Final

Charles Hwang

7/2/2022

Charles Hwang

Dr. Matthews

STAT 488-001

2 July 2022

Problem 1

Problem 1(a)

$(\lambda e^{-\lambda x})(2e^{-2x})$

Problem 1(b)

```
rm(list=ls())
x<-c(0.2,0.4,0.7,0.7,1)
set.seed(2272)
b<-"model {for (i in 1:length(x)){x[i] ~ dexp(lambda)}
lambda ~ dexp(2)
for (i in 1:length(x)) {y[i] ~ dexp(lambda)}}"
write(b,"/Users/newuser/Desktop/1b.bug")
```

Problem 1(c)

```
library(rjags)
```

```
## Loading required package: coda
```

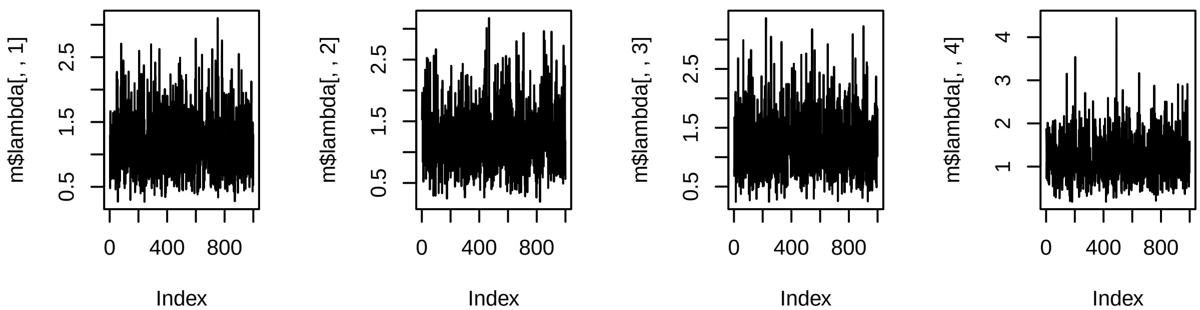
```
## Linked to JAGS 4.3.1
```

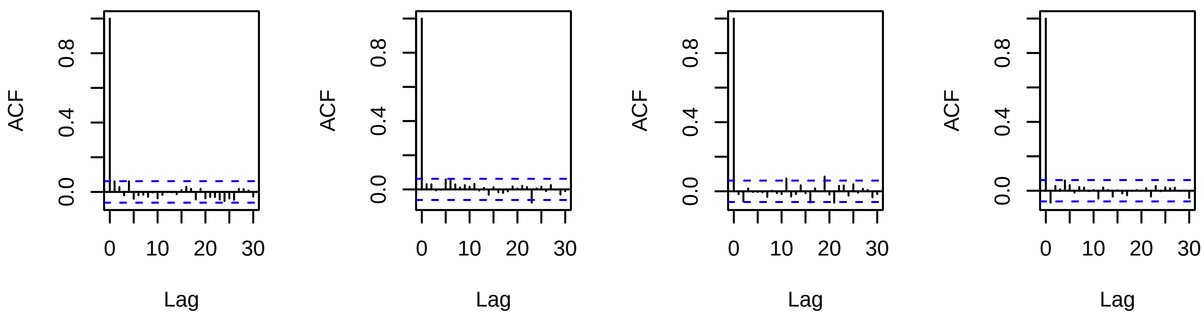
```
## Loaded modules: basemod,bugs
```

```
set.seed(2272)
jagsb<-jags.model("/Users/newuser/Desktop/1b.bug",list('x'=x),n.chains=4,n.adapt=5000)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 5
##    Unobserved stochastic nodes: 6
##    Total graph size: 12
##
## Initializing model
```

```r
update(jagsb,1000) # "Burn-in" phase
m<-jags.samples(jagsb,c('lambda','y'),1000)
par(mfrow=c(2,4))
plot(m$lambda[,,1],type="l")
plot(m$lambda[,,2],type="l")
plot(m$lambda[,,3],type="l")
plot(m$lambda[,,4],type="l")
acf(m$lambda[,,1],main="")
acf(m$lambda[,,2],main="")
acf(m$lambda[,,3],main="")
acf(m$lambda[,,4],main="")
```





We can see each of the four chains have converged and there are clearly no issues with autocorrelation in λ.

Problem 1(d)

```r
mean(m$lambda[,,1]) # Using first chain from model
```

```
## [1] 1.204493
```

```r
quantile(m$lambda[,,1],c(0.1/2,1-0.1/2))
```

```
##        5%       95%
## 0.5129261 2.0599295
```

Problem 1(e)

```r
mean(1/m$lambda[,,1]) # Using first chain from model
```

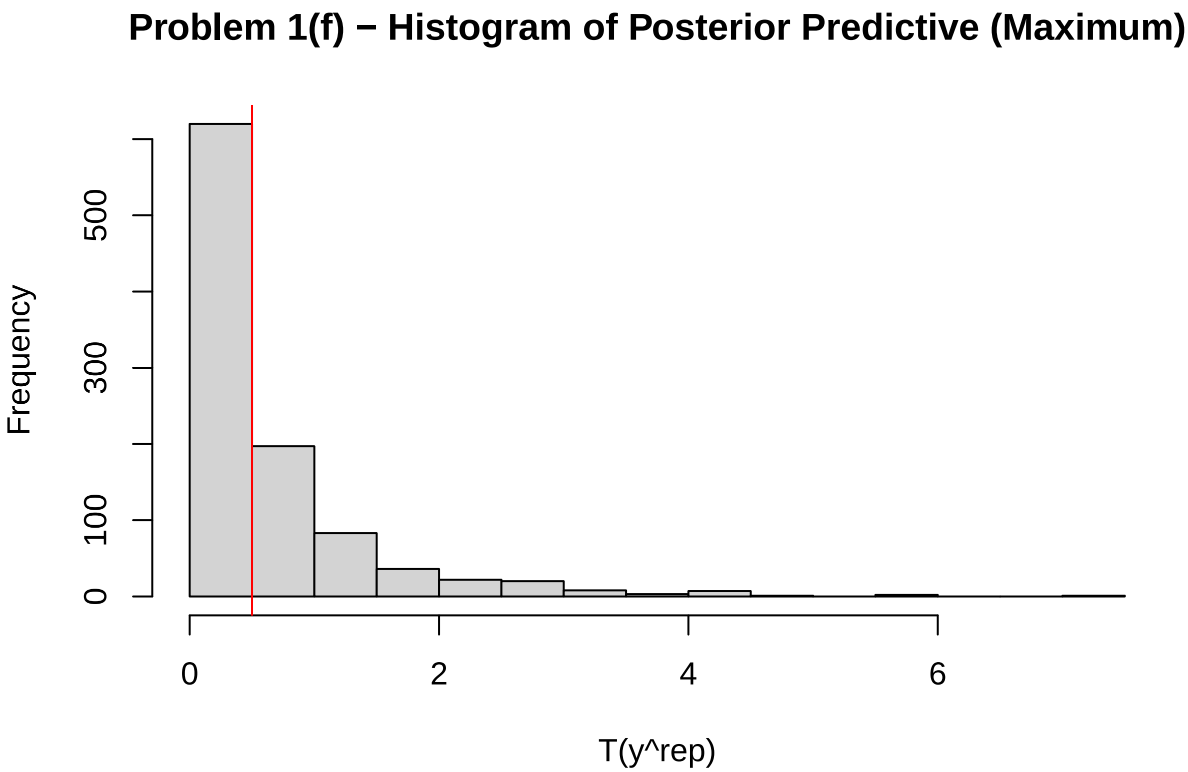
```
## [1] 0.9923419
```

```r
quantile(1/m$lambda[,,1],c(0.1/2,1-0.1/2))
```

```
##        5%       95%
## 0.4854537 1.9496177
```

Problem 1(f)

```r
set.seed(2272)
yu<-apply(replicate(1000,rexp(5,2)*rexp(1,2)),2,max)
hist(yu,20,xlab="T(y^rep)",main="Problem 1(f) - Histogram of Posterior Predictive (Maximum)")
abline(v=1/2,col="red")
```

