Applied Bayesian Statistics

Shinoj Philip John

22/09/2024

Introduction

To create a JAGS model diagram for the multiple linear regression setting and create required JAGS data and model taking into consideration of the prior information. Secondly use the MCMC diagnostic check for each parameter to check their appropriateness and create a posterior distribution of each parameter. Finally create a prediction model using the Bayesian point estimates of the model parameters and find prediction sale prices for the given data.

Analysis

Descriptive Analysis

- **Checking the summary of the data**

```
> summary(propPricesAus)
 SalePrice.100K.       Area          Bedrooms       Bathrooms       CarParks      PropertyType
 Min.   : 2.000   Min.   :  50.0   Min.   :1.000   Min.   :1.000   Min.   :0.000   Min.   :0.0000
 1st Qu.: 3.500   1st Qu.: 353.0   1st Qu.:2.000   1st Qu.:1.000   1st Qu.:1.000   1st Qu.:0.0000
 Median : 4.500   Median : 568.0   Median :3.000   Median :2.000   Median :2.000   Median :0.0000
 Mean   : 6.094   Mean   : 690.2   Mean   :2.889   Mean   :1.609   Mean   :1.672   Mean   :0.3162
 3rd Qu.: 6.550   3rd Qu.: 752.0   3rd Qu.:3.000   3rd Qu.:2.000   3rd Qu.:2.000   3rd Qu.:1.0000
 Max.   :70.000   Max.   :3500.0   Max.   :7.000   Max.   :4.000   Max.   :9.000   Max.   :1.0000
```

Figure 1:Summary of the data

The data is continuous. Here we see that the median is lesser than the mean which means that there is a *right skewness* in the data for the sale price, area and property type columns while for the remaining bedrooms, bathrooms and carparks columns the median is higher than the mean which means there is a *left skewness*. Here the sample mean of the dependant variable that is SalePrice(100K) is seen to be 6.094, while the minimum value is 2 and maximum value is 70 in the data set . All the values are in 100k values.

- **Scatter Plots**

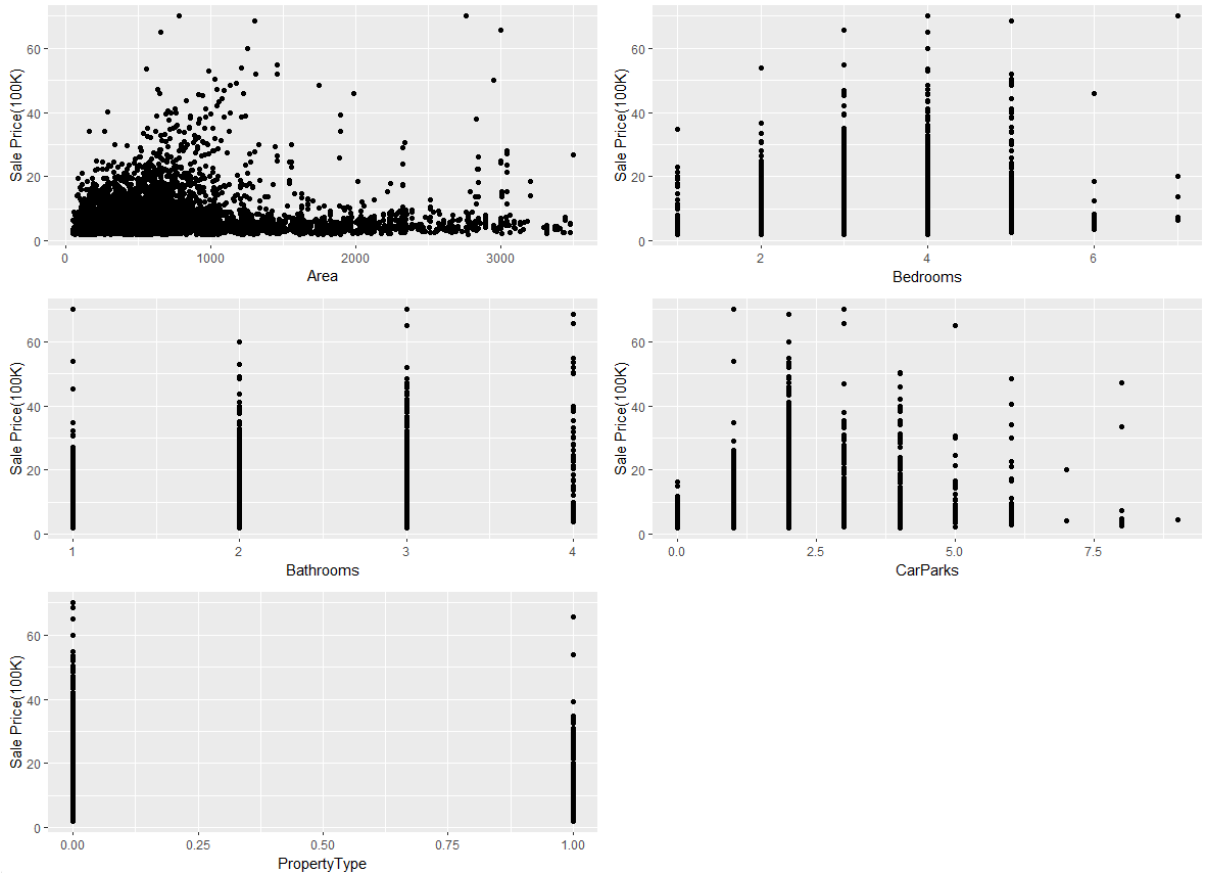

Figure 2:Scatter Plot

Discussion

The scatter plots shows correlations between the independent variables[Area, Bedrooms, Bathrooms, CarParks and PropertyType] with the dependent variable[SalePrice(100K)]. As for Area it can been that most of the data points are aligned to the bottom and maybe a positive correlation can be considered since the SalePrice increases with the Area. While for the Bedrooms the SalePrice increases with increase in bedrooms as more density in the data points can be seen but not entirely true as beyond 5 bedrooms the data is very less. Also for the SalePrice increases with the increase in Bathrooms as more density can be seem in the data points as the number of bathrooms increase. While for CarParks there is no clear increase that can be noticed and the SalePrice peaks at 2 CarParks and decreases as the value increases. Finally for the Property Type it is clearly seen that the SalePrice decreases with change in PropertyType from House to Unit.

- **Correlation Matrix**

```
CORRELATION MATRIX OF PREDICTORS:
 > show( round(cor(x),3) )
              Area Bedrooms Bathrooms CarParks PropertyType
Area         1.000   -0.270    -0.087   -0.096         0.32
Bedrooms    -0.270    1.000     0.538    0.435        -0.56
Bathrooms   -0.087    0.538     1.000    0.366        -0.29
CarParks    -0.096    0.435     0.366    1.000        -0.36
PropertyType 0.320   -0.560    -0.290   -0.360         1.00
```

Figure 3: Correlation Matrix

The correlation matrix shows that all the independent variables they are neither highly positively or negatively correlated and the highest positive correlation is seen between Bedrooms and Bathrooms of 0.538 which is logical while the highest negative correlation is seen between Bedrooms and PropertyType of -0.56.

- **Histogram of dependent variable**

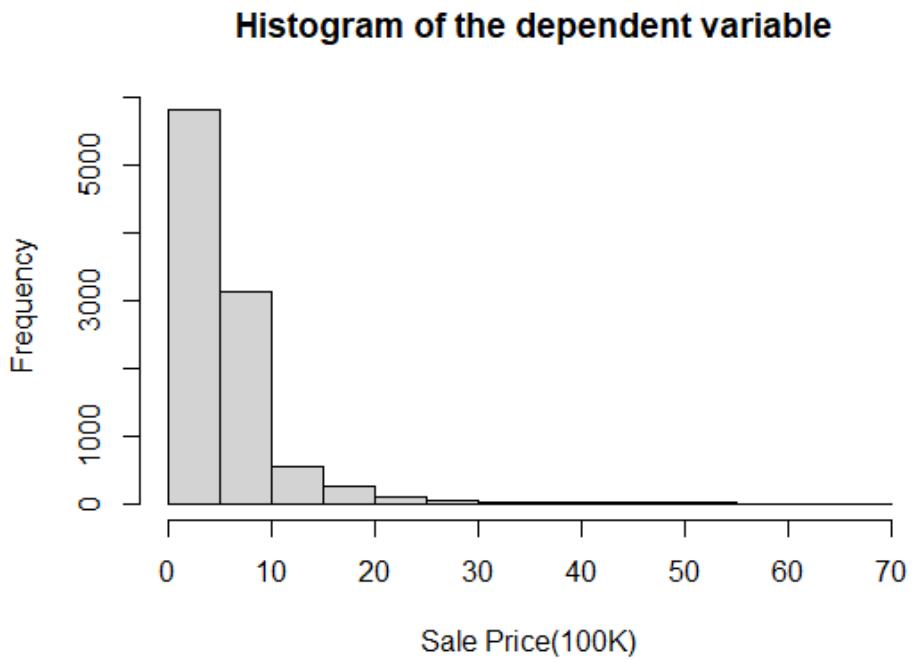

Figure 4: Histogram dependent variable

The sample data shows a *right skewedness* when a histogram is plotted. Thus the likelihood is considered to be gamma distributed while the prior information is taken as normal distribution. This is used for the modelling.

- **Density estimate of Sale Price:**

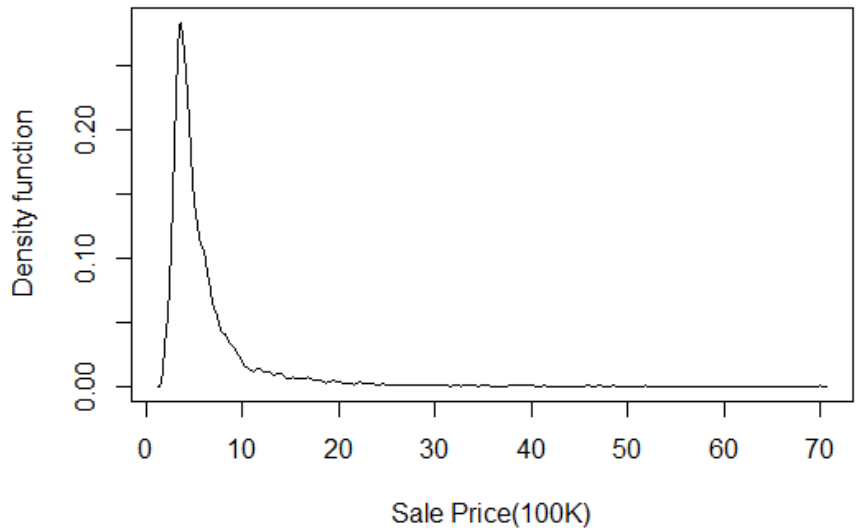

Figure 5: Density Estimate of Sale Price

The density plot shows the distribution is right skewed thus further confirming the analysis to use gamma likelihood.

