STAT 651 Mini Project 3

Stephen Merrill

February 11, 2017

1 Bootstrap Importance Sampling

1.1 Likelihood

I used a four parameter Beta function in order to align the support of the likelihood with the support of the data. I believe this distribution can appropriately capture probable data values, and can do so over the correct range of possible values (1-7). The pdf of this distribution is given as follows:

$$f(y;\alpha,\beta,a,c) = \frac{f(x;\alpha,\beta)}{c-a} = \frac{\left(\frac{y-a}{c-a}\right)^{\alpha-1}\left(\frac{c-y}{c-a}\right)^{\beta-1}}{(c-a)B(\alpha,\beta)} = \frac{(y-a)^{\alpha-1}(c-y)^{\beta-1}}{(c-a)^{\alpha+\beta-1}B(\alpha,\beta)}$$

Where c is fixed at 7 and a is fixed at 1. This is the pdf, and the likelihood would of course be the product of the pdf over the data values. A possible visualization of the likelihood can be seen in Figure 1.

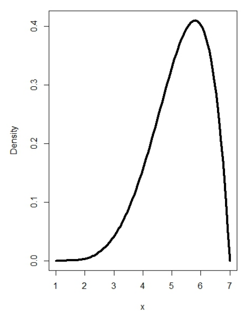

Figure 1: This four-parameter Beta Distribution, for example, represents my belief of how the rating data could be distributed.

1.2 Importance Function

I used a multivariate normal distribution as the importance function. This function works well as $I(\theta)$ because it mimics the values of the likelihood, as seen in Figure 2, and it covers the tails well. It is also a pdf that is easy to sample from and preserves the parameter space, as long as it remains greater than zero. Specifically, this distribution is:

$$MVN\left(\begin{bmatrix}17\\5\end{bmatrix}, \begin{bmatrix}25 & 5\\5 & 2\end{bmatrix}\right)$$

1.3 Prior Distributions

Since the support of α and β is greater than zero, I used a gamma distribution for both parameters. The shape and scale values were selected with consideration to the equations for the mean and variance of the beta likelihood. I believed the mean of teacher ratings at BYU would be above average, with a somewhat substantial amount of variance, and my choices for prior parameters values reflected that belief.

$$\alpha \sim Gamma(shape = 0.5, scale = 16)$$

$$\beta \sim Gamma(shape = 0.5, scale = 4)$$

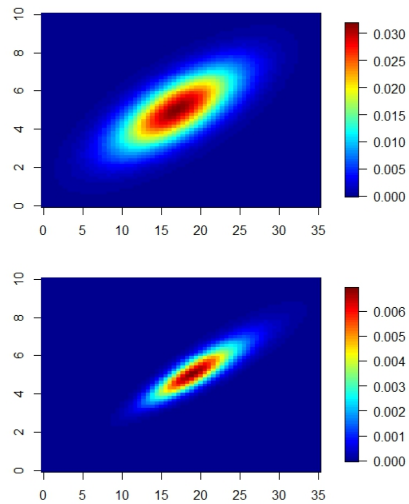

Figure 2: The Importance function above covers all values of $g(\theta)$

1.4 Posterior Distribution

The algorithm produces the bivariate posterior distribution in Figure 3.

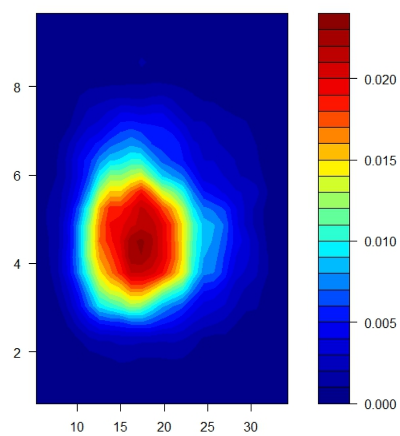

Figure 3: Bivariate posterior

1.5 Summary Statistics

- Mean, $E(\theta|Y) = \begin{bmatrix} 18.0421 \\ 4.7701 \end{bmatrix}$
- Variance, $V(\theta|Y) = \begin{bmatrix} 26.3348 & 0.0493 \\ 0.0493 & 1.6479 \end{bmatrix}$
- Standard Deviation, $\sqrt{V(\theta|Y)} = \begin{bmatrix} 5.1317 \\ 1.2837 \end{bmatrix}$

1.6 Posterior Predictive Distribution

Figure 4 shows the distribution of the "next" score.