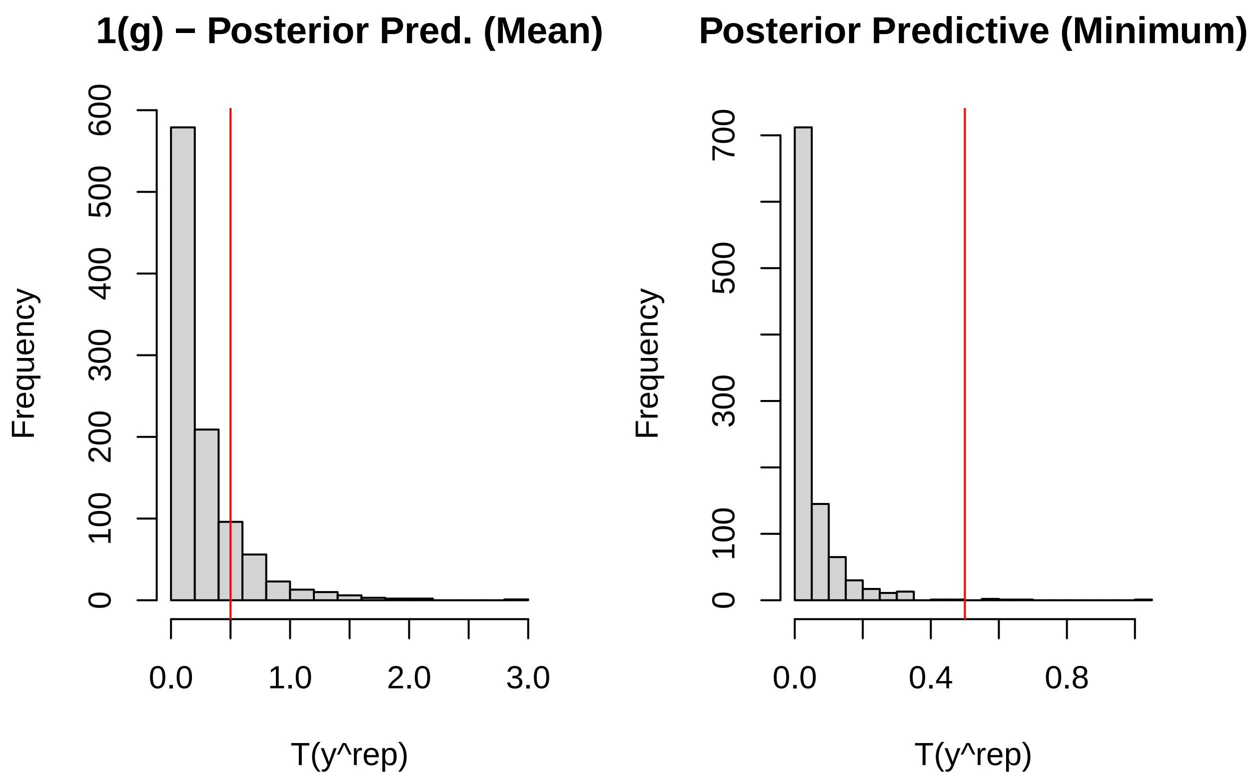



```r
mean(yu>1/2)
```

```
## [1] 0.38
```

Problem 1(g)

```r
set.seed(2272)
ym<-apply(replicate(1000,rexp(5,2)*rexp(1,2)),2,mean)
yl<-apply(replicate(1000,rexp(5,2)*rexp(1,2)),2,min)
par(mfrow=c(1,2))
hist(ym,20,xlab="T(y^rep)",main="1(g) - Posterior Pred. (Mean)")
abline(v=1/2,col="red")
hist(yl,20,xlab="T(y^rep)",main="Posterior Predictive (Minimum)")
abline(v=1/2,col="red")
```




```
mean(ym>1/2)
```

```
## [1] 0.155
```

```
mean(yl>1/2)
```

```
## [1] 0.005
```

Problem 1(h)

```
mean(m$y[,,]>1/0.5)
```

```
## [1] 0.12985
```

Problem 2

Problem 2(a)

```
c<-read.csv("/Users/newuser/Desktop/Notes/Graduate/STAT 488 - Bayesian Statistical Methods/nrippner-ols
c$medIncome<-c$medIncome/1000
```

Problem 2(b)

```
summary(c) # Output has been suppressed to conserve space.
sort(apply(is.na(c),2,sum)[apply(is.na(c),2,sum)!=0],decreasing=TRUE)
par(mfrow=c(2,4))
hist(c$TARGET_deathRate,main="",ylab="",yaxt="n")
hist(c$avgAnnCount,main="",ylab="",yaxt="n")
hist(c$avgDeathsPerYear,main="",ylab="",yaxt="n")
hist(c$incidenceRate,main="",ylab="",yaxt="n")
```

```r
hist(c$medIncome,main="",ylab="",yaxt="n")
hist(c$povertyPercent,main="",ylab="",yaxt="n")
hist(c$popEst2015,main="",ylab="",yaxt="n")
hist(c$studyPerCap,main="",ylab="",yaxt="n")
hist(c$MedianAge,main="",ylab="",yaxt="n")
hist(c$MedianAgeFemale,main="",ylab="",yaxt="n")
hist(c$MedianAgeMale,main="",ylab="",yaxt="n")
hist(c$AvgHouseholdSize,main="",ylab="",yaxt="n")
hist(c$PercentMarried,main="",ylab="",yaxt="n")
hist(c$PctMarriedHouseholds,main="",ylab="",yaxt="n")
hist(c$PctEmployed16_Over,main="",ylab="",yaxt="n")
hist(c$PctUnemployed16_Over,main="",ylab="",yaxt="n")
hist(c$PctNoHS18_24,main="",ylab="",yaxt="n")
hist(c$PctHS18_24,main="",ylab="",yaxt="n")
hist(c$PctSomeCol18_24,main="",ylab="",yaxt="n")
hist(c$PctBachDeg18_24,main="",ylab="",yaxt="n")
hist(c$PctHS25_Over,main="",ylab="",yaxt="n")
hist(c$PctBachDeg25_Over,main="",ylab="",yaxt="n")
hist(c$PctPrivateCoverage,main="",ylab="",yaxt="n")
hist(c$PctPrivateCoverageAlone,main="",ylab="",yaxt="n")
hist(c$PctEmpPrivCoverage,main="",ylab="",yaxt="n")
hist(c$PctPublicCoverage,main="",ylab="",yaxt="n")
hist(c$PctPublicCoverageAlone,main="",ylab="",yaxt="n")
hist(c$PctWhite,main="",ylab="",yaxt="n")
hist(c$PctBlack,main="",ylab="",yaxt="n")
hist(c$PctAsian,main="",ylab="",yaxt="n")
hist(c$PctOtherRace,main="",ylab="",yaxt="n")
hist(c$BirthRate,main="",ylab="",yaxt="n")
cm<-data.frame(cor(c[,c(3,1:2,4:5,7:6,8,10,12:11,14:15,33,22:23,16:21,24:32,34)],use="complete.obs"))
names(cm)<-NULL
round(cm,1)
```

The variables for "PctSomeCol18_24", "PctPrivateCoverageAlone", and "PctEmployed16_Over" contain missing values. The response variable appears to be approximately normal, along with incidence rate, median income, median age (despite a few impossible outliers), household size, percent married, percent employed and unemployed, education (except both percentages for bachelor's degree), and all coverage types. Looking at the histograms and sorting the values in each column, some quantitative variables appear to have a few impossible outliers, perhaps from errors with data entry. The correlation matrix shows that percentage of residents ages 25+ with a bachelor's degree had the strongest correlation with the response variable ($r =$ -0.4400733). The three pairs between population and annual cases and deaths from cancer were the only pairs of variables with $|r| > .95$.

