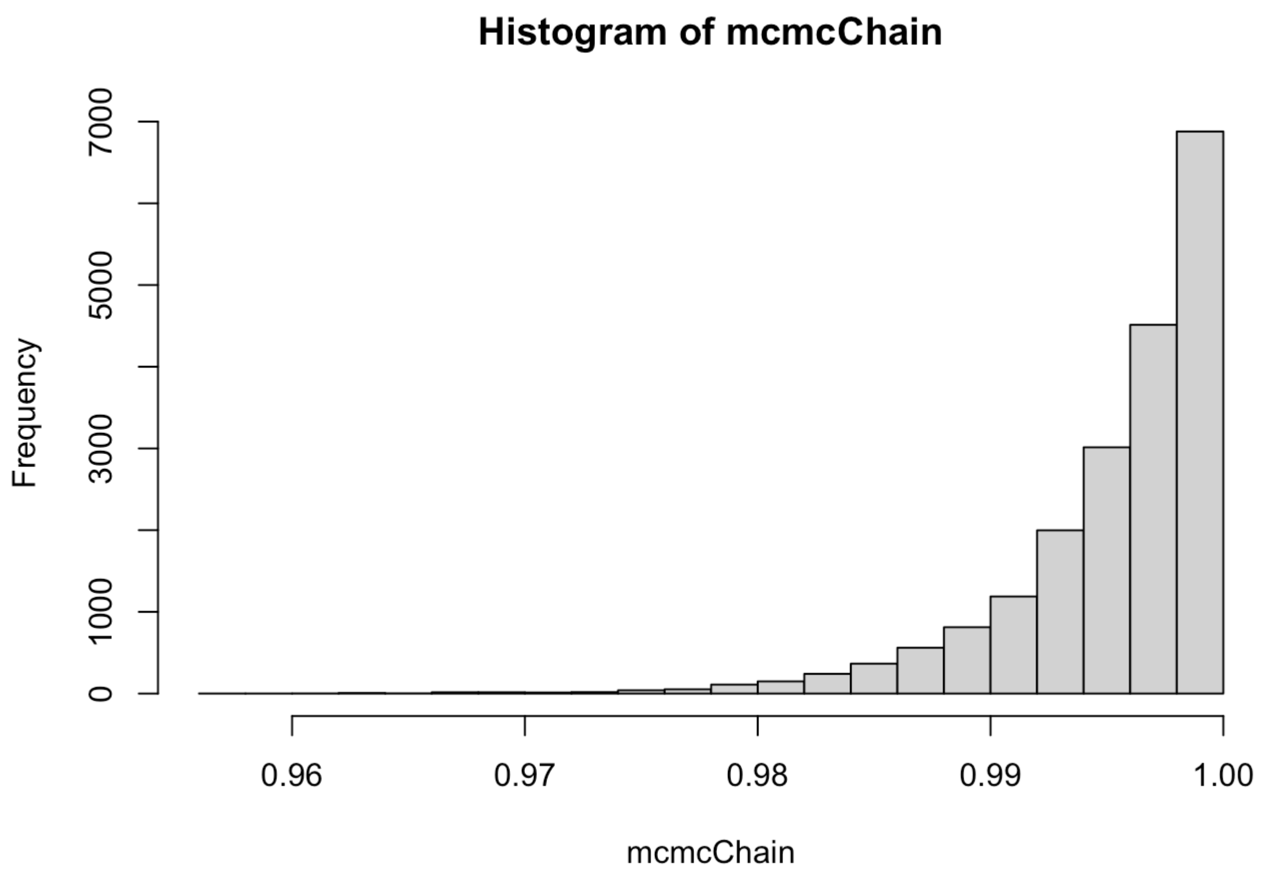
```
mcmcChain <- EDG.sim$BUGSoutput$sims.matrix[,2]

plot(mcmcChain,type="l")
```




```
acf(mcmcChain)
```




```
hist(mcmcChain,br=30)
```



```
EDG.sim$BUGSoutput$DIC
```

```
## [1] 4.033328
```

```
#summary(mcmcChain)

my.point <- mean(mcmcChain)

#95% Credible Interval
my.cred <- quantile(mcmcChain,c(0.025,0.975))

my.len <- my.len <- my.cred[2] - my.cred[1]



#prior beta(1,1)
#posterior beta(1+212, 1+212-212)

#pi <- seq(0,1,length=1001)
#plot(pi, dbeta(213, 1))

# Their analysis: Posterior Beta(239.3, 0.5)
# Our analysis: Posterior Beta(213, 1)

#pi <- seq(0, 1, length = 1001)
#plot(pi, dbeta(pi, 213, 1), type = "l", xlab = expression(pi), ylab = "Density",
#     main = "Beta Posterior")

text.point <- 239.3 / (239.3 + 0.5)
#my.point <- 213 / (213+1)

text.cred <- qbeta(c(0.025, 0.975), 239.3, 0.5)
#my.cred <- qbeta(c(0.025, 0.975), 213, 1)

text.len <- text.cred[2] - text.cred[1]
#my.len <- my.cred[2] - my.cred[1]
```

Using the uniform prior, our point estimate is closer to what the data is, meaning it is closer to one.

