Gamma-Inverse Gamma Models

Saikat Kar

13/08/2021

library(rstan)  
library(ggmcmc)  
library(bayesplot)

# Gamma-Inverse Gamma models:

Here, in our model we take likelihood i.e where pdf of Gamma distribution is

and taking prior distribution of ~ InverseGamma(,). where pdf of InverseGamma(,) is

Now, our target is to draw random sample from the posterior distribution i.e. from distribution of

Now, we defining models of likelihood and prior as well as parameters as the following:

data{  
 int<lower=0> N; //Number of Trials  
 real<lower=0> x; //random sample values taken from Gamma distribution  
}  
  
parameters{  
 real<lower=0> theta; //scale parameter of Gamma distribution; theta(>0)   
}  
  
  
model{  
 target+= inv\_gamma\_lpdf(theta|3,4); //Prior distribution  
 target+= gamma\_lpdf(x|10,theta); //Likelihood  
}

Here we taking hyperparamters , as 3,4 respectively(as W.K.T in InverseGamma(,) , >0).

As ~InverseGamma; so theta belong to (0,). That’s why we define theta in *parameter block* and that belong to .

Taking n of Gamma(n,) as 10 and theta is unknown parameter. Now,constructing 4 chains of markov models to draw random samples from required posterior distribution.

model\_file=file.choose()  
N=100  
theta=0.5  
x=rgamma(N,shape=1,scale = theta)  
  
stan\_data = list(N=N,x=sum(x))  
stan\_model=stan(model\_file, #Stan Program   
 data=stan\_data, #named list of data  
 chains=4, #Number of Markov Chain  
 iter=2000, #Total number of iterations per chain  
 warmup= 1000, #Number of Warmup iterations per chain  
 refresh=0)

Note: Here,instead of running a markov chain of 8,000; we run 4 chains each for 2000 iterations and treat 1,000 as warmup value()

After finishing the computations,summaries of the inferences and convergences are shown below:

print(stan\_model,digits=2)

## Inference for Stan model: Gamma\_Inv\_Gamma.  
## 4 chains, each with iter=2000; warmup=1000; thin=1;   
## post-warmup draws per chain=1000, total post-warmup draws=4000.  
##   
## mean se\_mean sd 2.5% 25% 50% 75% 97.5% n\_eff Rhat  
## theta 0.34 0.00 0.06 0.24 0.30 0.34 0.38 0.48 1464 1  
## lp\_\_ -11.78 0.02 0.70 -13.82 -11.94 -11.51 -11.34 -11.28 1527 1  
##   
## Samples were drawn using NUTS(diag\_e) at Sat Aug 21 18:57:25 2021.  
## For each parameter, n\_eff is a crude measure of effective sample size,  
## and Rhat is the potential scale reduction factor on split chains (at   
## convergence, Rhat=1).

- potential scale reduction factor on split chains

n\_eff is a crude measure of effective sample size(# of samples required to reach convergence)

In the above table, mean is the estimated standard posterior mean,se\_mean is the estimated standard error of mean of simulations,sd is the standard deviation.

## Diagnostics of checking convergence of chain:

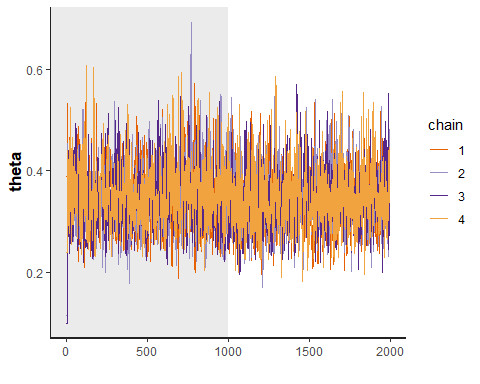
1. **Use multiple starting points**
2. **Trace Plots**
3. **Autocorrelation Plots/Autocorrelogram**

## 1. **Use multiple starting points(Gelman Rubin Test)**:

In this model, we see that i.e. estimated potential scale reduction factor is 1; means that the between chains the average variance isn’t so large; after certain points it converges.