Problem 2(c)

```r
y<-c$TARGET_deathRate
X<-c[,c("incidenceRate","medIncome","povertyPercent")]
n<-nrow(c)
p<-ncol(X)
a<-"model {for (i in 1:n){y[i] ~ dnorm(mu[i],sigma)
    mu[i]<-alpha+X[i,]%*%beta}
    alpha ~ dnorm(0,0.001)
    for (j in 1:p){beta[j] ~ dnorm(0,0.001)}
    sigma ~ dchisq(1)}"
```

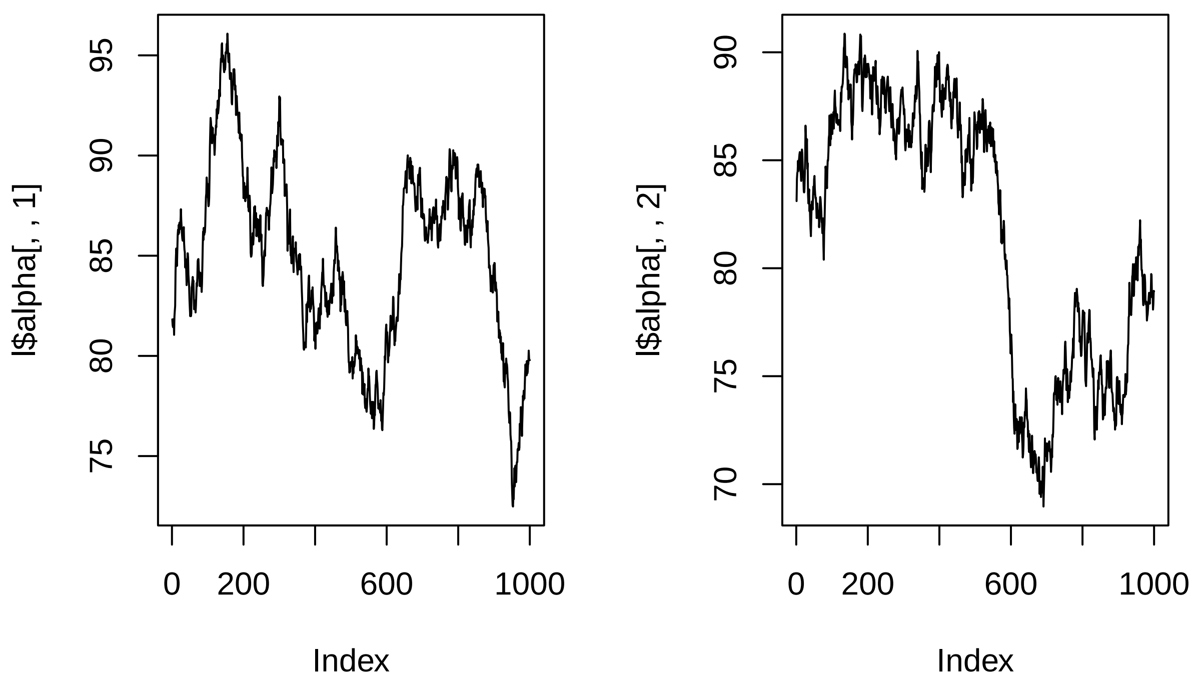
```r
write(a,"/Users/newuser/Desktop/2.bug")
set.seed(2272)
gs<-jags.model("/Users/newuser/Desktop/2.bug",list('y'=y,'X'=X,'n'=n,'p'=p),n.chains=2,n.adapt=1000)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 3047
##    Unobserved stochastic nodes: 5
##    Total graph size: 21340
##
## Initializing model
```

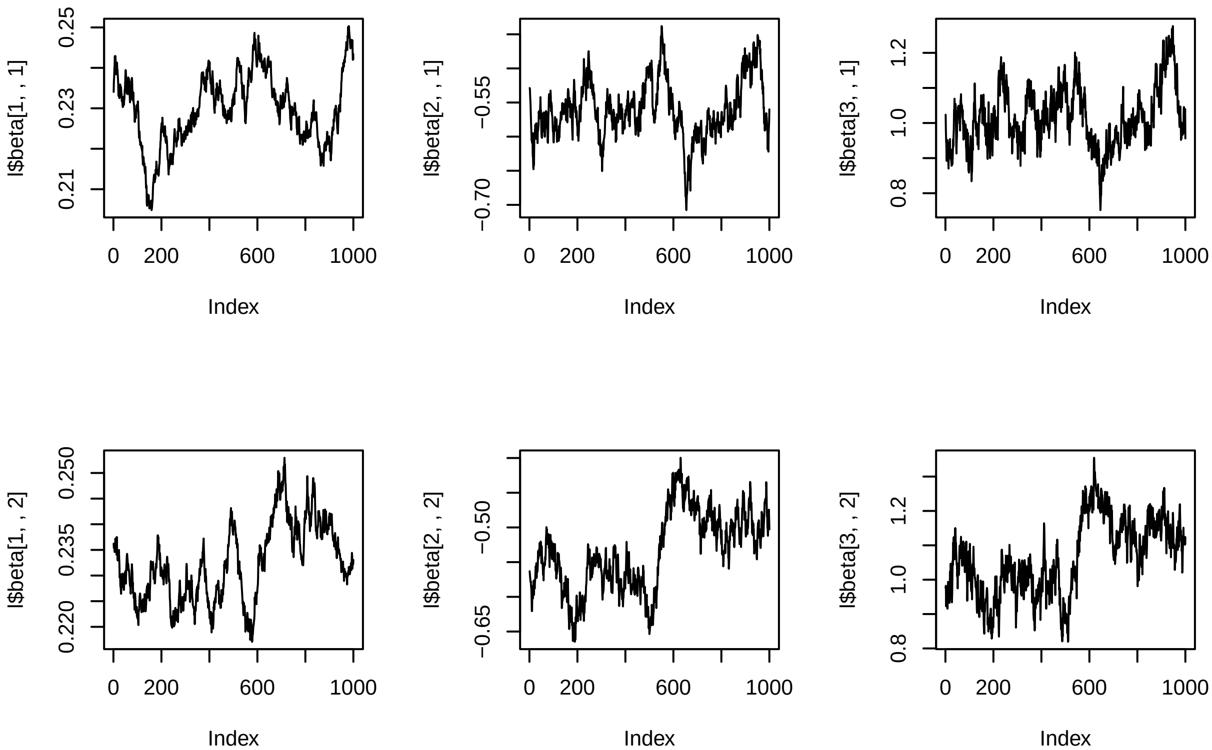
```r
update(gs,1000) # "Burn-in" phase
l<-jags.samples(gs,c('alpha','beta','sigma','y'),1000)
```

Problem 2(d)

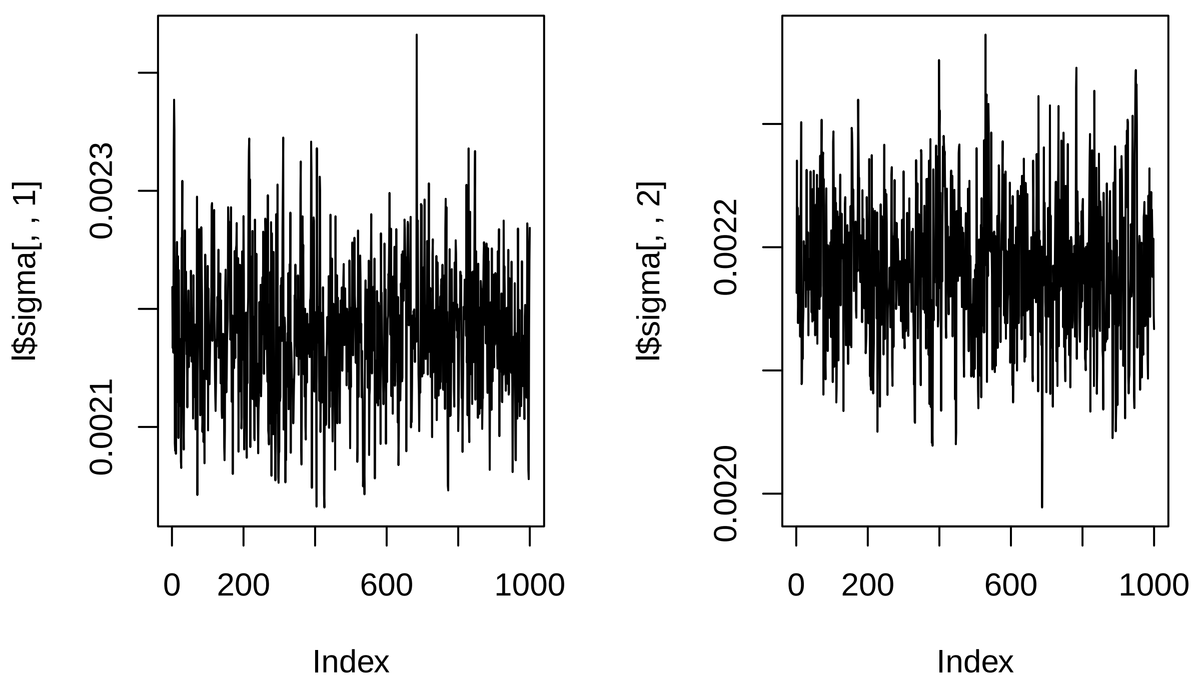
```r
par(mfrow=c(1,2))
plot(l$alpha[,,1],type="l")
plot(l$alpha[,,2],type="l")
```




```r
par(mfrow=c(2,3))
plot(l$beta[1,,1],type="l")
plot(l$beta[2,,1],type="l")
plot(l$beta[3,,1],type="l")
plot(l$beta[1,,2],type="l")
plot(l$beta[2,,2],type="l")
plot(l$beta[3,,2],type="l")
```