- **Mathematical model:**

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_5 + \epsilon$$

$$\epsilon \sim Gamma\left(\frac{\mu^2}{\sigma^2}, \frac{\mu}{\sigma^2}\right)$$

- **JAGS Model Diagram**

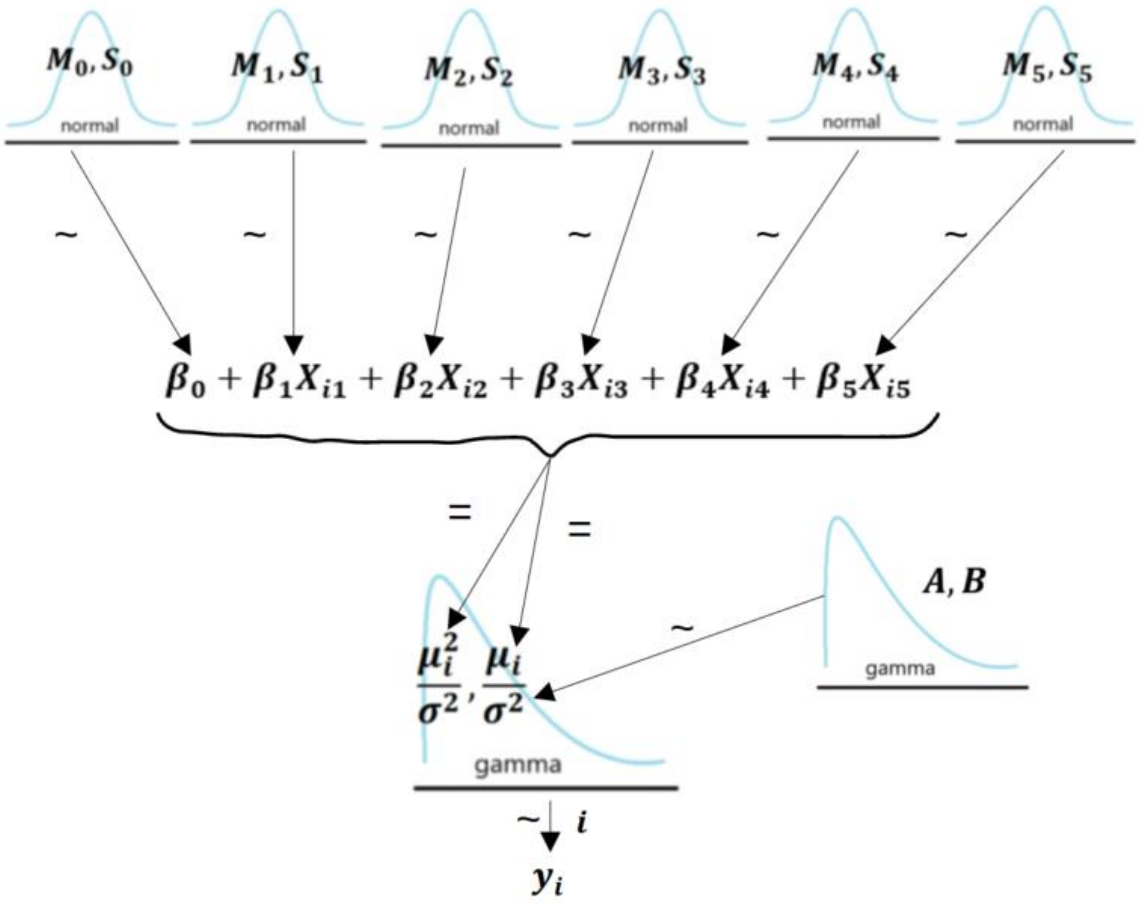

Figure 6: JAGS Model

The prior distribution is normally distributed for all the independent variables and here all the observations that is Ntotal is gamma distributed which is the likelihood. Here the gamma priors A and B are given as 0.01 since it is a non-informative prior. This model diagram is useful to write the JAGS model text from bottom to top.

- **Model Text**

```
modelString = "
# Standardize the data:
data {
  ysd <- sd(y)
  for ( i in 1:Ntotal ) {
    zy[i] <- y[i] / ysd
  }
  for ( j in 1:Nx ) {
    xsd[j] <-   sd(x[,j])
    for ( i in 1:Ntotal ) {
      zx[i,j] <- x[i,j] / xsd[j]
    }
  }
}
# Specify the model for scaled data:
model {
  for ( i in 1:Ntotal ) {
    zy[i] ~ dgamma( (mu[i]^2)/zVar , mu[i]/zVar )
    mu[i] <- zbeta0 + sum( zbeta[1:Nx] * zx[i,1:Nx] )
  }
  # Priors on standardized scale:
  zbeta0 ~ dnorm( 0 , 1/2^2 )
  zbeta[1] ~ dnorm( (90/100000)/xsd[1] , 1/(0.1/xsd[1]^2) )
  zbeta[2] ~ dnorm( 1/xsd[2] , 1/(4/xsd[2]^2) )
  zbeta[3] ~ dnorm( 0 , 1/4 )
  zbeta[4] ~ dnorm( 1.2/xsd[4] , 1/(1/xsd[4]^2) )
  zbeta[5] ~ dnorm( (-1.5)/xsd[5] , 1/(0.1/xsd[5]^2) )

  zVar ~ dgamma( 0.01 , 0.01 )
  # Transform to original scale:
  beta[1:Nx] <- ( zbeta[1:Nx] / xsd[1:Nx] ) * ysd
  beta0 <- zbeta0*ysd
  tau <- zVar * (ysd)^2

  # Compute predictions at every step of the MCMC
  for ( i in 1:5){
    pred[i] <- beta0 + beta[1] * xPred[i,1] + beta[2] * xPred[i,2] + beta[3] * xPred[i,3] + beta[4] * xPred[i,4]+ beta[5] * xPred[i,5]
  }


}
"
```

Figure 7: Model Text

First the data is standardized and then the model is specified on the standardized data. The model text is given with the necessary priors according to the expert information, as seen the zbeta0 mean is given as 0 and a large variance of 4 and for zbeta[1] the mean is given as 90/100000 as the data is in 100K and is standardized as the data is standardized and 0.1 is given as the variance along with standardisation as it is a very strong belief by the expert. While for zbeta[2] the mean is given as 1 as the data is in 100K and is standardized as the data is standardized and 4 is given as the variance along with standardisation as it is a weak belief by the expert. While for zbeta[3] the mean is given as 0 and 4 is given as the variance as there is no expert knowledge. While for zbeta[4] the mean is given as 1.2 as the data is in 100K and is standardized as the data is standardized and 1 is given as the variance along with standardisation as it is a strong belief by the expert . While for zbeta[5] the mean is given as -1.5 as the data is in 100K and is standarised as the data is standardized and it is negative as when the PropertyType is a Unit the SalePrice will decrease by 150K and 0.1 is given as the variance along with standardisation as it is a very strong belief by the expert . All of these standardized priors are normally distributed thus dnorm() function is used.

While the likelihood is taken as gamma distribution and thus dgamma() function is used with the A and B values of 0.01 and 0.01 respectively as it is non informative.

Then the beta values and tau is scaled back to the original scale. Followed by computing prediction at each step of the MCMC for the 5 predictions that is assigned. This model text is finally written to a text file “TEMPmodel.txt” so that JAGS can access it.

```
initsList <- list(
  zbeta0 = 6.25,
  zbeta = c(1.5, 1.3, 1.4, 1.2,1),
  var = 26.3
)
```

Figure 8: Initial List

The initial list is initialised using the list() function and different initial values were considered firstly with no initials which resulted in the parent values error as the random values that JAGS had initialised resulted in the analysis crashing thus started with a very high value of zbeta0 of 2000 and variance of 12000 but it was seen that the chains did not converge and thus changed the values. The zbeta0 was changed to a value near the mean of SalePrice which is 6.09 thus taking a value close to it that is 6.25 while the variance was also changed to a value close to the variance of the SalePrice which is 26.24 thus the value 26.3 was taken.

```
parameters = c( "zbeta0" ,  "zbeta" , "beta0" ,  "beta" ,  "tau", "zVar")

adaptSteps = 500
burnInSteps = 10000
nChains = 2
thinSteps = 3
numSavedSteps = 5000
nIter = ceiling( ( numSavedSteps * thinSteps ) / nChains )
```

Figure 9: Parameters

The parameters such as adapt steps, burn in steps, chains, thin steps and number of saved steps were also tuned to get maximum efficiency and accurate result. First it was ran with an adaptstep of 550 burn in steps of 2000, 2 chains and thinning step of 7 and number of saved steps as 10000 along with the first initial values but the result took way too long about 8 to 9 hours which did not seem reasonable and the chains hadn't converged at all along with a high shrink factor and non convergence in the density plot too. Thus the final setting of adapt steps of 500, burn in steps of 10000 after seeing the previous result, 2 chains and thinning steps of 3 as further thinning can be performed later and number of saved steps of 5000 was selected.

```
. Initializing model
. Adapting 500
-------------------------------------------------| 500
+++++++++++++++++++++++++++++++++++++++++++++++++ 100%
Adaptation successful
. Updating 10000
-------------------------------------------------| 10000
************************************************** 100%
. . . . . . . Updating 15000
-------------------------------------------------| 15000
************************************************** 100%
. . . . Updating 0
. Deleting model
All chains have finished
Simulation complete.  Reading coda files...
Coda files loaded successfully
Finished running the simulation
```

Figure 10: Model Running Output

```
> show(elapsedTime)
    user   system  elapsed
    9.73     3.42 10511.13
```

Figure 11: Elapsed RunTime

This Resulted in a run which successfully completed within 3 hours approximately, which seemed reasonable for 10000 observation dataset and seemed efficient.

```
# Parallel run
startTime = proc.time()
runJagsOut <- run.jags( method="parallel" ,
                        model="TEMPmodel.txt" ,
                        monitor=c( "zbeta0" ,  "zbeta" , "beta0" ,  "beta" ,  "tau", "zvar", "pred")  ,
                        data=dataList ,
                        inits=initsList ,
                        n.chains=nChains ,
                        adapt=adaptSteps ,
                        burnin=burnInSteps ,
                        sample=numSavedSteps ,
                        thin=thinSteps , summarise=FALSE , plots=FALSE )
codaSamples = as.mcmc.list( runJagsOut )
stopTime = proc.time()
elapsedTime = stopTime - startTime
show(elapsedTime)

save.image(file="absassignment2rprog1-prefinal-219241final.RData")

nrow(codaSamples[[1]])

# Do further thinning from the codaSamples
furtherThin <- 9
thiningSequence <- seq(1,nrow(codaSamples[[1]]), furtherThin)
newCodaSamples <- mcmc.list()
for ( i in 1:nChains){
  newCodaSamples[[i]] <- as.mcmc(codaSamples[[i]][thiningSequence,])
}

summary(codaSamples)
```