## 2. **Trace Plots:**

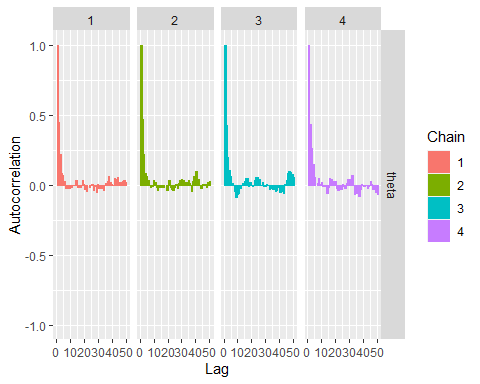
traceplot(stan\_model,inc\_warmup=T)



In the dark shaded area i.e. that is of first 1,000 iteration and after 1,000 we see that 4 chains are converging in such a manner that, we can’t distinguish 4 chains one from another.So,for surity in convergence; we have to look at only at the Traceplot of warmup values.

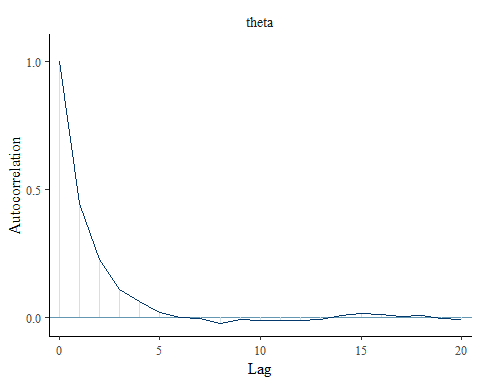
## 3.**Autocorrelation Plot/Autocorrelogram:**

obj =ggs(stan\_model)  
ggs\_autocorrelation(obj)



1st plot showing autocorrelation plot for each chain; after how much lag it is going to thin that means there is absence of dependency/non stationarity in the samples as per our required.We always want to draw random samples from required/target distribution; so if from autocorrelation plot we understand autocorrelation is not coming thin(i.e. significant) after certain points; then we have to take more iterations for drawing random samples.In this fig, we can say there is no dependency present after certain point for each chain.

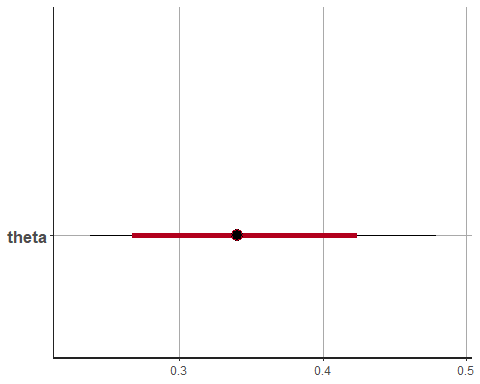
gamma\_invgamma\_fit\_score = as.data.frame(stan\_model)  
mcmc\_acf(gamma\_invgamma\_fit\_score,pars="theta")



W.K.T. for stationary series the ACF plot dies off very quickly.2nd plot showing after lag 5 there is no more significant values.So,the series of data/samples are stationary

**Credible Interval for theta:**

plot(stan\_model,ci\_level=0.8,outer\_level=0.95)



Here red highlighted line is 95% credible interval for ; whereas light black line is the 80% credible interval for .

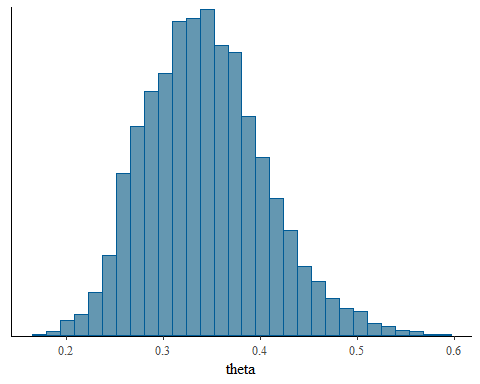
**95% confidence Interval:**

print(stan\_model,"theta",probs = c(0.05,0.95))

## Inference for Stan model: Gamma\_Inv\_Gamma.  
## 4 chains, each with iter=2000; warmup=1000; thin=1;   
## post-warmup draws per chain=1000, total post-warmup draws=4000.  
##   
## mean se\_mean sd 5% 95% n\_eff Rhat  
## theta 0.34 0 0.06 0.25 0.45 1464 1  
##   
## Samples were drawn using NUTS(diag\_e) at Sat Aug 21 18:57:25 2021.  
## For each parameter, n\_eff is a crude measure of effective sample size,  
## and Rhat is the potential scale reduction factor on split chains (at   
## convergence, Rhat=1).