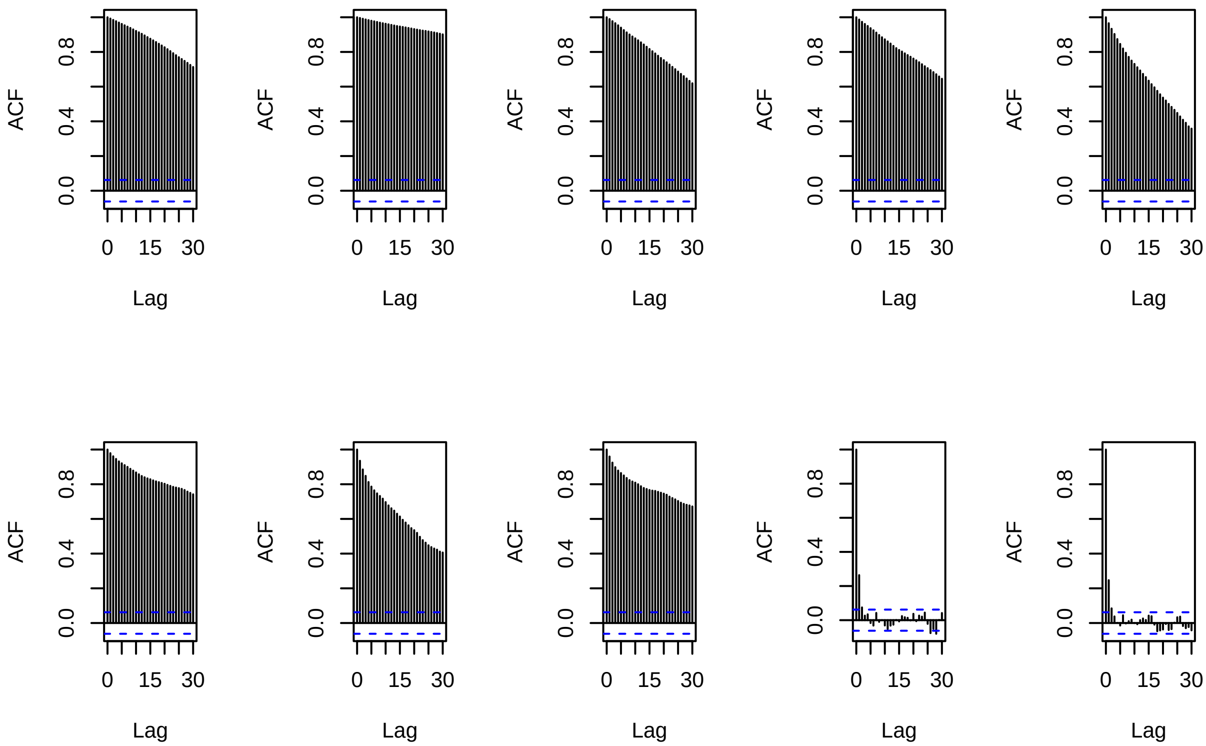



```r
par(mfrow=c(1,2))
plot(l$sigma[,,1],type="l")
plot(l$sigma[,,2],type="l")
```

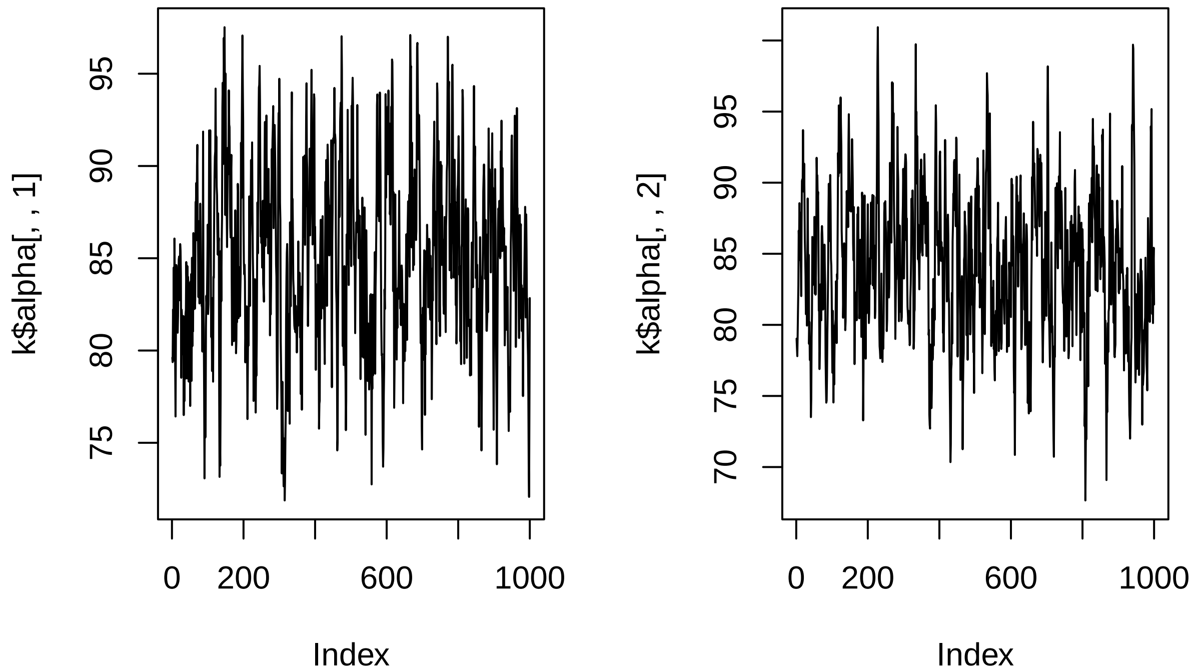



```r
# We can see that alpha and all three betas have clearly not converged.
par(mfrow=c(2,5))
acf(l$alpha[,,1],main="")
acf(l$alpha[,,2],main="")
acf(l$beta[1,,1],main="")
```