Figure 12: Parallel Run

Parallel method of running is used to run the JAGS model thus reducing the runtime and further thinning of 9 is done to reduce the autocorrelation and increase the accuracy.

```
> summary(codaSamples)

Iterations = 10501:25498
Thinning interval = 3
Number of chains = 2
Sample size per chain = 5000

1. Empirical mean and standard deviation for each variable,
   plus standard error of the mean:

              Mean        SD  Naive SE Time-series SE
zbeta0   9.770e-01 2.685e-02 2.685e-04      1.189e-03
zbeta[1] 2.887e-04 5.584e-04 5.584e-06      5.520e-06
zbeta[2] 1.884e-02 7.602e-03 7.602e-05      3.185e-04
zbeta[3] 4.912e-02 7.243e-03 7.243e-05      2.395e-04
zbeta[4] 1.701e-02 6.483e-03 6.483e-05      1.497e-04
zbeta[5] 2.229e-03 6.536e-03 6.536e-05      2.048e-04
beta0    5.006e+00 1.375e-01 1.375e-03      6.089e-03
beta[1]  2.619e-06 5.064e-06 5.064e-08      5.006e-08
beta[2]  1.076e-01 4.343e-02 4.343e-04      1.820e-03
beta[3]  4.149e-01 6.117e-02 6.117e-04      2.023e-03
beta[4]  1.052e-01 4.008e-02 4.008e-04      9.259e-04
beta[5]  2.456e-02 7.201e-02 7.201e-04      2.257e-03
tau      1.267e+01 2.243e-01 2.243e-03      2.478e-03
zVar     4.828e-01 8.547e-03 8.547e-05      9.443e-05
pred[1]  6.182e+00 7.095e-02 7.095e-04      1.542e-03
pred[2]  5.956e+00 5.616e-02 5.616e-04      1.412e-03
pred[3]  5.745e+00 6.459e-02 6.459e-04      2.393e-03
pred[4]  7.631e+00 1.496e-01 1.496e-03      4.677e-03
pred[5]  6.289e+00 7.242e-02 7.242e-04      1.736e-03

2. Quantiles for each variable:

               2.5%         25%         50%         75%     97.5%
zbeta0    0.9235226  9.588e-01 9.773e-01 9.954e-01 1.029e+00
zbeta[1] -0.0008049 -8.906e-05 2.946e-04 6.642e-04 1.383e-03
zbeta[2]  0.0039433  1.380e-02 1.883e-02 2.385e-02 3.409e-02
zbeta[3]  0.0348834  4.428e-02 4.908e-02 5.385e-02 6.360e-02
zbeta[4]  0.0043492  1.255e-02 1.696e-02 2.144e-02 2.997e-02
zbeta[5] -0.0105785 -2.121e-03 2.176e-03 6.562e-03 1.527e-02
beta0     4.7314280  4.912e+00 5.007e+00 5.100e+00 5.270e+00
beta[1]  -0.0000073 -8.077e-07 2.672e-06 6.024e-06 1.254e-05
beta[2]   0.0225274  7.882e-02 1.076e-01 1.363e-01 1.948e-01
beta[3]   0.2946047  3.740e-01 4.145e-01 4.548e-01 5.371e-01
beta[4]   0.0268914  7.760e-02 1.048e-01 1.326e-01 1.853e-01
beta[5]  -0.1165468 -2.336e-02 2.398e-02 7.229e-02 1.682e-01
tau      12.2455875  1.252e+01 1.267e+01 1.282e+01 1.312e+01
zVar      0.4665405  4.769e-01 4.828e-01 4.885e-01 4.998e-01
pred[1]   6.0436495  6.135e+00 6.182e+00 6.230e+00 6.324e+00
pred[2]   5.8457393  5.918e+00 5.956e+00 5.993e+00 6.066e+00
pred[3]   5.6155965  5.701e+00 5.745e+00 5.789e+00 5.870e+00
pred[4]   7.3388195  7.529e+00 7.630e+00 7.732e+00 7.929e+00
pred[5]   6.1475690  6.239e+00 6.289e+00 6.338e+00 6.430e+00
```

Figure 13: Summary

Summary of the codaSamples is outputted for each variable to check the mean and the standard deviation along with the quantiles.

MCMC Diagnostic Checking

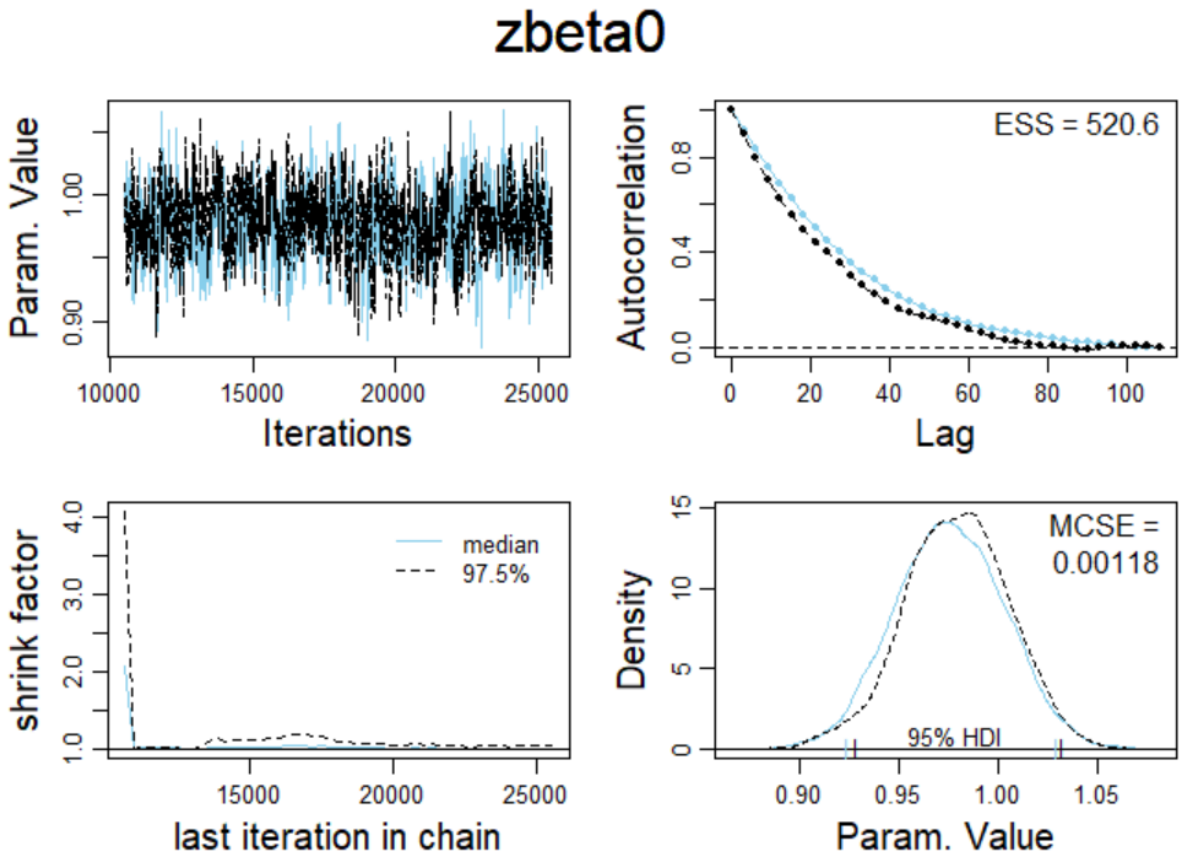

Figure 14:MCMC Diagnostics for zbeta0

Discussion

- The diagnostics show that for zbeta0 the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 1 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00118 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is above the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is not desirable.
- The thinning value of 3 shows that there is significant autocorrelation as seen in the autocorrelation plot. The effective sample size(ESS) is at 520.6.

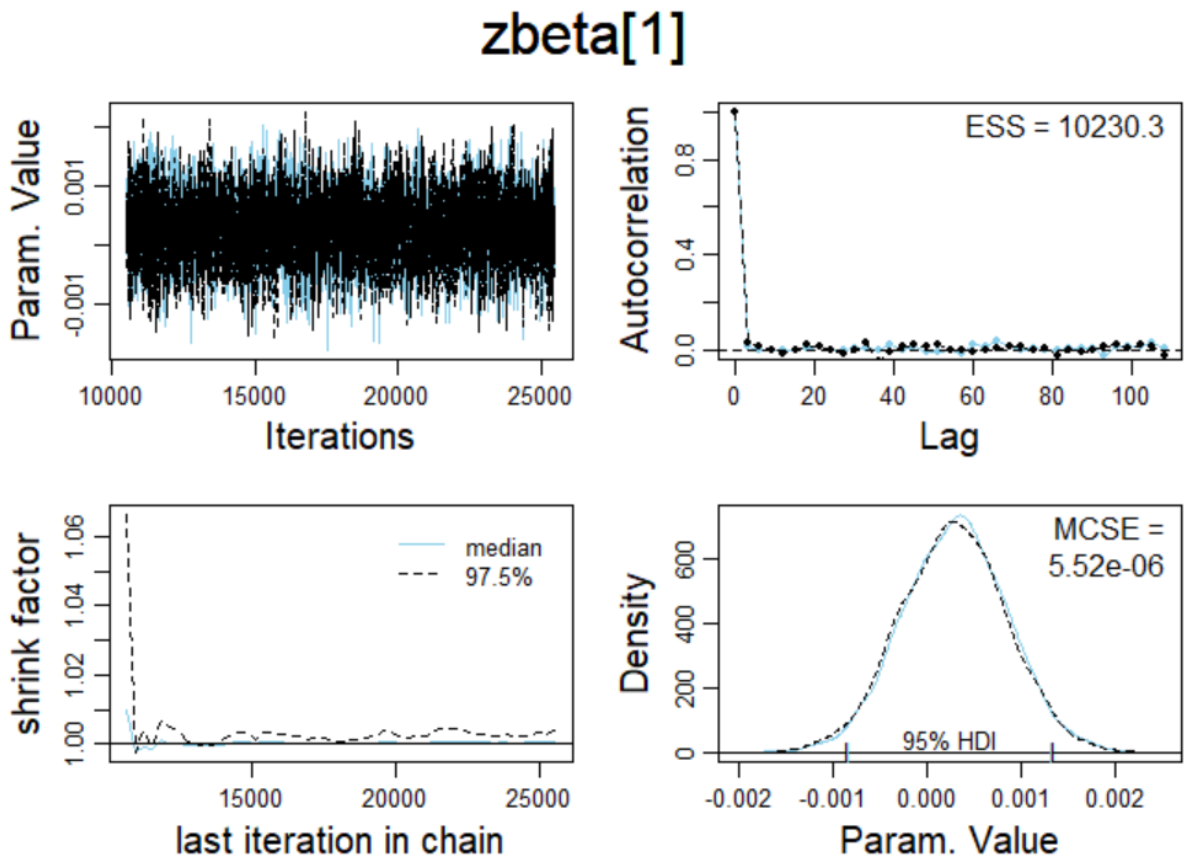

Figure 15:MCMC Diagnostics for zbeta[1]