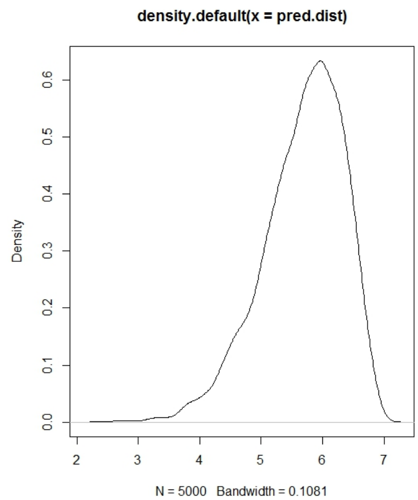

Figure 4: Posterior Predictive Distribution

- Probability next score is greater than 5, $P(\tilde{Y} > 5) = .8544$

2 Metropolis-Hastings Algorithm

Figure 5 shows 10000 draws generated through the M-H algorithm from the distribution of X where:

$$f^*(x) \propto (1 + (x - 10)^2/3)^{-2}$$

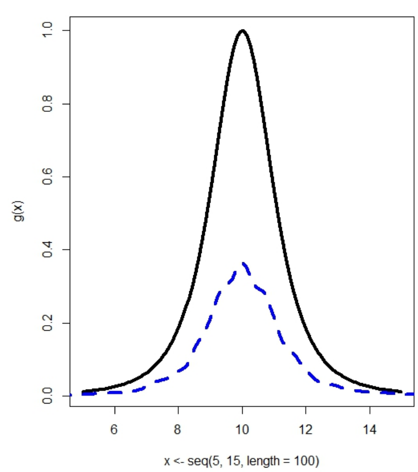

Figure 5: The dashed line density of the draws model the unnormalized distribution of f^*, displayed as a solid line

3 Gibbs Sampler

3.1 Likelihood

I used a Weibull likelihood, defined as follows:

$$f(x) = (\gamma/\beta)x^{\gamma-1}exp(-x^{\gamma})\beta$$

$$x \in (0, \infty), \gamma \in (0, \infty), \beta \in (0, \infty)$$

Where γ is the shape parameter, and β is the scale parameter. Figure 6 shows a possible visualization from this likelihood.

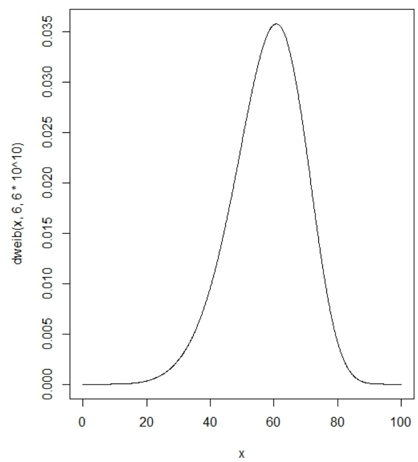

Figure 6: Here, $\gamma = 6$ and $\beta = 6e + 10$

3.2 Prior Distributions

$$\beta \sim InverseGamma(a, b)$$

Which is conjugate and allows for the posterior to be expressed in a closed form as:

$$\beta|x \sim InverseGamma(n + a, \sum x_i^{\gamma} + b)$$

$$\gamma \sim InverseGamma(c, d)$$

Figure 7 shows a plot of the prior distributions, Figure 8 shows a prior predictive plot.

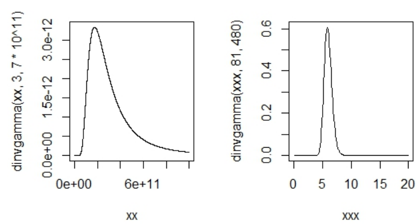

Figure 7: γ is on the right and β the left

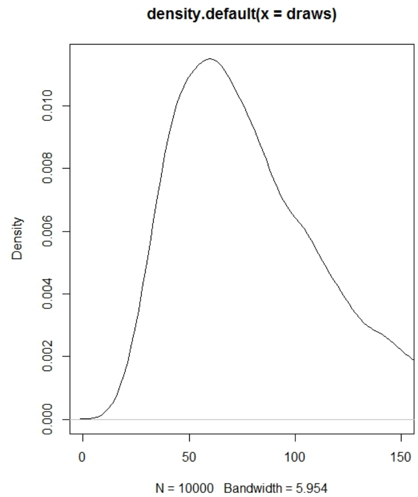

Figure 8: The prior distributions accurately represent the data.