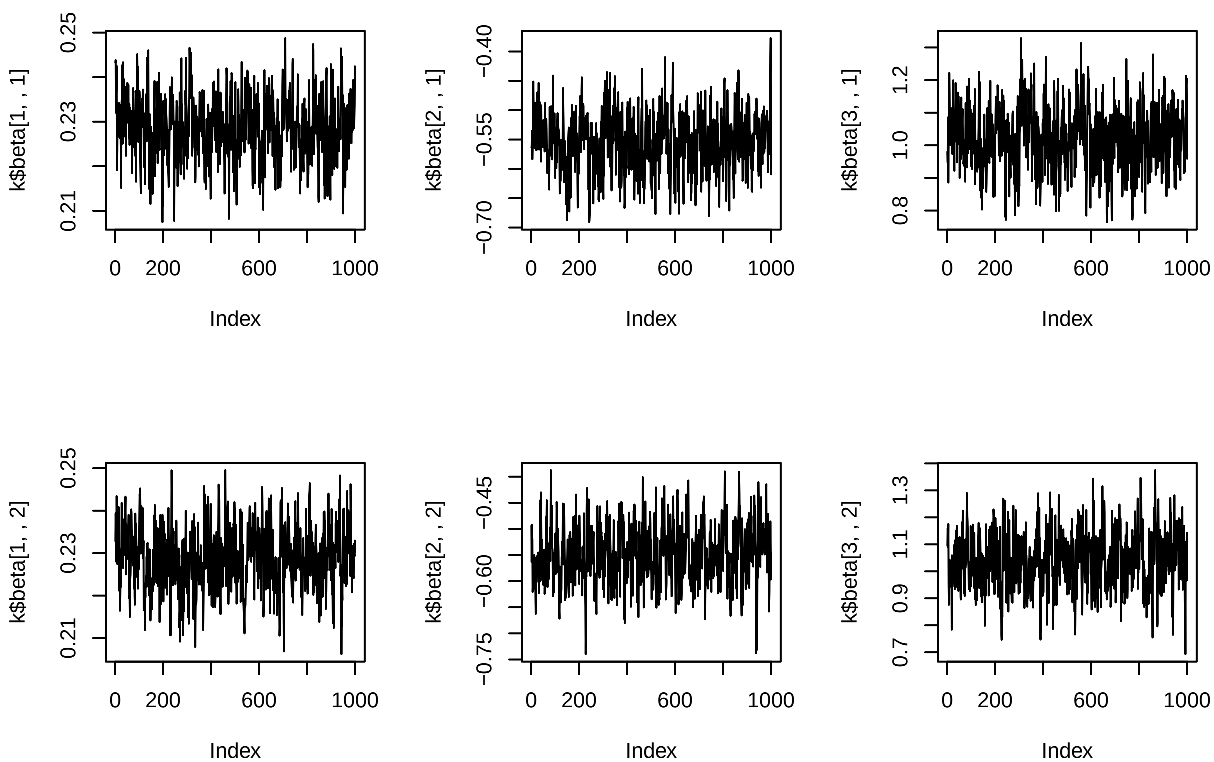
```
acf(l$beta[1,,2],main="")
acf(l$beta[2,,1],main="")
acf(l$beta[2,,2],main="")
acf(l$beta[3,,1],main="")
acf(l$beta[3,,2],main="")
acf(l$sigma[,,1],main="")
acf(l$sigma[,,2],main="")
```



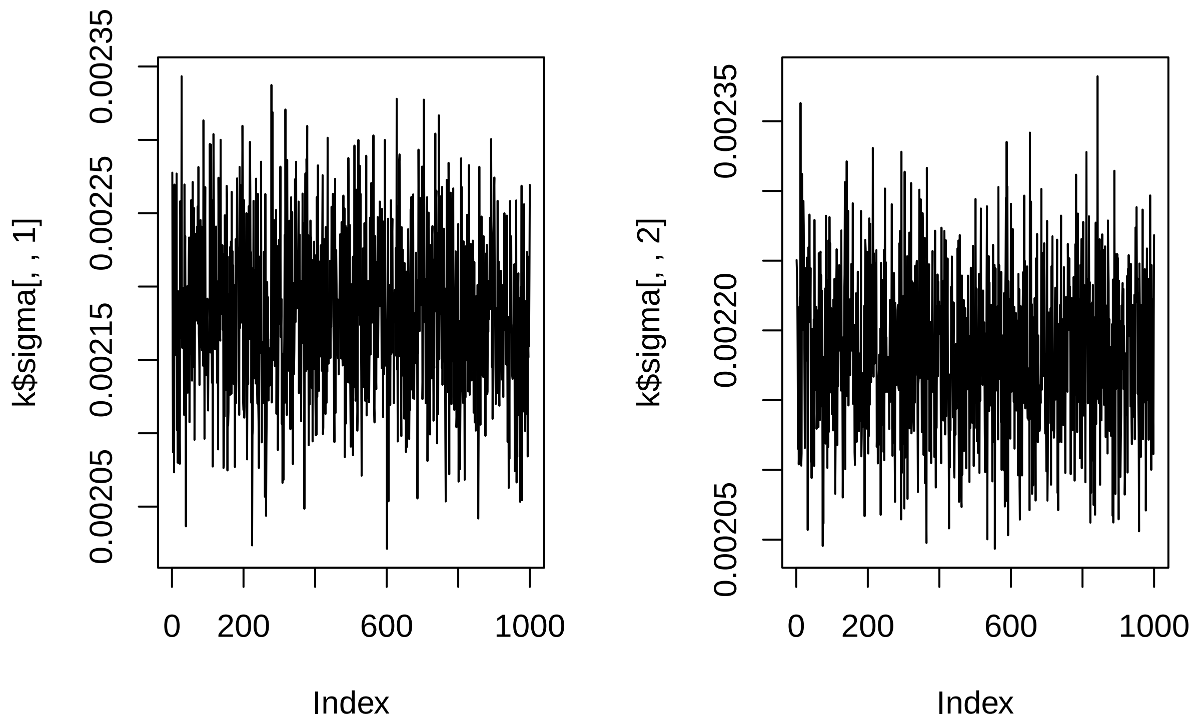

```
# There are clearly problems with autocorrelation among alpha and all three betas.
k<-jags.samples(gs,c('alpha','beta','sigma','y'),36*1000,thin=36) # Maximum thinning interval
par(mfrow=c(1,2))
plot(k$alpha[,,1],type="l")
plot(k$alpha[,,2],type="l")
```



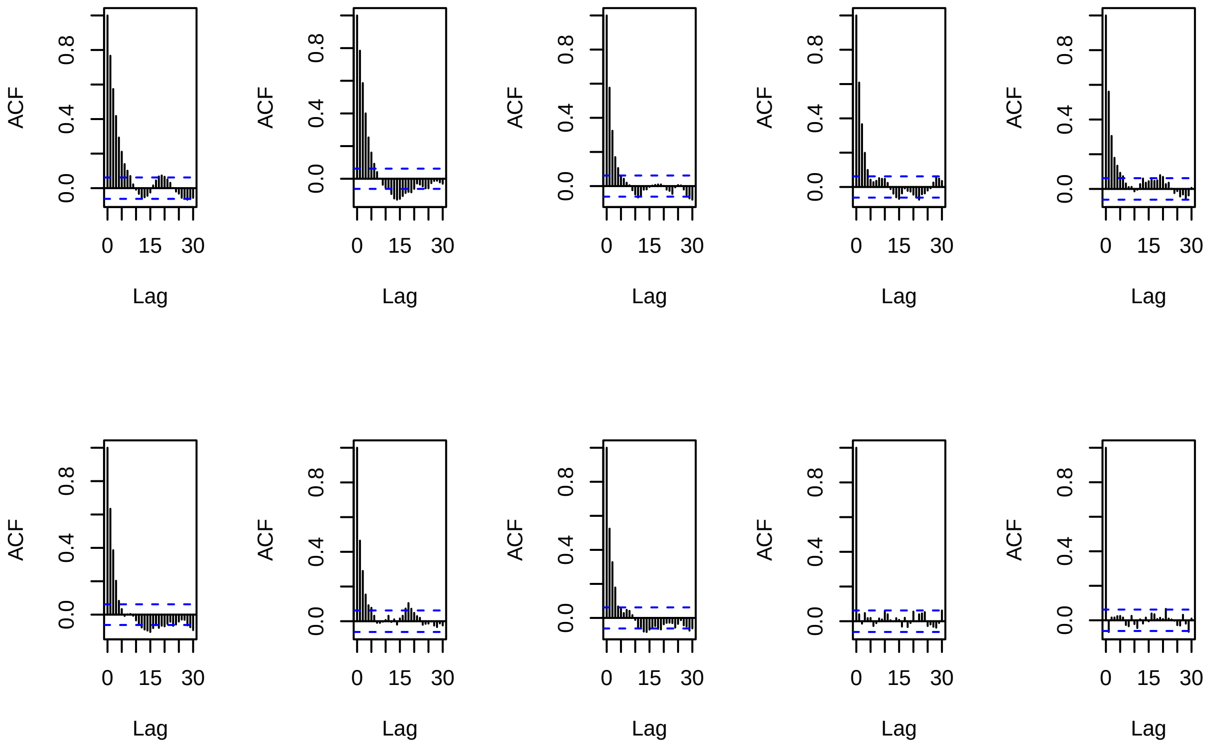

```
par(mfrow=c(2,3))
plot(k$beta[1,,1],type="l")
plot(k$beta[2,,1],type="l")
plot(k$beta[3,,1],type="l")
plot(k$beta[1,,2],type="l")
plot(k$beta[2,,2],type="l")
plot(k$beta[3,,2],type="l")
```



```r
par(mfrow=c(1,2))
plot(k$sigma[,,1],type="l")
plot(k$sigma[,,2],type="l")
```