Discussion

- The diagnostics show that for zbeta[1] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 5.52e-06 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The thinning value of 3 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 10230.3 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

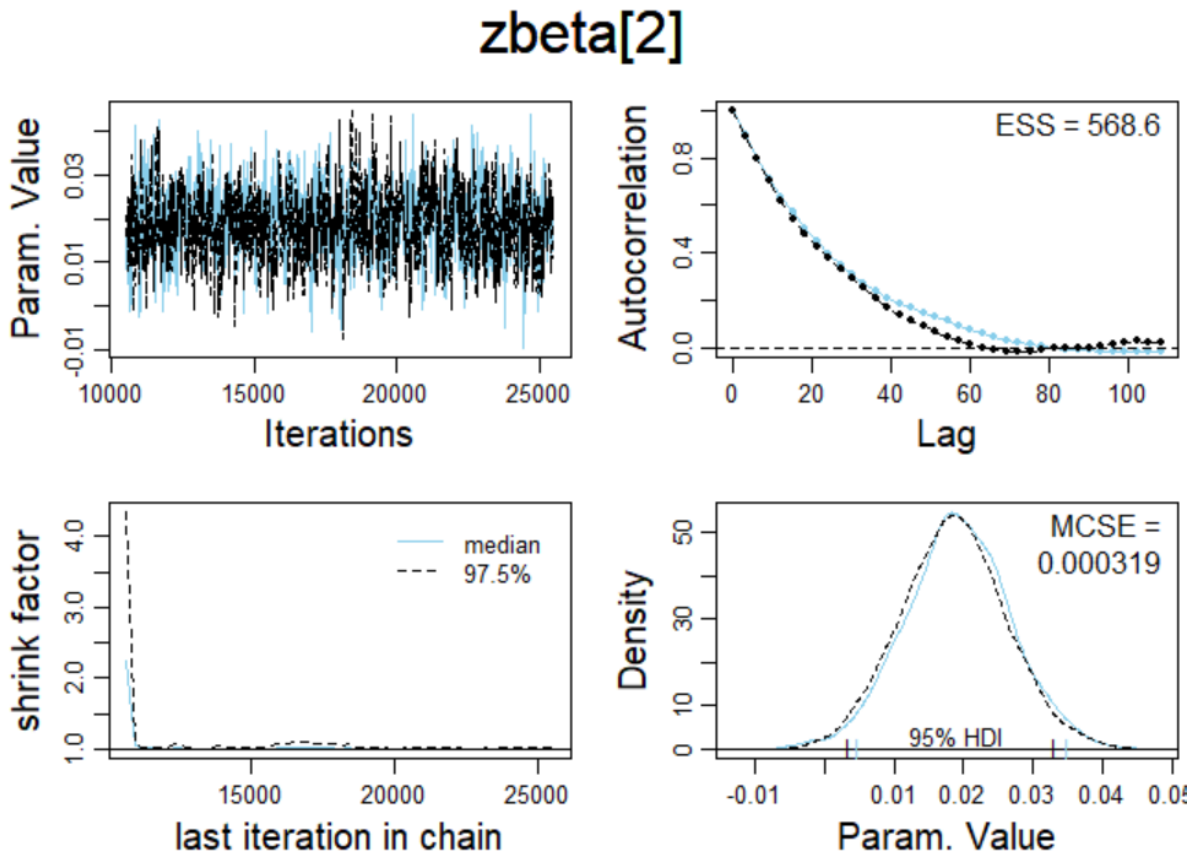

Figure 16:MCMC Diagnostics for zbeta[2]

Discussion

- The diagnostics show that for zbeta[2] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0.02 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.000319 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is above the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is not desirable.
- The thinning value of 3 shows that there is significant autocorrelation as seen in the autocorrelation plot. The effective sample size(ESS) is at 568.6.

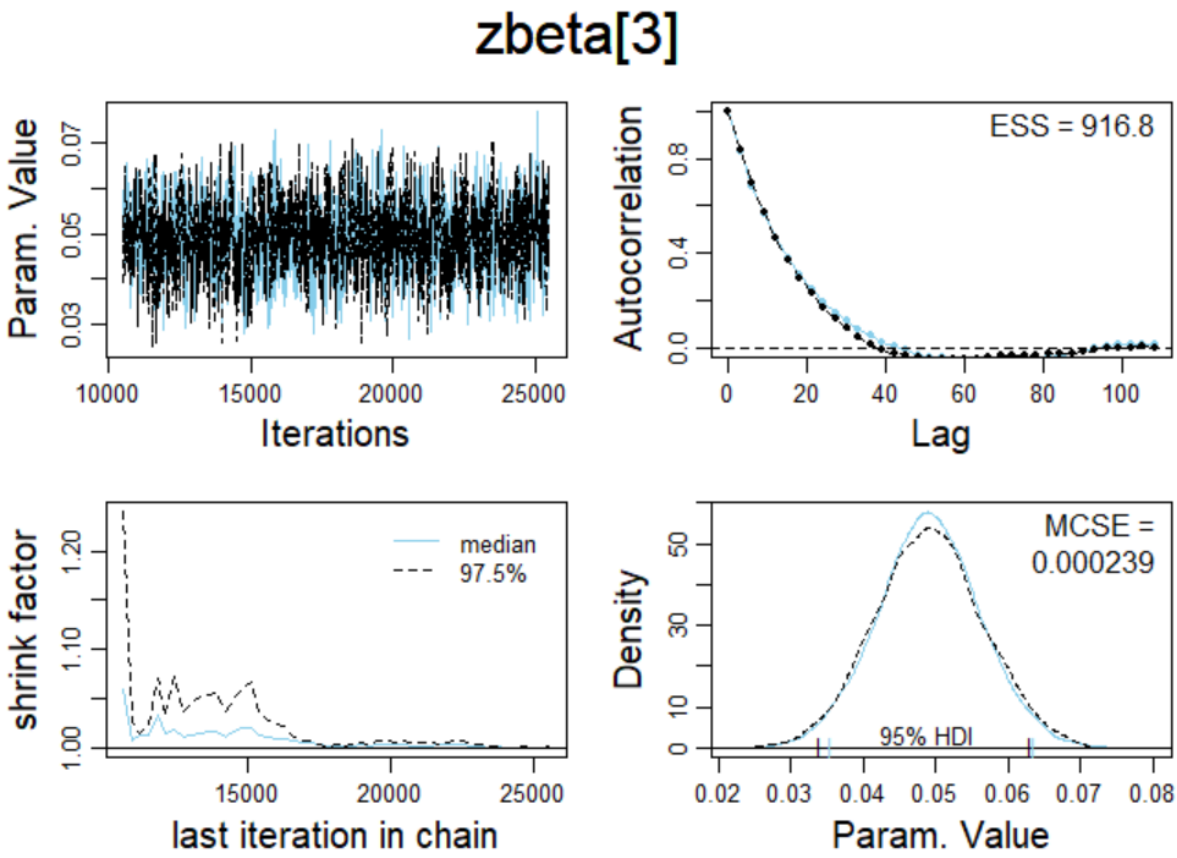

Figure 17:MCMC Diagnostics for zbeta[3]

Discussion

- The diagnostics show that for zbeta[3] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0.05 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.000239 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The thinning value of 3 shows that there is significant autocorrelation as seen in the autocorrelation plot. The effective sample size(ESS) is at 916.8.

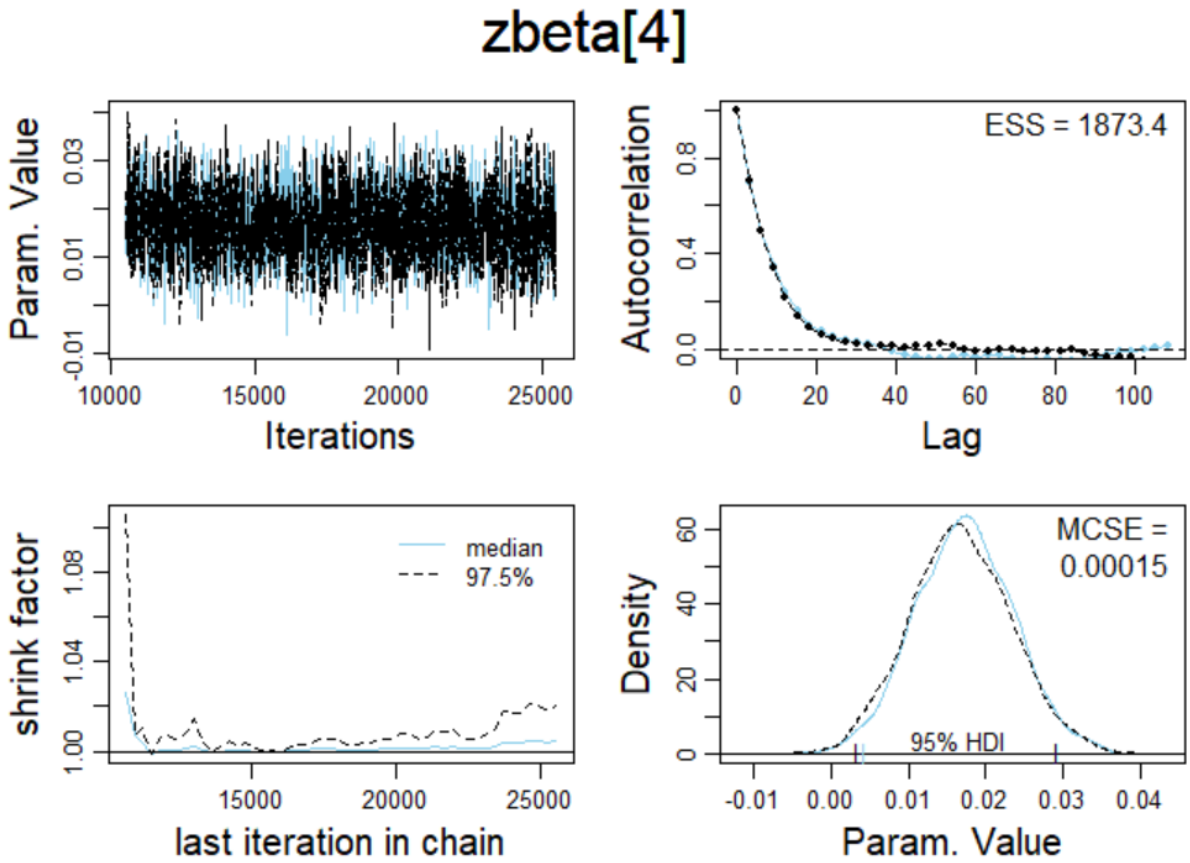

Figure 18:MCMC Diagnostics for zbeta[4]

Discussion

- The diagnostics show that for zbeta[4] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0.02 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00015 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The thinning value of 3 shows that there is significant autocorrelation as seen in the autocorrelation plot. The effective sample size(ESS) is at 1873.4.

