**R Code for Midterm**

Problem 1, Part 3 R Code:

>x=seq(0,5,length=10000)

>y=dgamma(x,1.4,2)

>plot(x,y, type='l', main='Prior Density - (Problem 1, part 3)')

> 0.886\*(1.4/2)^(-0.5)

[1] 1.058973

> qgamma(.025, 1.4, 2)

[1] 0.04340487

> qgamma(.975, 1.4, 2)

[1] 2.242941

> 0.886\*(qgamma(.025, 1.4, 2))^(-1/2)

[1] 4.252698

> 0.886\*(qgamma(.975, 1.4, 2))^(-1/2)

[1] 0.5915954

> qgamma(.005, 1.4, 2)

[1] 0.01341205

> qgamma(.995, 1.4, 2)

[1] 3.102909

> 0.886\*(qgamma(.005, 1.4, 2))^(-1/2)

[1] 7.650435

> 0.886\*(qgamma(.995, 1.4, 2))^(-1/2)

[1] 0.5029782

Problem 1, Part 4 R Code:

> x=seq(0,5,length=10000)

> y=dgamma(x,11.4,11.092)

> plot(x,y, type='l', main='Problem 1 part 4-Posterior density')

> #95% posterior intervals

> qgamma(.025, 11.4, 11.092)

[1] 0.5205005

> qgamma(.975, 11.4, 11.092)

[1] 1.704705

> x=rgamma(100000,11.4,11.092)

> hist(x)

> mean(x)

[1] 1.028696

> var(x)

[1] 0.09267666

> quantile(x,0.025)

2.5%

0.5222731

> quantile(x,0.975)

97.5%

1.705622

Problem 3 Mathematica Code:

In[1]:= Sum[(1/N)\*(0.99)^N, {N, 203, Infinity}]

Out[1]= 0.0465803

Problem 4 Part 4 R Code:

I=5

J=1

sigma.a=1

sigma.e=1

B=1000 #burnout period

N=10000

alpha=matrix(0,I,N+B)

alpha[,1]=numeric(I)

mu1=numeric(N+B)

mu1[1]=0

#Generating data from (1)

Y=matrix(0,I,J)

for (i in 1:I){

alpha.i=rnorm(n=1,0,sigma.a)

for (j in 1:J){

Y[i,j]=rnorm(n=1,alpha.i,sigma.e)

}

}

#Gibbs sampler for parameterization(i):

for (k in 2:(B+N)){

x=matrix(0,I,J)

for (i in 1:I){

for (j in 1:J){

x[i,j]=Y[i,j]-(alpha[i,k-1])

}

}

new=sum(x)/(I\*J)

mu1[k]=rnorm(n=1,new,sigma.e/sqrt(I\*J))

for (i in 1:I){

sigma.i=sqrt((J/sigma.e^2+1/sigma.a^2)^(-1))

mu1.i=sum(Y[i,]-mu1[k])/sigma.e^2\*sigma.i^2

alpha[i,k]=rnorm(1,mu1.i,sigma.i)

}

}

#Gibbs sampler or parameterization(ii) :

mu2=numeric(N+B)

mu2[1]=0

e=matrix(0,I,B+N)

e[,1]=numeric(I)

for (k in 2:(B+N)){

mu2[k]=rnorm(n=1,mean(e[,k-1]),sigma.a/sqrt(I))

for (i in 1:I){

sigma.i=sqrt((J/sigma.e^2+1/sigma.a^2)^(-1))

mu1.i=(sum(Y[i,])/sigma.e^2+mu2[k]/sigma.a^2)\*

sigma.i^2

e[i,k]=rnorm(1,mu1.i,sigma.i)

}

}

#Investigation of the sample ACF

acf(mu1[(B+1):(B+N)], main="Sample ACF for mu - parametrization (i) (Part 4)")

acf(alpha[1,(B+1):(B+N)], main="Sample ACF for alpha - parametrization (i), (Part 4)")

acf(mu2[(B+1):(B+N)], main="Sample ACF for mu - parametrization (ii), (Part 4)")

acf(e[1,(B+1):(B+N)], main="Sample ACF for eta - parametrization (ii), (Part 4)")

Problem 4 Part 5 R Code:

I=5

J=1

sigma.a=sqrt(10)

sigma.e=1

B=1000 #Burnout period

N=10000

alpha=matrix(0,I,N+B)

alpha[,1]=numeric(I)

mu1=numeric(N+B)

mu1[1]=0

#Generating data from (1)

Y=matrix(0,I,J)

for (i in 1:I){

alpha.i=rnorm(n=1,0,sigma.a)

for (j in 1:J){

Y[i,j]=rnorm(n=1,alpha.i,sigma.e)

}

}

#Gibbs sampler for parameterization(i) :

for (k in 2:(B+N)){

x=matrix(0,I,J)

for (i in 1:I){

for (j in 1:J){

x[i,j]=Y[i,j]-(alpha[i,k-1])

}

}

new=sum(x)/(I\*J)

mu1[k]=rnorm(n=1,new,sigma.e/sqrt(I\*J))

for (i in 1:I){

sigma.i=sqrt((J/sigma.e^2+1/sigma.a^2)^(-1))

mu1.i=sum(Y[i,]-mu1[k])/sigma.e^2\*sigma.i^2

alpha[i,k]=rnorm(1,mu1.i,sigma.i)

}

}

#Gibbs sampler for parameterization(ii) :

mu2=numeric(N+B)

mu2[1]=0

e=matrix(0,I,B+N)

e[,1]=numeric(I)

for (k in 2:(B+N)){

mu2[k]=rnorm(n=1,mean(e[,k-1]),sigma.a/sqrt(I))

for (i in 1:I){

sigma.i=sqrt((J/sigma.e^2+1/sigma.a^2)^(-1))

mu1.i=(sum(Y[i,])/sigma.e^2+mu2[k]/sigma.a^2)\*

sigma.i^2

e[i,k]=rnorm(1,mu1.i,sigma.i)

}

}

#Investigation of the sample ACF

acf(mu1[(B+1):(B+N)], main="Sample ACF for mu - parametrization (i), (Part 5)")

acf(alpha[1,(B+1):(B+N)], main="Sample ACF for alpha - parametrization (i), (Part 5)")

acf(mu2[(B+1):(B+N)], main="Sample ACF for mu - parametrization (ii), (Part 5)")

acf(e[1,(B+1):(B+N)], main="Sample ACF for eta - parametrization (ii), (Part 5)")