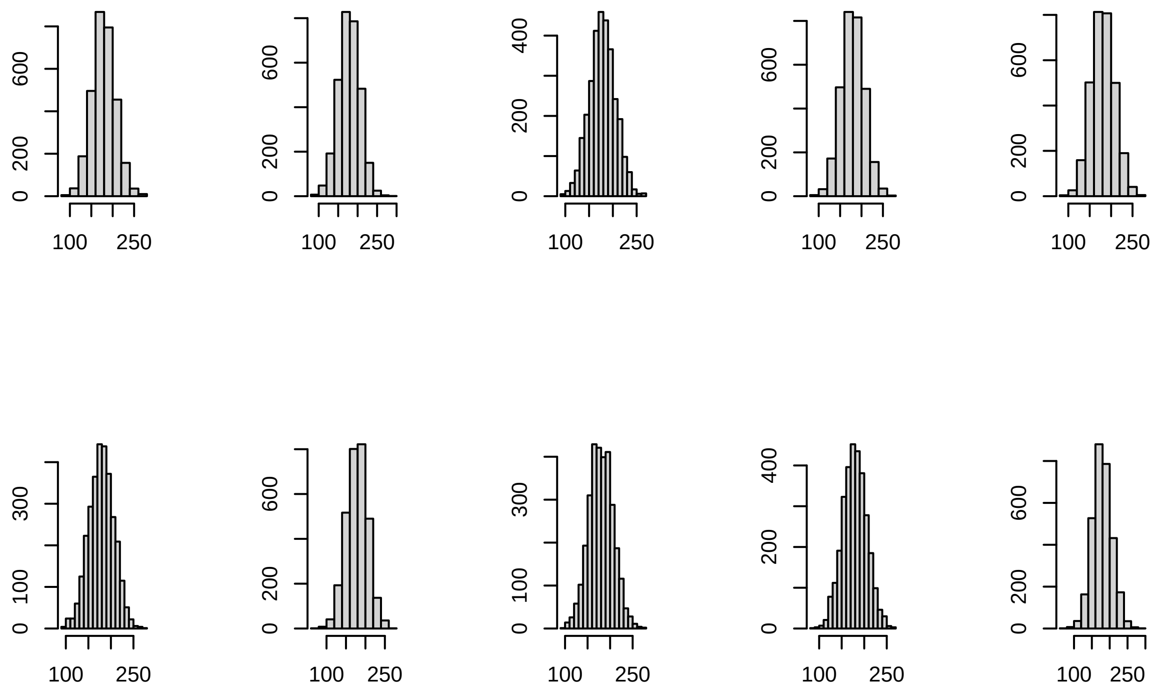

```r
# After thinning, all chains appear to have converged sufficiently.
par(mfrow=c(2,5))
acf(k$alpha[,,1],main="")
acf(k$alpha[,,2],main="")
acf(k$beta[1,,1],main="")
acf(k$beta[1,,2],main="")
acf(k$beta[2,,1],main="")
acf(k$beta[2,,2],main="")
acf(k$beta[3,,1],main="")
acf(k$beta[3,,2],main="")
acf(k$sigma[,,1],main="")
acf(k$sigma[,,2],main="")
```



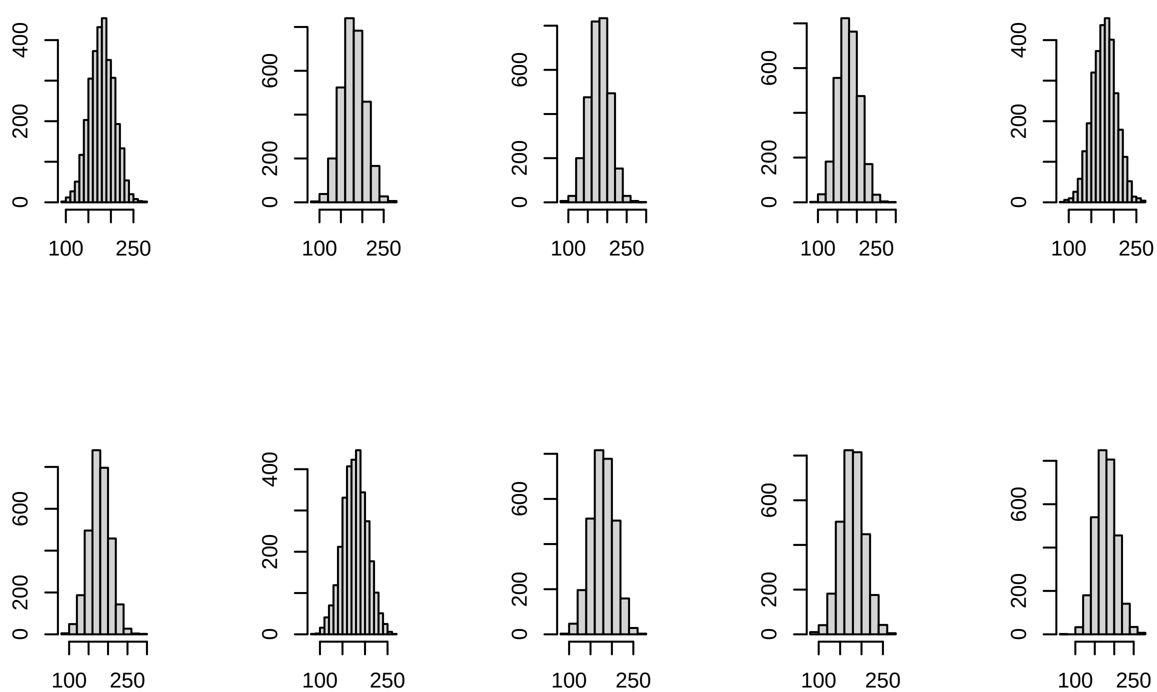
There is still autocorrelation among α and all three β's after thinning, but it has been reduced by a significant amount. It should be okay to proceed.