3.3 Posterior Distribution

Figure 9 shows the marginal prior and posterior distributions. Figure 10 shows the bivariate contour plots.

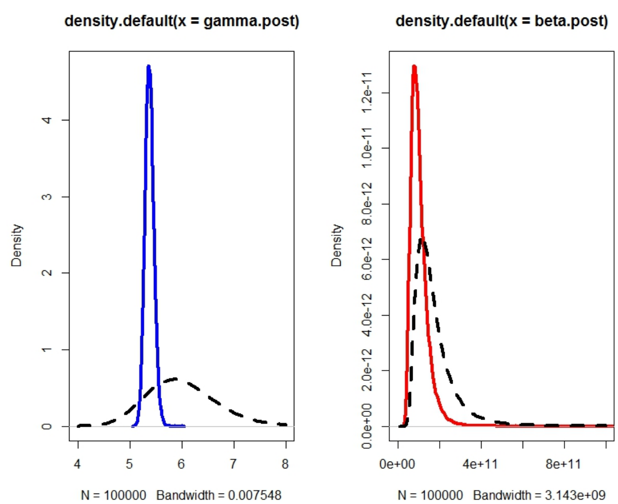

Figure 9: The dashed lines are the prior, and the solid lines are the posterior.

The 97% HPD Interval values are:
$\gamma = (5.203, 5.574)$
$\beta = (39637287078, 207119767272)$
Figure 11 shows the posterior distributions with the interval lines.

3.4 Summary Statistics

- Mean, $E(\gamma|x) = 5.3854$
- Variance, $V(\gamma|x) = 0.0073$
- Mean, $E(\beta|x) = 103629763271$

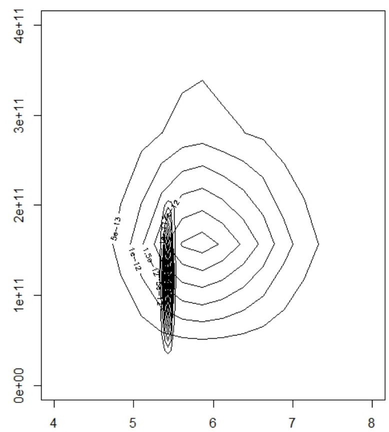

Figure 10: The prior distribution is the widely spaced contour and the posterior is the tightly clustered contour.

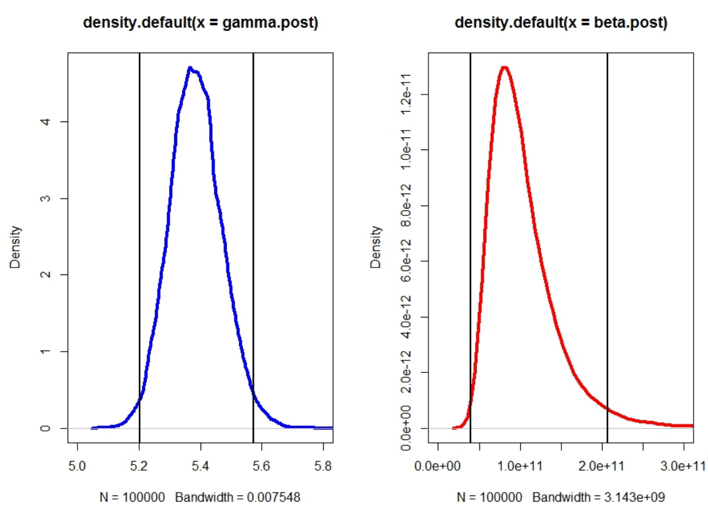

Figure 11: Posterior distributions with 97% HPD intervals

- Variance, $V(\beta|x) = 1.95482e + 21$

3.5 Posterior Predictive Distribution

Figure 12 shows the distribution of the "next" ball bearing.