Textbook point estimate: 0.9979149

My point estimate: 0.9952873

Textbook point estimate: 0.989547, 0.9999979

My point estimate: 0.9825682, 0.9998871

Textbook length: 0.0104509

My length: 0.0173189

From these results, we also see that the uniform prior leads to a larger credible interval.

Question 4

Tetrahedron Inc. (1996) presents success/failure data for tests on blowout prevention (BOP) systems, which prevent uncontrolled releases of reservoir fluids in the drilling of oil and gas fields. A test is a success if no piece of equipment in the BOP system needs repair. The initial experiment consisted of 44 unused BOP systems tested with 17 successes. Estimate the initial BOP system reliability and provide a credible interval.

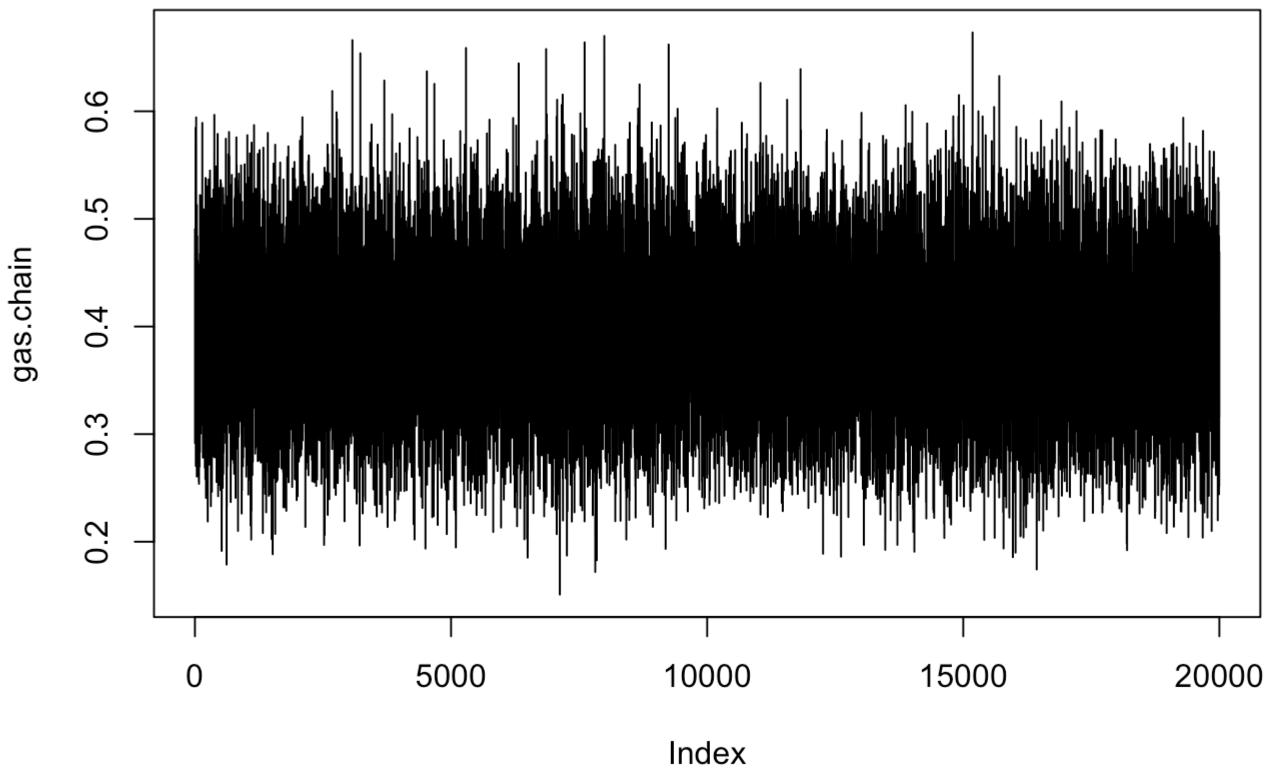
```
gas.data <- c(rep(1, 17), rep(0, 27))

Gas.model <- "model {
  for(i in 1:44){
    gas.data[i] ~ dbin(pi, 1);
  }
  pi ~ dbeta(1,1)

}
"

Gas.sim <- jags(
  data=c('gas.data'),
  parameters.to.save=c('pi'),
  model.file=textConnection(Gas.model),
  n.iter=12000,
  n.burnin=2000,
  n.chains=2,
  n.thin=1
)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 44
##    Unobserved stochastic nodes: 1
##    Total graph size: 46
##
## Initializing model
```