Problem 2(e)

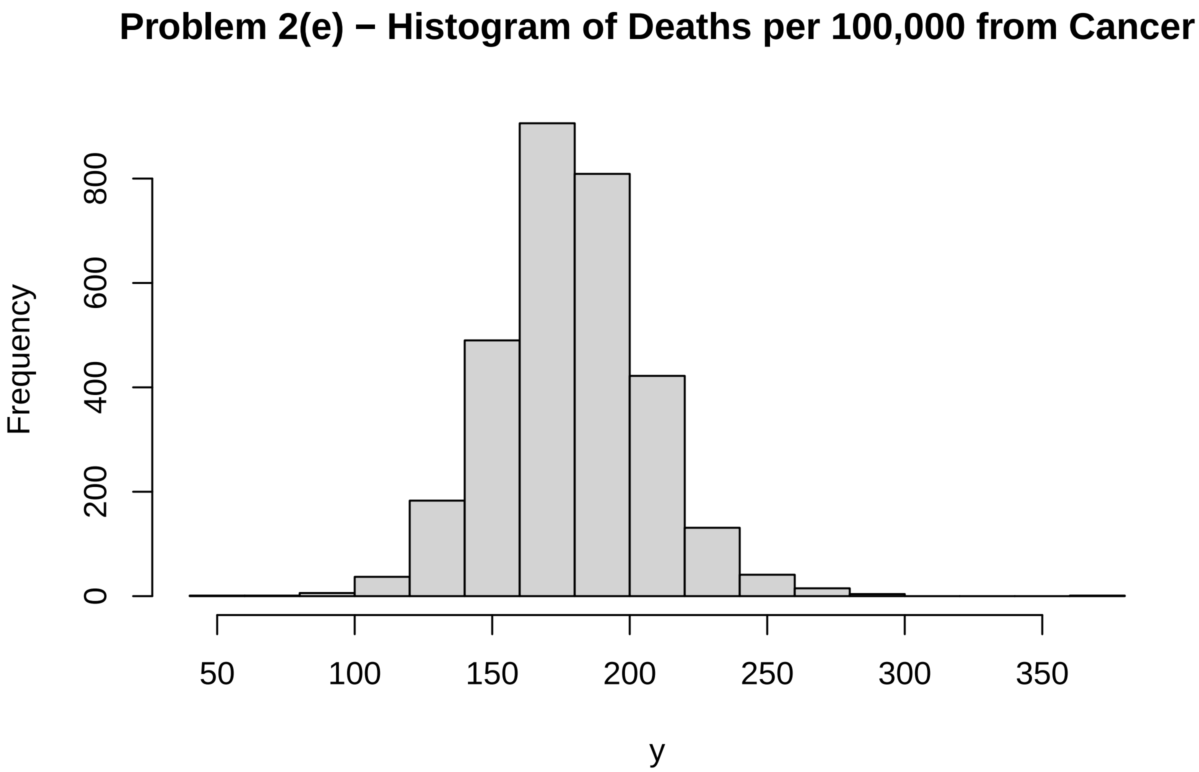
```
set.seed(2272)
s<-(nrow(c)-1)*var(y)/rchisq(1,nrow(c)-1)
r<-replicate(20,rnorm(nrow(c),rnorm(1,mean(y),sqrt(s/nrow(c))),sqrt(s)))
par(mfrow=c(2,5))
hist(r[,1],main="",xlab="",ylab="")
hist(r[,2],main="",xlab="",ylab="")
hist(r[,3],main="",xlab="",ylab="")
hist(r[,4],main="",xlab="",ylab="")
hist(r[,5],main="",xlab="",ylab="")
hist(r[,6],main="",xlab="",ylab="")
hist(r[,7],main="",xlab="",ylab="")
hist(r[,8],main="",xlab="",ylab="")
hist(r[,9],main="",xlab="",ylab="")
hist(r[,10],main="",xlab="",ylab="")
```



```
hist(r[,11],main="",xlab="",ylab="")
hist(r[,12],main="",xlab="",ylab="")
hist(r[,13],main="",xlab="",ylab="")
hist(r[,14],main="",xlab="",ylab="")
hist(r[,15],main="",xlab="",ylab="")
hist(r[,16],main="",xlab="",ylab="")
hist(r[,17],main="",xlab="",ylab="")
hist(r[,18],main="",xlab="",ylab="")
hist(r[,19],main="",xlab="",ylab="")
hist(r[,20],main="",xlab="",ylab="")
```



```r
par(mfrow=c(1,1))
hist(y,main="Problem 2(e) - Histogram of Deaths per 100,000 from Cancer")
```



Yes, the model appears to fit adequately. All 20 histograms look quite similar to the histogram of the data in terms of shape, spread, and location.

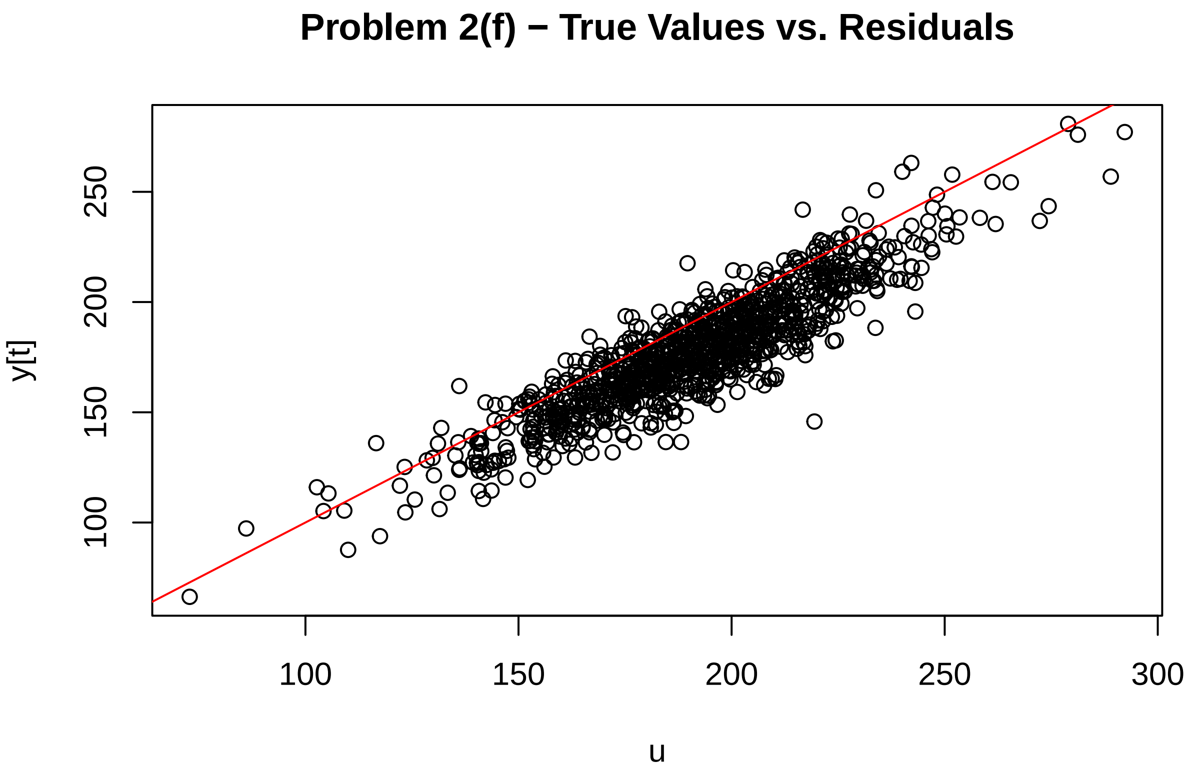
Problem 2(f)

```r
d<-"model {for (i in 1:n){yt[i] ~ dnorm(mu[i],sigma)
    mu[i]<-alpha+X[i,]%*%beta}
    alpha ~ dnorm(0,0.001)
    for (j in 1:p){beta[j] ~ dnorm(0,0.001)}
    sigma ~ dchisq(1)}"
write(d,"/Users/newuser/Desktop/2f.bug")
set.seed(2272)
gsd<-jags.model("/Users/newuser/Desktop/2f.bug",list('y'=y,'X'=X,'n'=n,'p'=p),n.chains=2,n.adapt=1000,i
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 0
##    Unobserved stochastic nodes: 3052
##    Total graph size: 21340
##
## Initializing model
```

```r
update(gsd,1000)
o<-jags.samples(gsd,c('alpha','beta','sigma','yt'),1000)
t<-sample(1:nrow(c),1000)
```