- Probability next ball bearing will last 120 (10^6 revolutions) $P(\tilde{Y} > 120) = .19881$

3.6 MCMC Settings

- Burn = 100
- $\sigma_{cand} = 0.25$
- Length = 100000

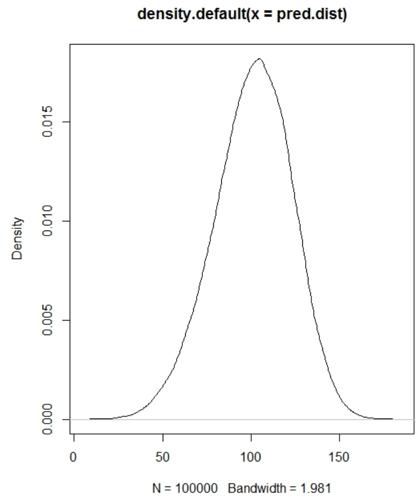

Figure 12: Posterior Predictive Distribution

- Candidate density for γ: Normal (centered at previous draw,σ_{cand}
- Figure 13 shows trace plots
- Acceptance rate for γ draws = 0.21963

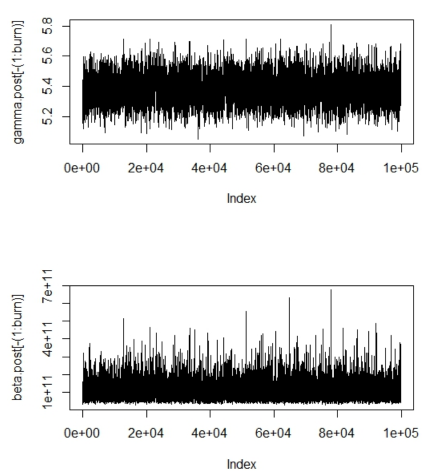

Figure 13: Trace plots suggest satisfactory convergence of the parameters.