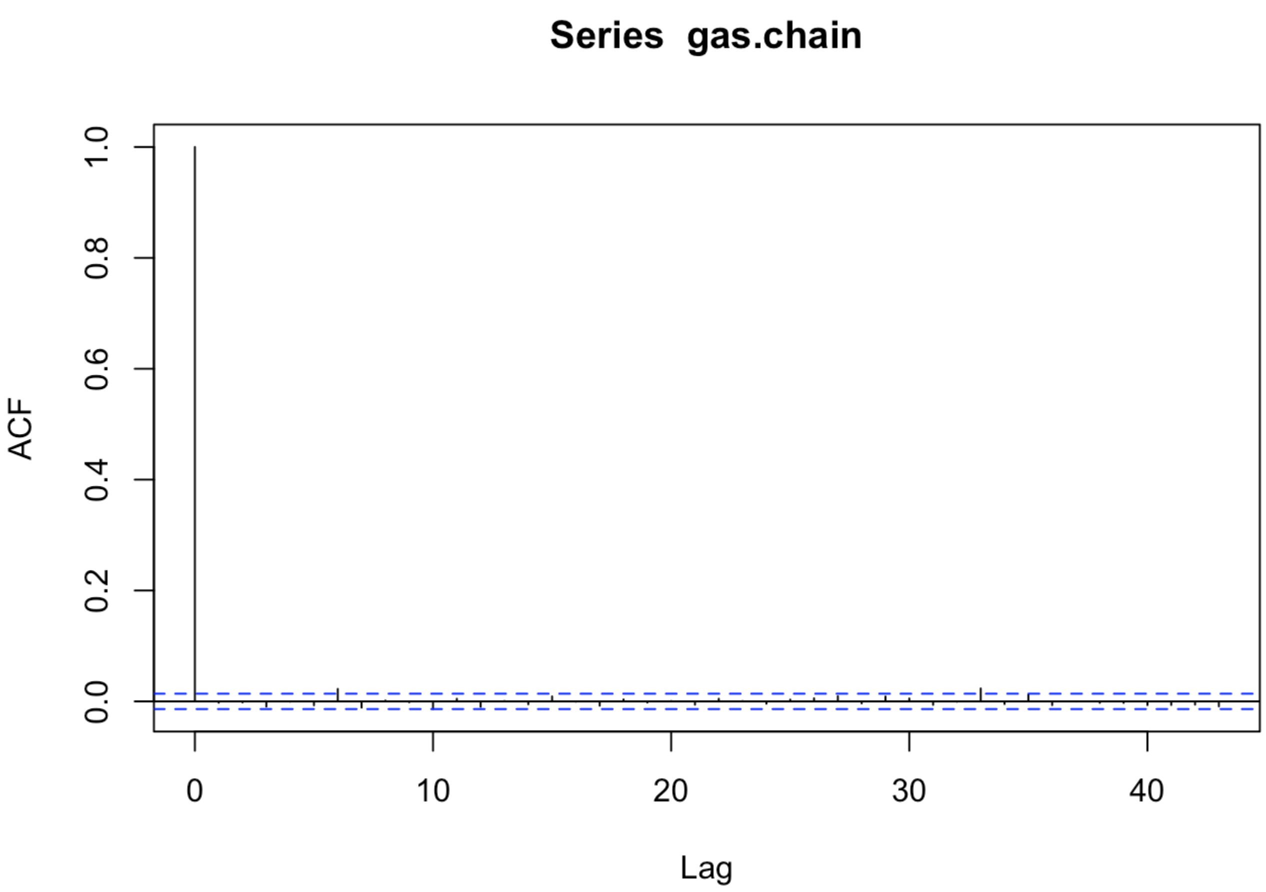
```
head(Gas.sim$BUGSoutput$sims.matrix)
```

```
##      deviance        pi
## [1,] 60.63378 0.4905270
## [2,] 60.51652 0.2916110
## [3,] 58.70471 0.3847707
## [4,] 59.97305 0.3064227
## [5,] 59.97162 0.4705876
## [6,] 58.71126 0.3802265
```