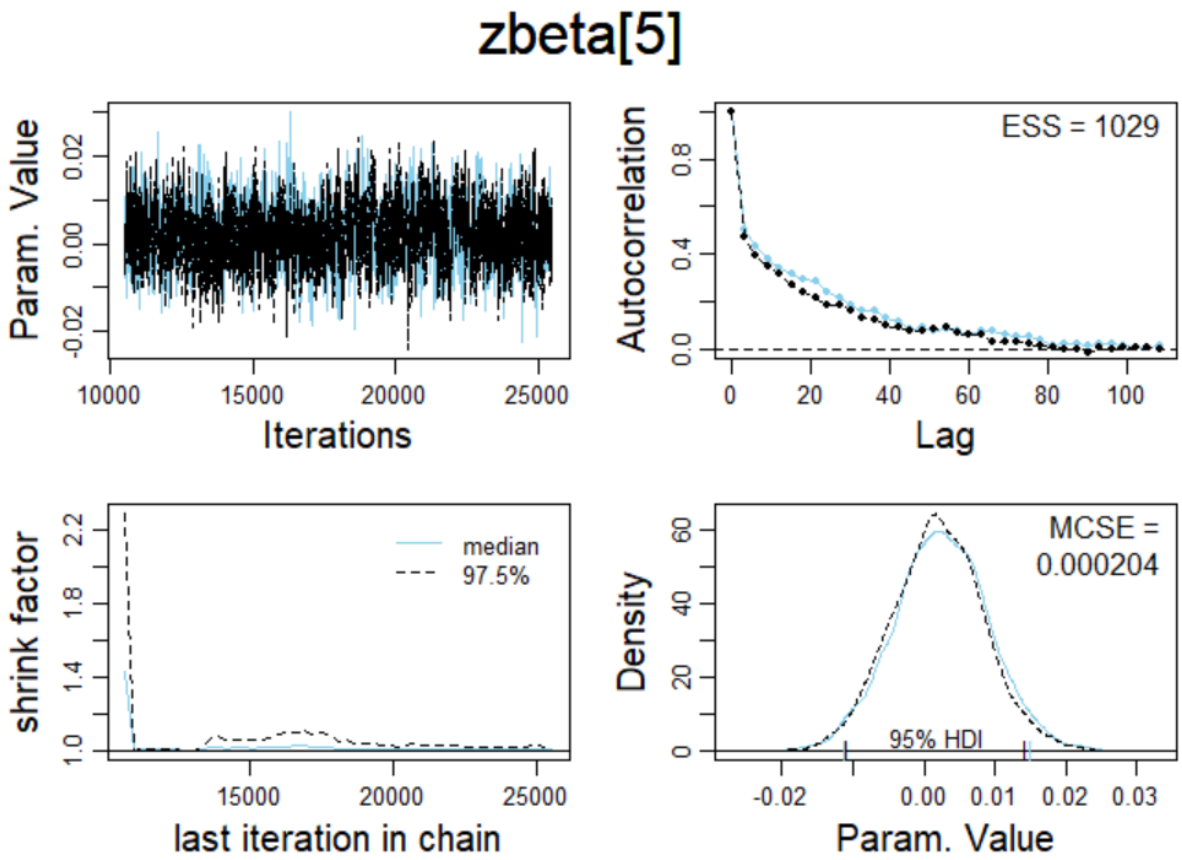

Figure 19:MCMC Diagnostics for zbeta[5]

Discussion

- The diagnostics show that for zbeta[5] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0.015 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.000204 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is above the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is not desirable.
- The thinning value of 3 shows that there is significant autocorrelation as seen in the autocorrelation plot. The effective sample size(ESS) is at 1029.

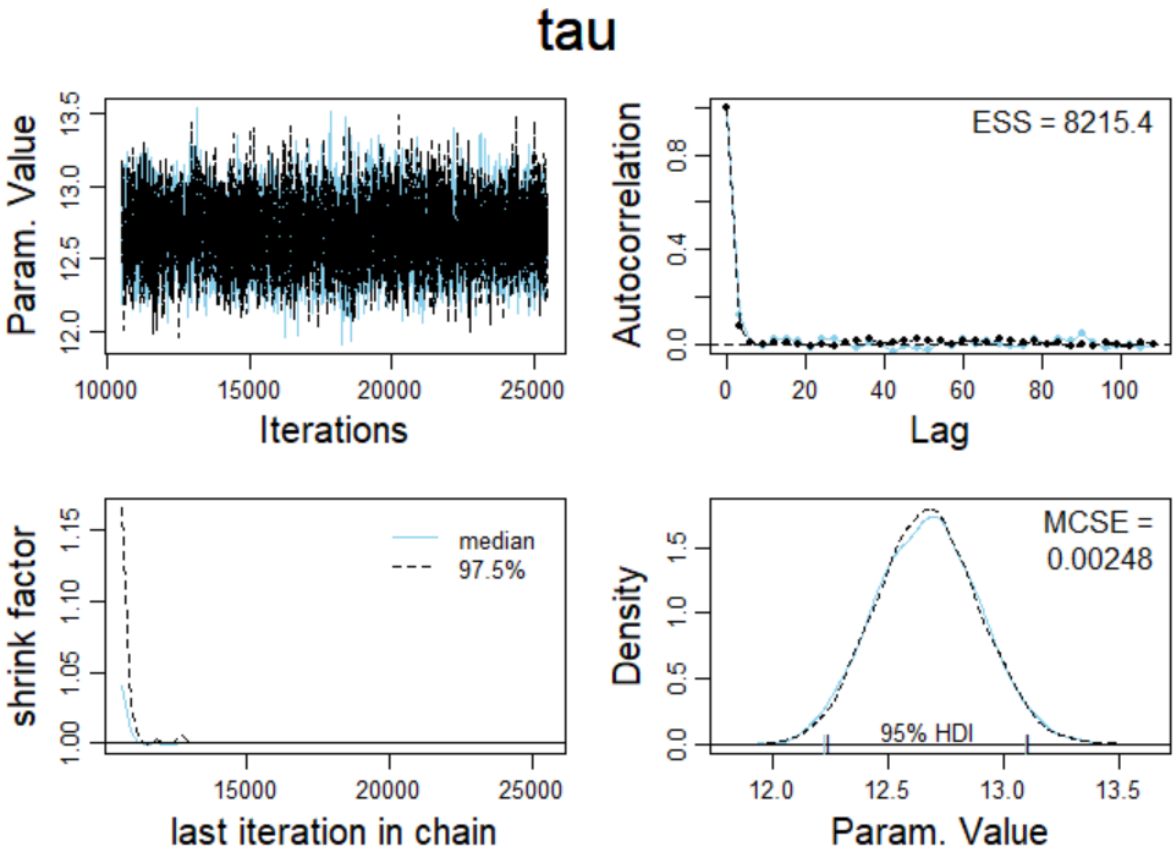

Figure 20:MCMC Diagnostics for tau

Discussion

- The diagnostics show that for tau the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 12.7 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00248 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The thinning value of 3 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 8215.4 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

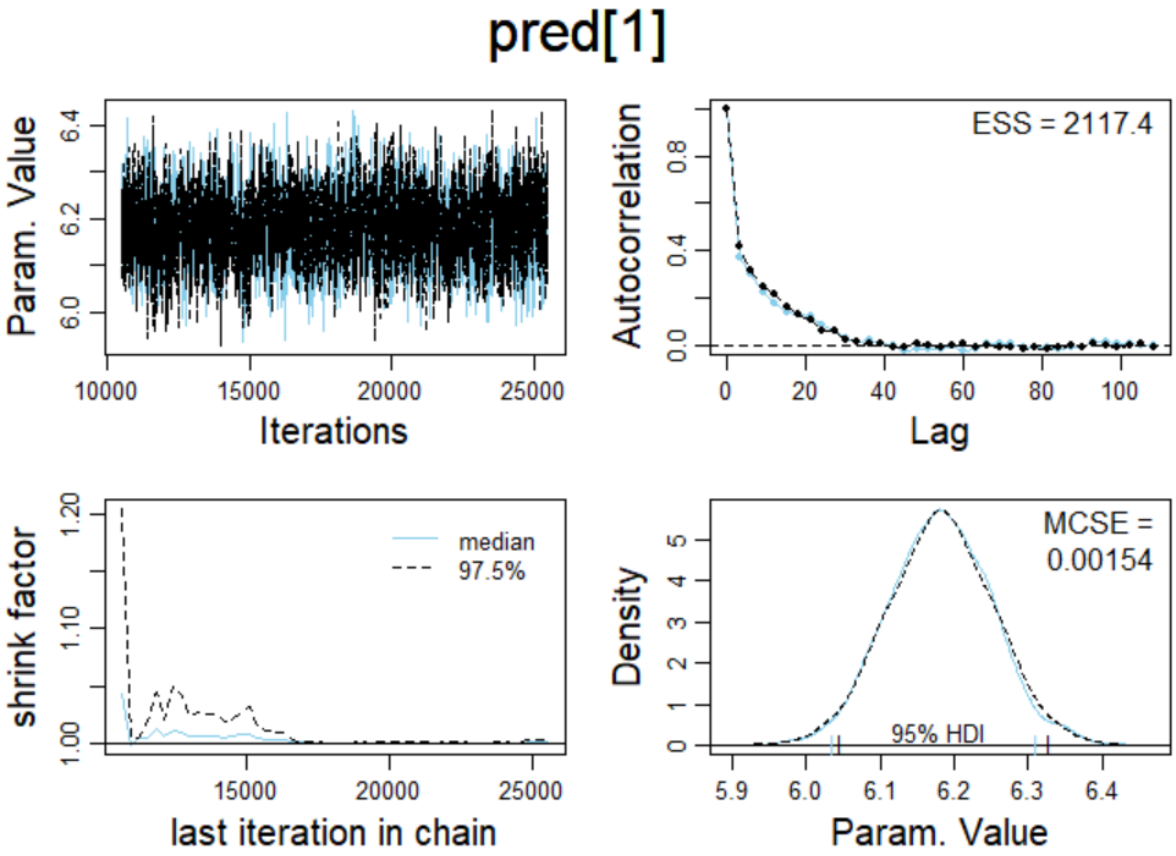

Figure 21:MCMC Diagnostics for pred[1]

Discussion

- The diagnostics show that for pred[1] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 6.2 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00154 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The thinning value of 3 shows that there is significant autocorrelation as seen in the autocorrelation plot. The effective sample size(ESS) is at 2117.4.