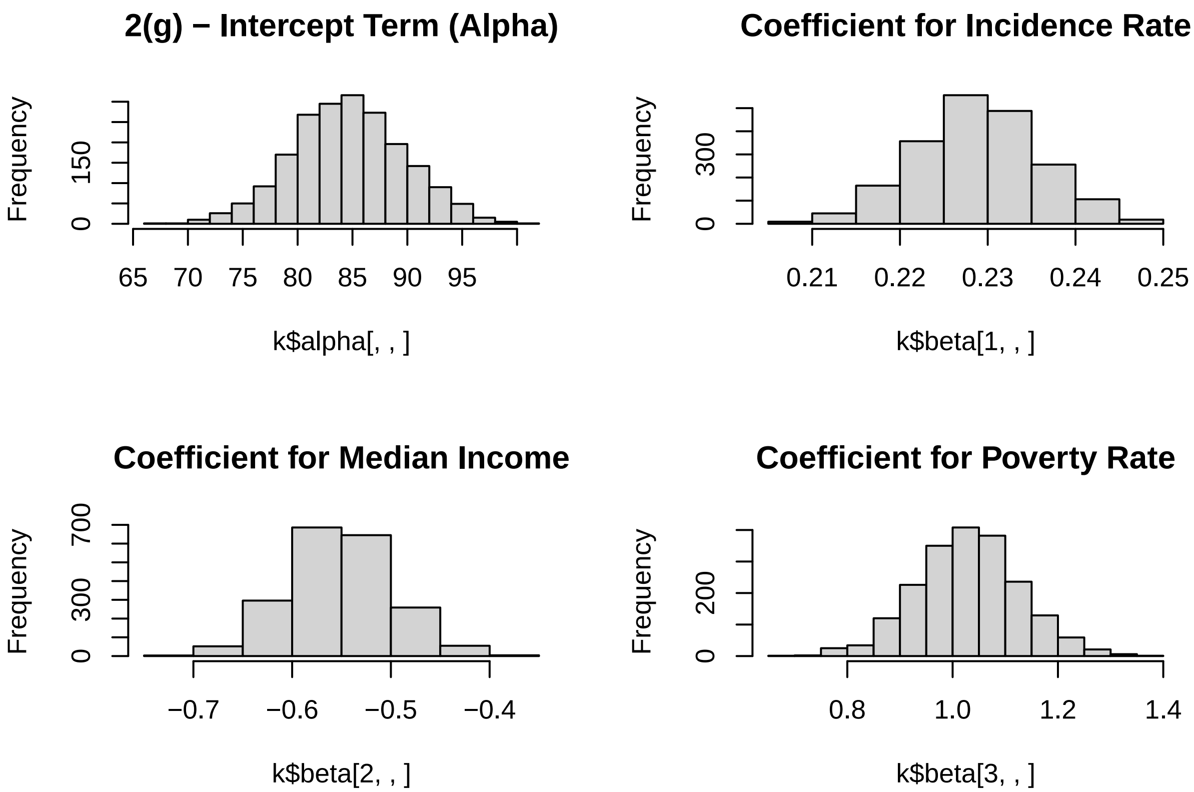
```
u<-y[t]-apply(o$yt[t,,1],1,mean)
plot(u,y[t],main="Problem 2(f) - True Values vs. Residuals")
abline(0,1,col="red")
```



It is difficult to comment on the plot specifically, as the `set.seed()` function does not work for JAGS models and after some online research it appears to be difficult to seed the models for reproducibility (thus causing the plot to change each time the document is knitted, as a new model is created each time). However, in most cases, it appears the points form a positive ellipse in the plot. The identity function $y = x$ has been added to the plot for additional interpretation.

Problem 2(g)

```
par(mfrow=c(2,2))
hist(k$alpha[,,],main="2(g) - Intercept Term (Alpha)")
hist(k$beta[1,,],main="Coefficient for Incidence Rate")
hist(k$beta[2,,],main="Coefficient for Median Income")
hist(k$beta[3,,],main="Coefficient for Poverty Rate")
```



Problem 2(h)

```
quantile(k$alpha[,,1],c(0.05/2,1-0.05/2)) # Using first chain from model
```

```
##     2.5%    97.5%
## 75.35034 94.48369
```

```
quantile(k$beta[1,,1],c(0.05/2,1-0.05/2))
```

```
##      2.5%     97.5%
## 0.2149145 0.2424362
```

```
quantile(k$beta[2,,1],c(0.05/2,1-0.05/2))
```

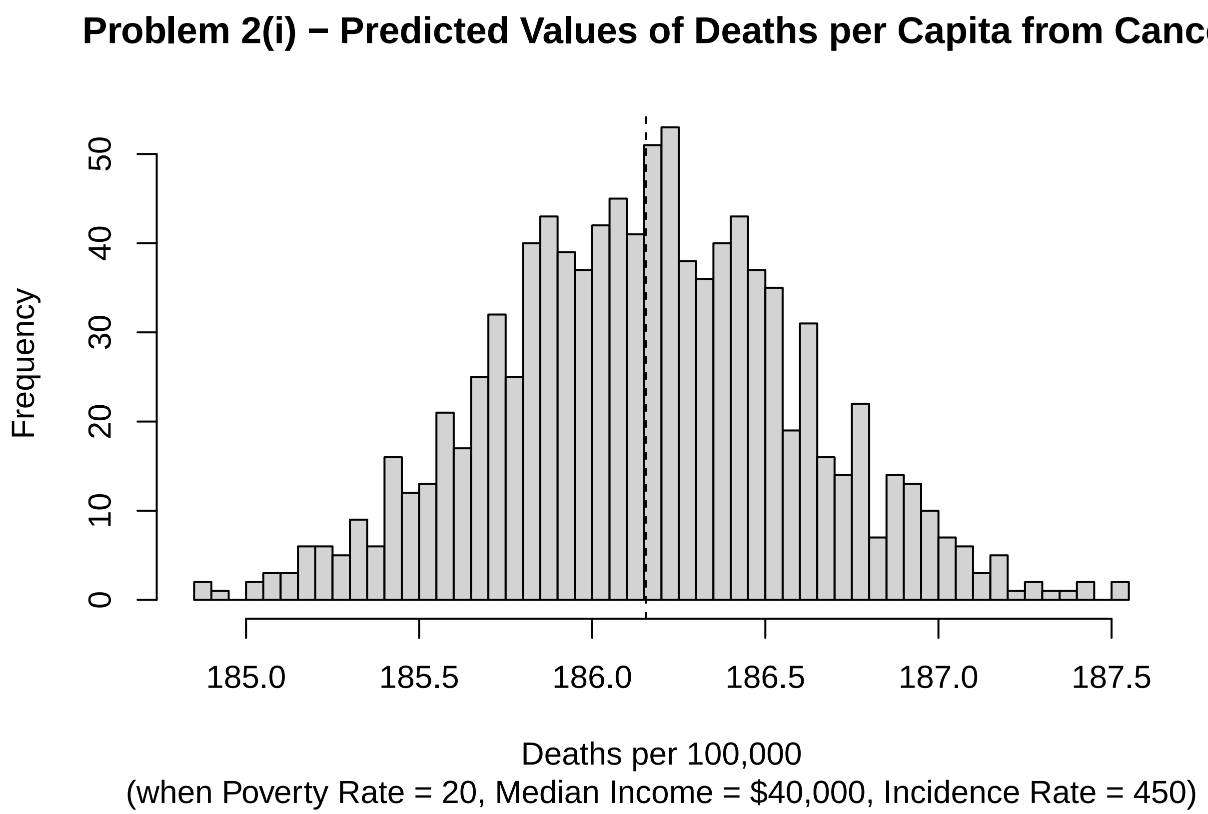
```
##       2.5%      97.5%
## -0.6547117 -0.4617297
```

```
quantile(k$beta[3,,1],c(0.05/2,1-0.05/2))
```

```
##      2.5%     97.5%
## 0.8411091 1.2033495
```

Problem 2(i)

```
hist(k$alpha[,,1]+c(20,40000/1000,450)%*%k$beta[3:1,,1],39,xlab="Deaths per 100,000",main="Problem 2(i)
abline(v=mean(k$alpha[,,1]+c(20,40000/1000,450)%*%k$beta[3:1,,1]),lty=2) # Posterior mean
```



Problem 2(j)

```
mean(k$beta[1,,]>1) # Coefficient for incidence rate is beta_1
```

```
## [1] 0
```

We can see the probability that $\beta_1 > 1$ is less than $\frac{1}{2000} = 0.0005$.