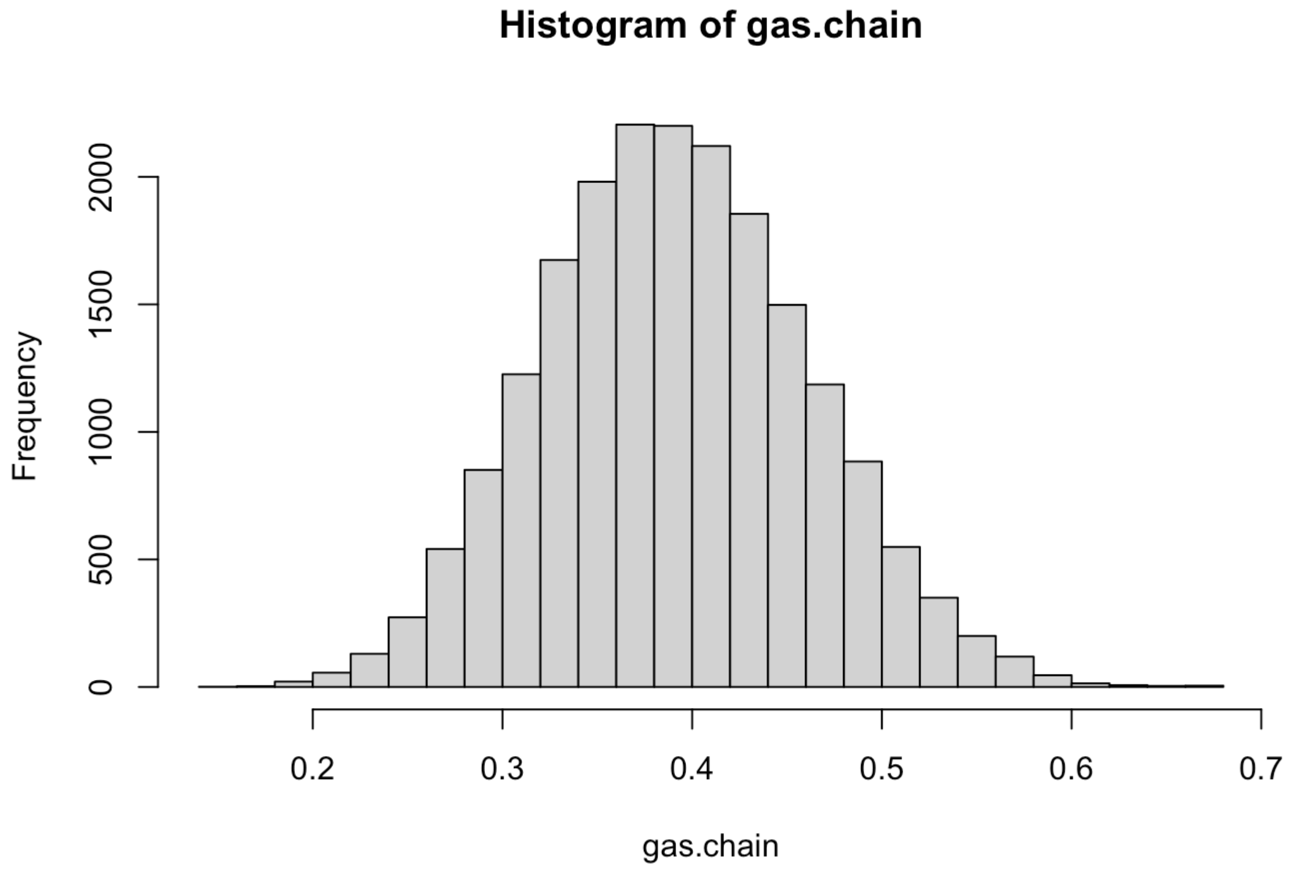
```
gas.chain <- Gas.sim$BUGSoutput$sims.matrix[,2]

plot(gas.chain,type="l")
```




```
acf(gas.chain)
```




```
hist(gas.chain,br=30)
```




```
Gas.sim$BUGSoutput$DIC
```

```
## [1] 60.49586
```

```
#summary(gas.chain)

gas.point <- mean(gas.chain)

#95% Credible Interval
gas.cred <- quantile(gas.chain,c(0.025,0.975))
```

point estimate: 0.3913869

95% credible interval: (0.2611241, 0.5319957)