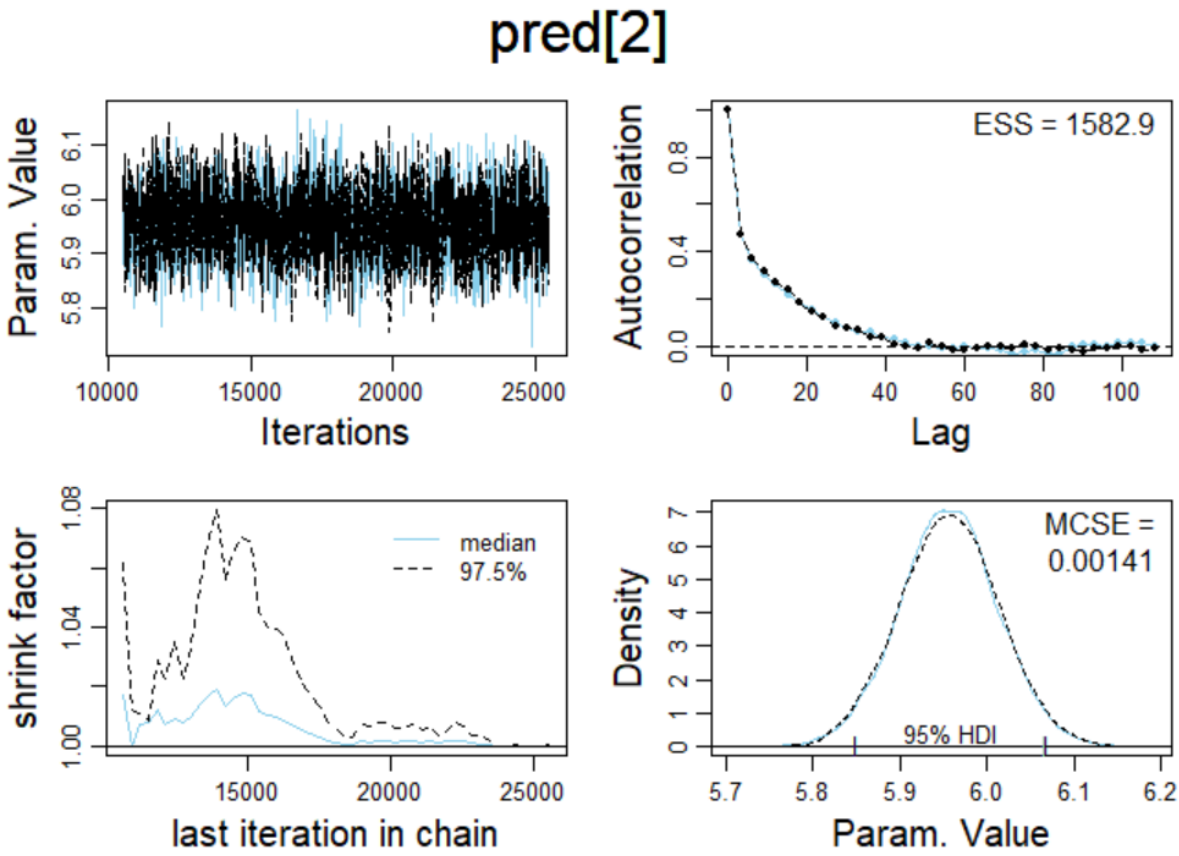

Figure 22:MCMC Diagnostics for pred[2]

Discussion

- The diagnostics show that for pred[2] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 5.95 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00141 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The thinning value of 3 shows that there is significant autocorrelation as seen in the autocorrelation plot. The effective sample size(ESS) is at 1582.9.

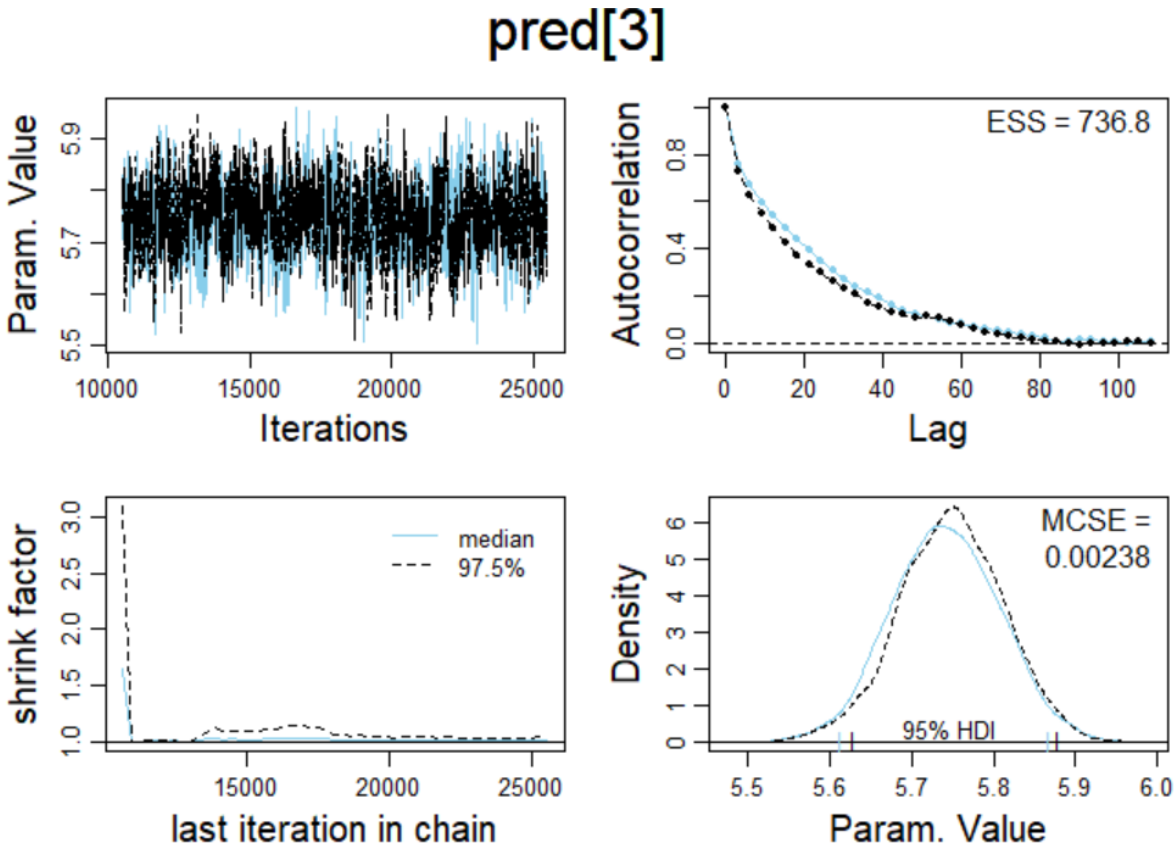

Figure 23:MCMC Diagnostics for pred[3]

Discussion

- The diagnostics show that for pred[3] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 5.75 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00238 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is above the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is not desirable.
- The thinning value of 3 shows that there is significant autocorrelation as seen in the autocorrelation plot. The effective sample size(ESS) is at 736.8.

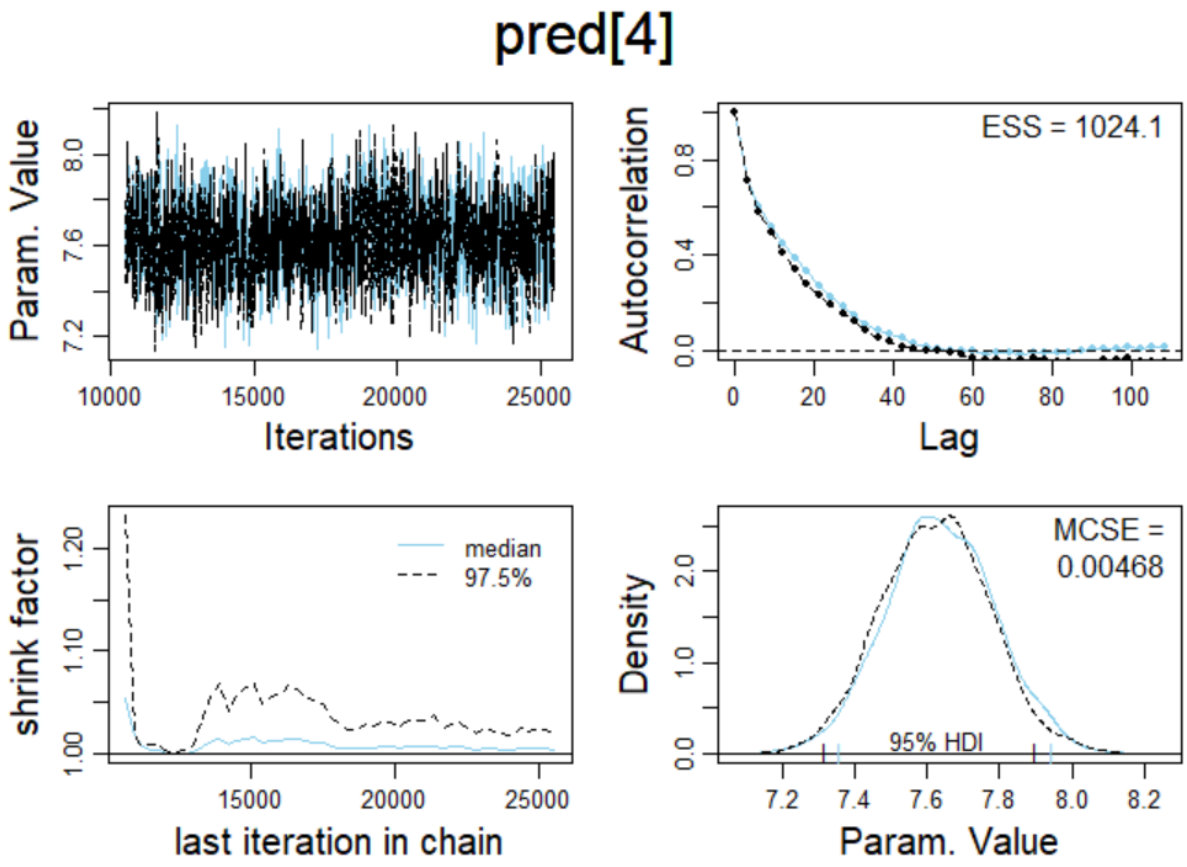

Figure 24:MCMC Diagnostics for pred[4]

Discussion

- The diagnostics show that for pred[4] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 7.6 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00468 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The thinning value of 3 shows that there is significant autocorrelation as seen in the autocorrelation plot. The effective sample size(ESS) is at 1024.1.