So, here (0.25,0.45) is the 95% credible interval for . Also, we can see Posterior mean i.e.  = 0.34 which is also Bayes Estimator; if we consider loss function as squared error loss.

**Histogram of Posterior distribution:**

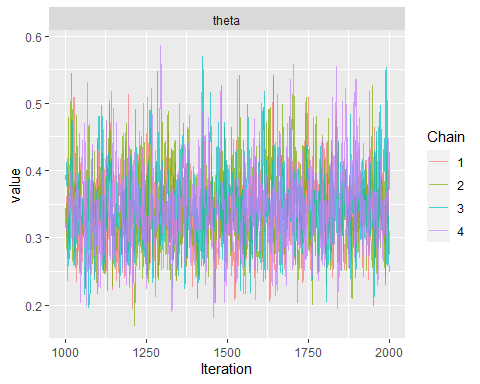


This is the histogram of Posterior distribution;. Basically we drawing likelihood from Gamma(n=10,) and prior from InverseGamma(=3,=4); so from calculation W.K.T posterior should come from InverseGamma(+np=1003,+=59.38003) as =55.38003.

So,Basically the above histogram is of distribution of ~InverseGamma(1003,59.38003)

**Traceplot Of Warmup values:**

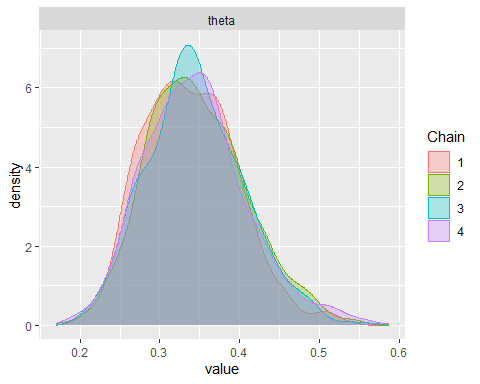
ggs\_traceplot(obj)



This is the traceplot of iteration of 4 chains after warmup i.e. after . From this, we get more clear idea about convergence of marcov chains.

**Density plots:**

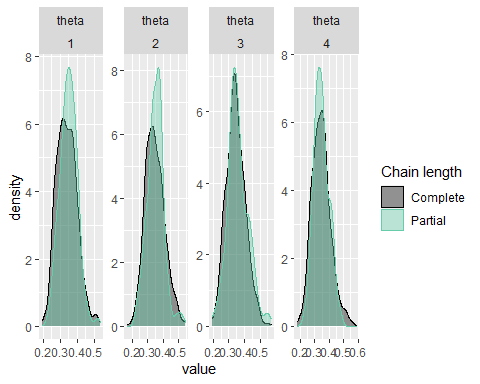
ggs\_density(obj)



This is the density plot of 4 chains of posterior distribution.These density plots are same;overlapping on each other.

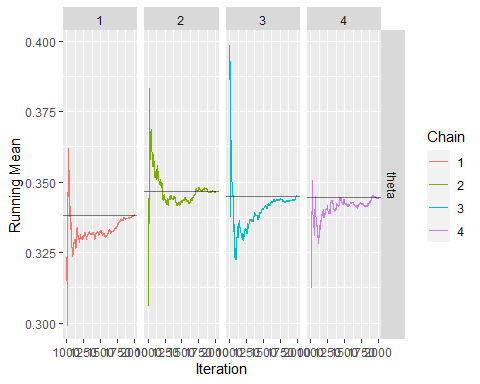
**Complete vs Partial plots for each chain:**

ggs\_compare\_partial(obj)



Complete means density plot of total 4000 post warmup samples and partial means 1000 samples for each chain and here is 4 graphs fro each chain.This also shows for each chain; how samples partially drawn for each chain.

ggs\_running(obj)



This shows how samples are going for each chain;i.e. whaether it is going to converge quickly or it is taking time for any chain or how much time it’s taking.