4 Appendix: Code

```
score<-read.table("http://madison.byu.edu/bayes/faculty.dat")
library(ExtDist)
library(mvtnorm)
```

```
g <-function(a,b) {
  J <- numeric(M)
  for(i in 1:M) {
    J[i] <- prod(dBeta_ab(score[,1],a[i],b[i],1,7))*dgamma(a[i],.5,scale=16)*dgamma(b[i],.5
  }
  J
}

I <-function(a,b) {
  J <- numeric(M)
  for(i in 1:M) {
    J[i] <- dmvnorm(x=c(a[i],b[i]),mean=c(17,5),sigma=matrix(c(25,5,5,2),nrow=2))
  }
  J
}

M<-5000
vals <- rmvnorm(M,mean=c(17,5),sigma=matrix(c(25,5,5,2),nrow=2))

w <- g(vals[,1],vals[,2])/I(vals[,1],vals[,2])
q <- w/sum(w)

out.a <- sample(vals[,1],size=M,replace=T,prob=q)
out.b <- sample(vals[,2],size=M,replace=T,prob=q)

mean(out.a)
var(cbind(out.a,out.b))
sqrt(var(out.a))

mean(out.b)
var(out.b)
sqrt(var(out.b))

hist(out.a,nclass=50,prob=T,xlab="")
lines(density(out.a),lwd=4,col="blue")
hist(out.b,nclass=50,prob=T,xlab="")
lines(density(out.b),lwd=4,col="blue")

library(MASS)
filled.contour(kde2d(out.a,out.b),color.palette=colorRampPalette(c('white','blue','yellow'
filled.contour(kde2d(out.a,out.b),color.palette=colorRampPalette(c('darkblue','blue','cyan

#predictive distribution
pred.dist <- numeric(M)
for(i in 1:M) {
  index <- sample(1:M,1)
  a.samp <- out.a[index]
  b.samp <- out.b[index]
  pred.dist[i] <- rBeta_ab(1,a.samp,b.samp,1,7)
}

plot(density(pred.dist))

#prob > 5
```

```
length(which(pred.dist>=5))/M

#2

g <- function(x) {
  (1+((x-10)^2)/3)^-2
}
M<-10000
draws<-numeric(M)
draws[1]<-10
csig<-5
acc<-0
for(i in 2:M) {
  u<-runif(1)
  candy<-rnorm(1,draws[i-1],csig)
  alpha<-g(candy)/g(draws[i-1])
  draws[i]<-draws[i-1]
  if(u<alpha) {
    draws[i]<-candy
    acc<-acc+1
  }
}
acc<-acc/M
plot(draws,type="l")
plot(x<-seq(5,15,length=100),g(x),type="l",lwd=4)
lines(density(draws),lwd=4,lty=2,col="blue")

#3
ball <- read.table("http://madison.byu.edu/bayes/ballbearing2.dat")
dweib <- function(x,gamma,beta) {
  (gamma/beta)*x^(gamma-1)*exp((-x^gamma)/beta)
}
x <- seq(0,100,by=.1)
plot(x,dweib(x,6,6*10^10),type='l')

#prior for beta inv-gamma(a,b)
#conjugate is IV(n+a,sum(x^gamma)+b)

library(invgamma)

par(mfrow=c(1,2))
xx <- seq(10^8,10^12,by=10^8)
plot(xx,dinvgamma(xx,3,7*10^11),type='l')

#prior for gamma inv-gamma(a,b)
xxx <- seq(0,20,by=.1)
plot(xxx,dinvgamma(xxx,81,480),type='l')

par(mfrow=c(1,1))
#prior-predictive
draws.beta <- rinvgamma(M,3,7*10^11)
draws.gamma <- rinvgamma(M,81,480)
draws <- rweibull(M,scale=(draws.beta^(1/draws.gamma)),shape=draws.gamma)
plot(density(draws),xlim=c(0,150))
```

```
#gibbs sampler

#prior values
a.prior.beta <- 5
b.prior.beta <- 7*10^11
a.prior.gamma <- 81
b.prior.gamma <- 480

# starting values
M<-100000
beta.post<-gamma.post<-numeric(M)
gamma.post[1] <- 6
beta.post[1] <- 6*10^10

burn <- 100
csig <- 0.25
acc <- 0
n <- length(ball[,1])

for(i in 2:M) {
  cand <- rnorm(1,gamma.post[i-1],csig)
  gamma.post[i] <- gamma.post[i-1]
  if(cand > 0)
  {
    lpo <- sum(dweibull(ball[,1],gamma.post[i-1],beta.post[i-1]^(1/gamma.post[i-1]),log=T))
    lpn <- sum(dweibull(ball[,1],cand,beta.post[i-1]^(1/cand),log=T))+dinvgamma(cand,shape=

    if((lpn-lpo)>log(runif(1))){
      gamma.post[i] <- cand
      acc <- acc+1
    }
  }
  beta.post[i] <- rinvgamma(1,(n+a.prior.beta),(sum(ball^gamma.post[i]))+b.prior.beta)
}

accept <- acc/M

par(mfrow=c(2,1))
plot(gamma.post[-(1:burn)],type="l")
plot(beta.post[-(1:burn)],type="l")

par(mfrow=c(1,2))
plot(density(gamma.post),lwd=4,col="blue",xlim = c(4,8))
curve(dinvgamma(x,shape = a.prior.gamma, rate = b.prior.gamma),xlim = c(4,8),add = TRUE,lwd=

plot(density(beta.post),lwd=4,col="red",xlim = c(0,10^12))
curve(dinvgamma(x,shape = a.prior.beta,rate = b.prior.beta),xlim = c(0,10^12),add = TRUE,lw

mean(gamma.post)
mean(beta.post)
var(gamma.post)
var(beta.post)
```

```
library(MASS)
conts.post <- kde2d(gamma.post, beta.post)
filled.contour(conts.post, color.palette=colorRampPalette(c('white','blue','yellow','red','c

xx <- seq(4,8,length.out=M)
gamma.pre <- rinvgamma(xx,shape = a.prior.gamma, rate = b.prior.gamma)
xxx <- seq(0,10^11,length.out=M)
beta.pre <- rinvgamma(xxx,shape = a.prior.beta,rate = b.prior.beta)
conts.pre <- kde2d(gamma.pre,beta.pre)
filled.contour(conts.pre,color.palette=colorRampPalette(c('white','blue','yellow','red','da

contour(conts.pre$x,conts.pre$y,conts.pre$z,xlim=c(4,8),ylim=c(0,4e+11))
contour(conts.post$x,conts.post$y,conts.post$z,add=TRUE)

#predictive distribution
pred.dist <- numeric(M)
for(i in 1:M) {
  index <- sample(1:M,1)
  beta.samp <- beta.post[index]
  gamma.samp <- gamma.post[index]
  pred.dist[i] <- rweibull(1,scale=(beta.samp^(1/gamma.samp)),shape=gamma.samp)
}
par(mfrow=c(1,1))
plot(density(pred.dist))

length(which(pred.dist>=120))/M

#hpd
library(coda)
HPDinterval(as.mcmc(beta.post),prob=0.97)
HPDinterval(as.mcmc(gamma.post),prob=0.97)

par(mfrow=c(1,2))
plot(density(gamma.post),lwd=4,col="blue",xlim = c(5,5.8))
abline(v=5.203147,lwd=2)
abline(v=5.573943,lwd=2)
plot(density(beta.post),lwd=4,col="red",xlim = c(0,3e+11))
abline(v=39637287078,lwd=2)
abline(v=207119767272,lwd=2)
```