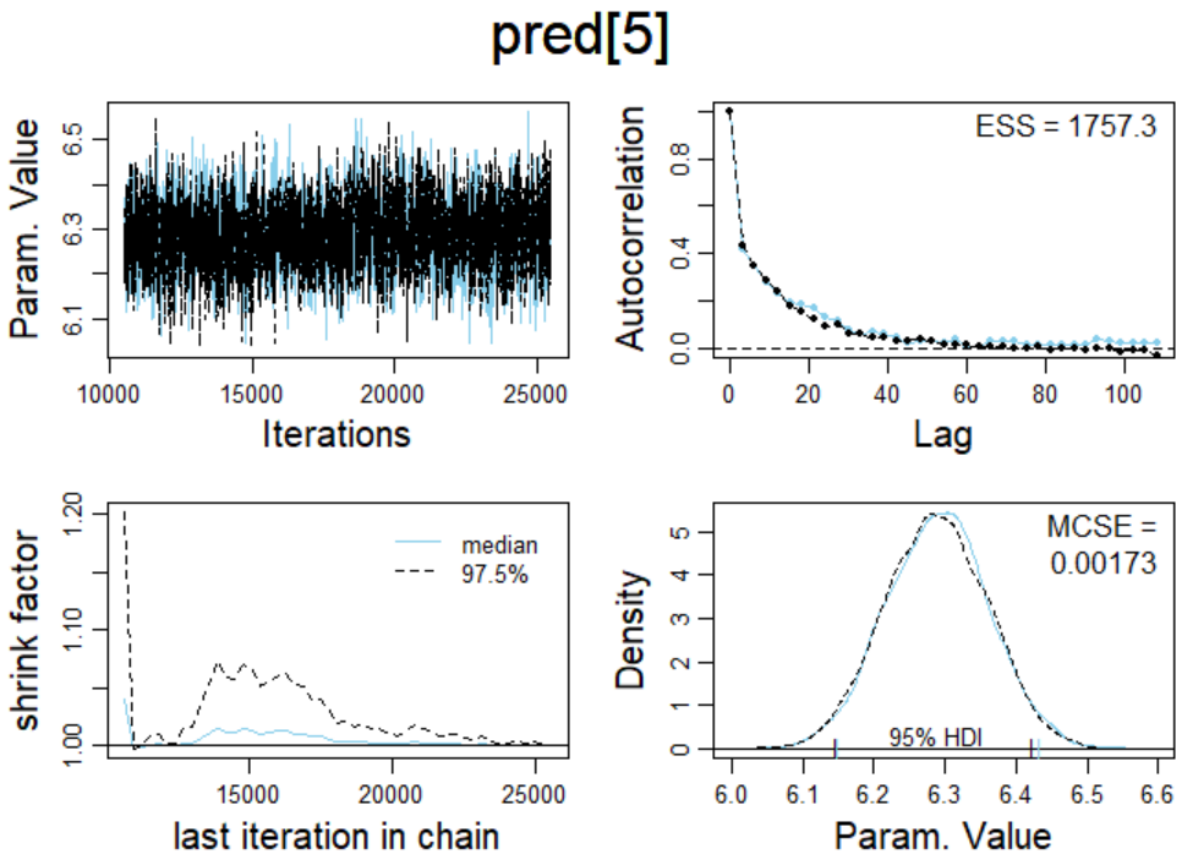

Figure 25:MCMC Diagnostics for pred[5]

Discussion

- The diagnostics show that for pred[5] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 6.3 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00173 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The thinning value of 3 shows that there is significant autocorrelation as seen in the autocorrelation plot. The effective sample size(ESS) is at 1757.3.

Further Thinning

Since there is significant autocorrelation present in the standardized values further thinning is executed to reduce the autocorrelation present which is done by the following code:

```
furtherThin <- 11
thiningSequence <- seq(1,nrow(codaSamples[[1]]), furtherThin)
newCodaSamples <- mcmc.list()
for ( i in 1:nChains){
  newCodaSamples[[i]] <- as.mcmc(codaSamples[[i]][thiningSequence,])
}
```

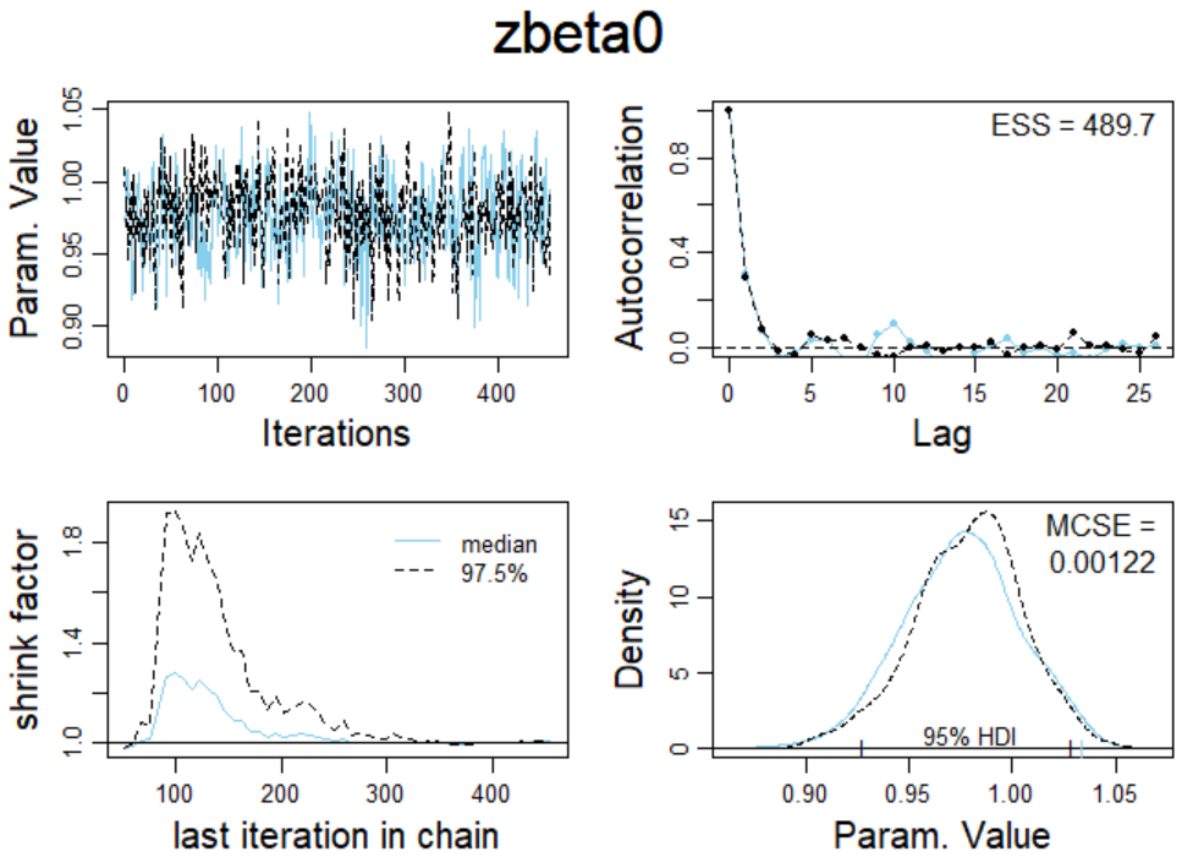

Figure 26:MCMC Diagnostics for zbeta0

Discussion

- The diagnostics show that for zbeta0 the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0.97 in the trace plot and thus it is representative of the posterior.

- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00122 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 489.7 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

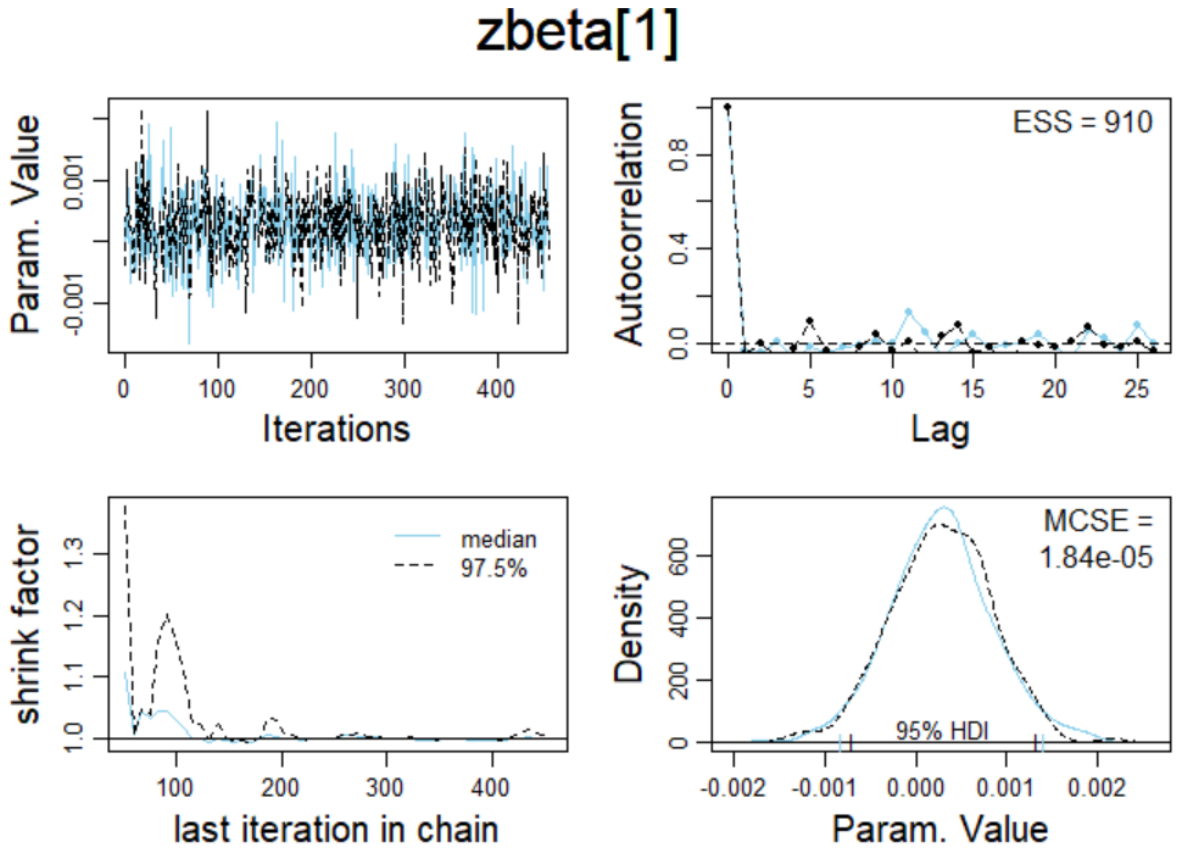

Figure 27:MCMC Diagnostics for zbeta[1]

Discussion

- The diagnostics show that for zbeta[1] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 1.84e-05 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable,

and it can be confirmed that the chains have converged effectively and is representative of the posterior.

- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 910 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

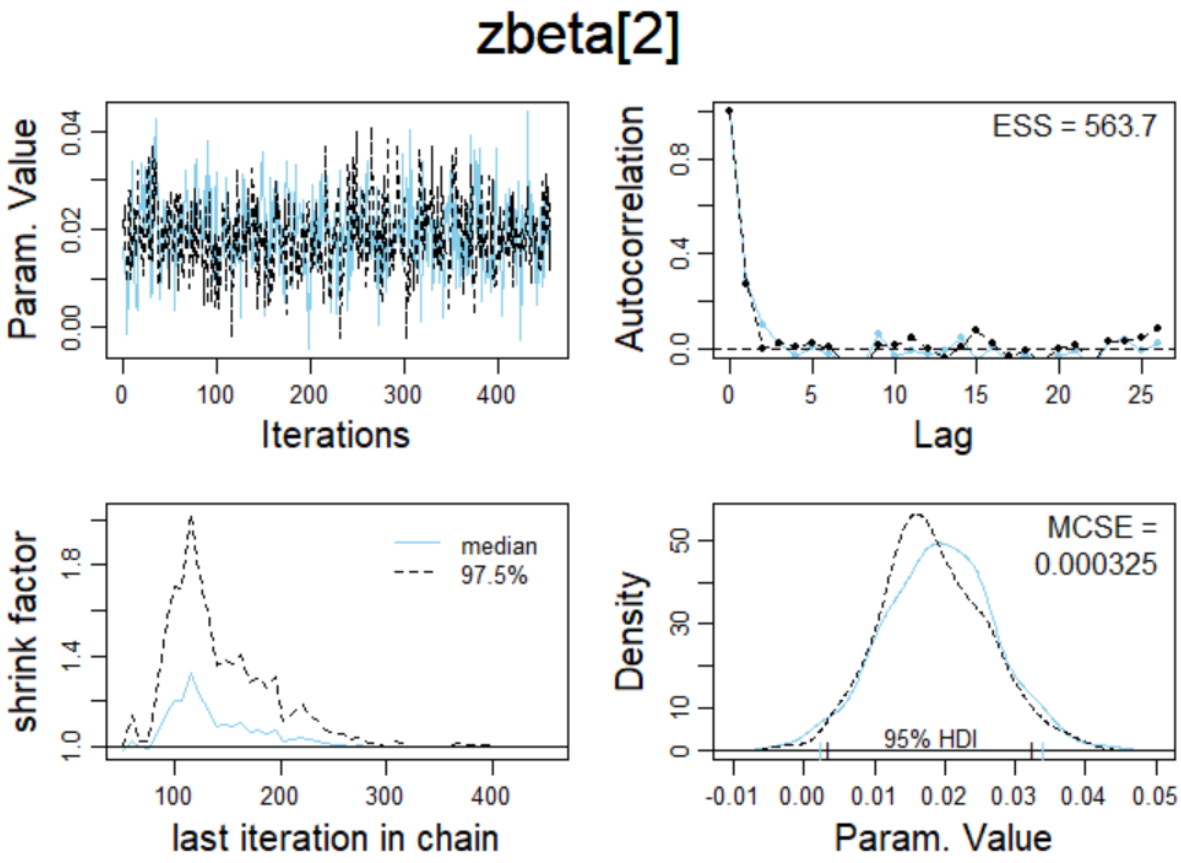

Figure 28:MCMC Diagnostics for zbeta[2]

Discussion

- The diagnostics show that for zbeta[2] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0.02 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.000325 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant

autocorrelation. The effective sample size(ESS) is at 563.7 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

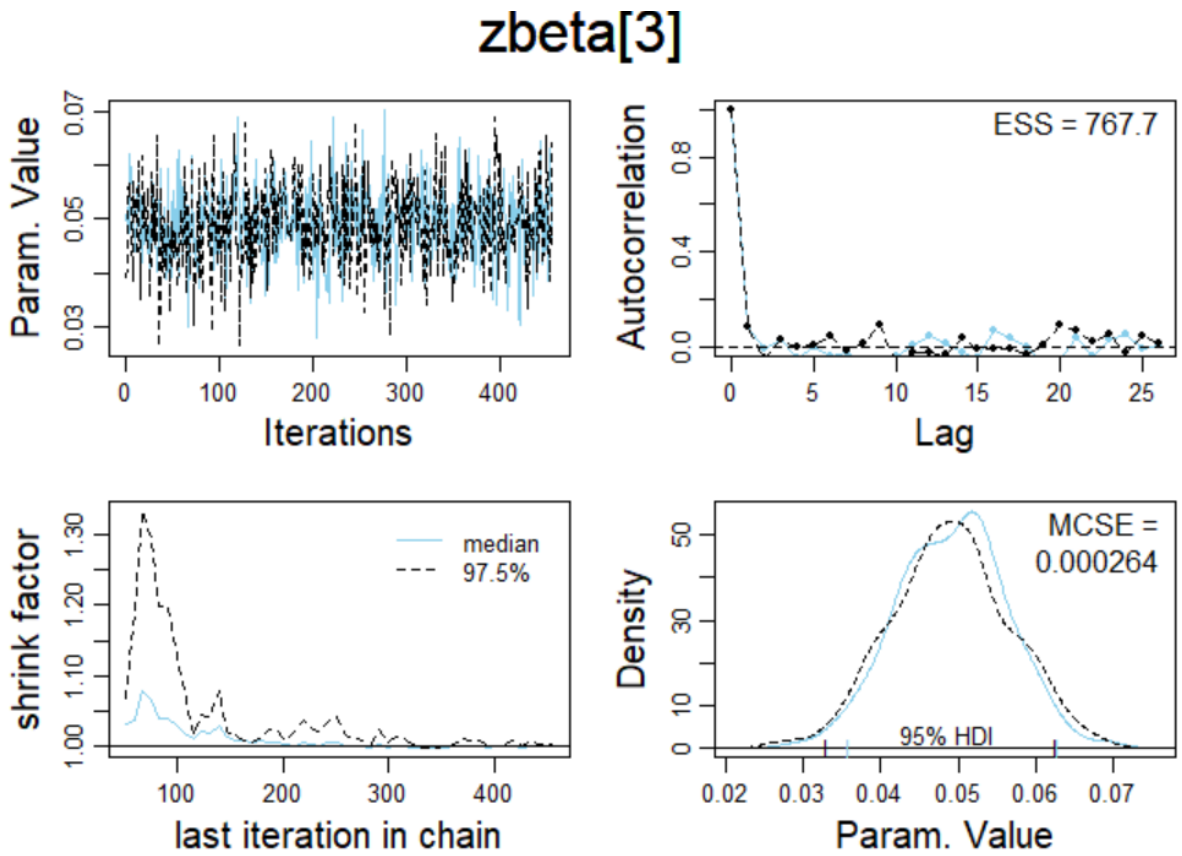

Figure 29:MCMC Diagnostics for zbeta[3]

Discussion

- The diagnostics show that for zbeta[3] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0.05 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.000264 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 767.7 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

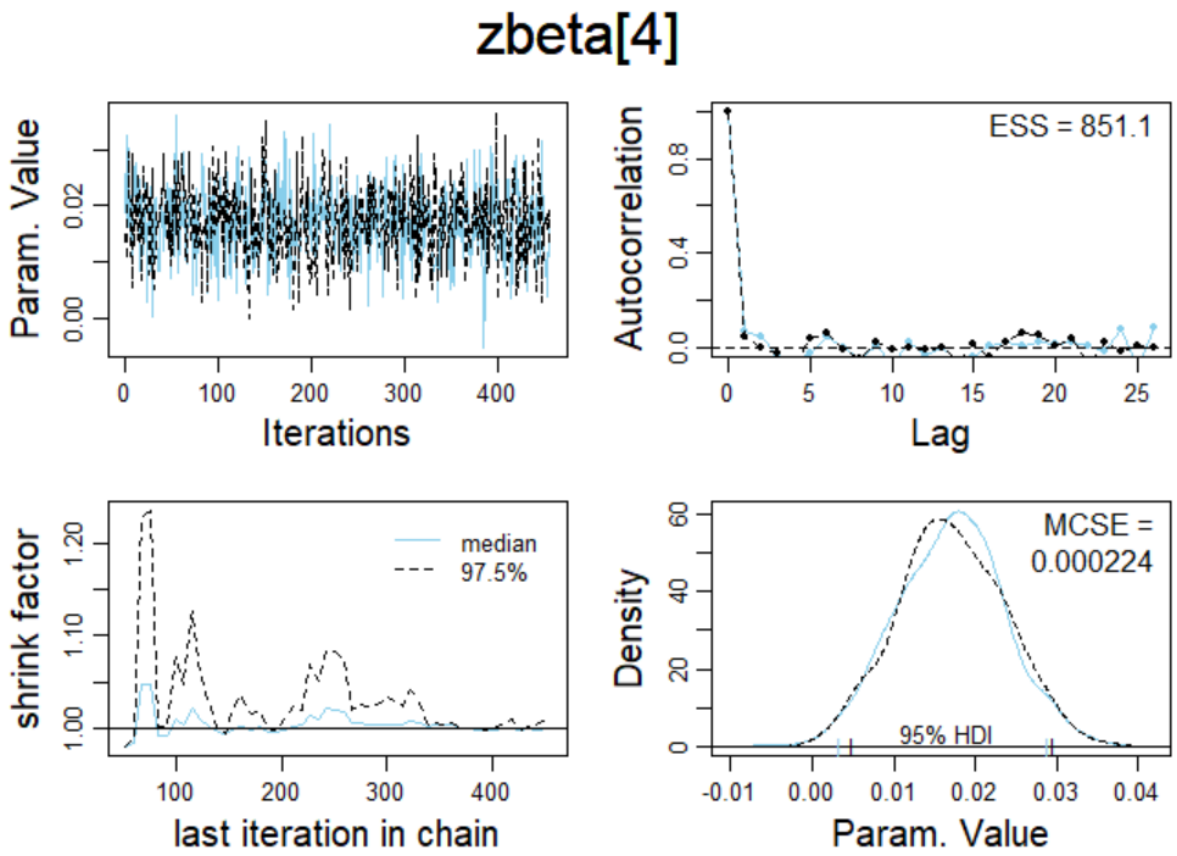

Figure 30:MCMC Diagnostics for zbeta[4]

Discussion

- The diagnostics show that for zbeta[4] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0.02 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.000224 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 851.1 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

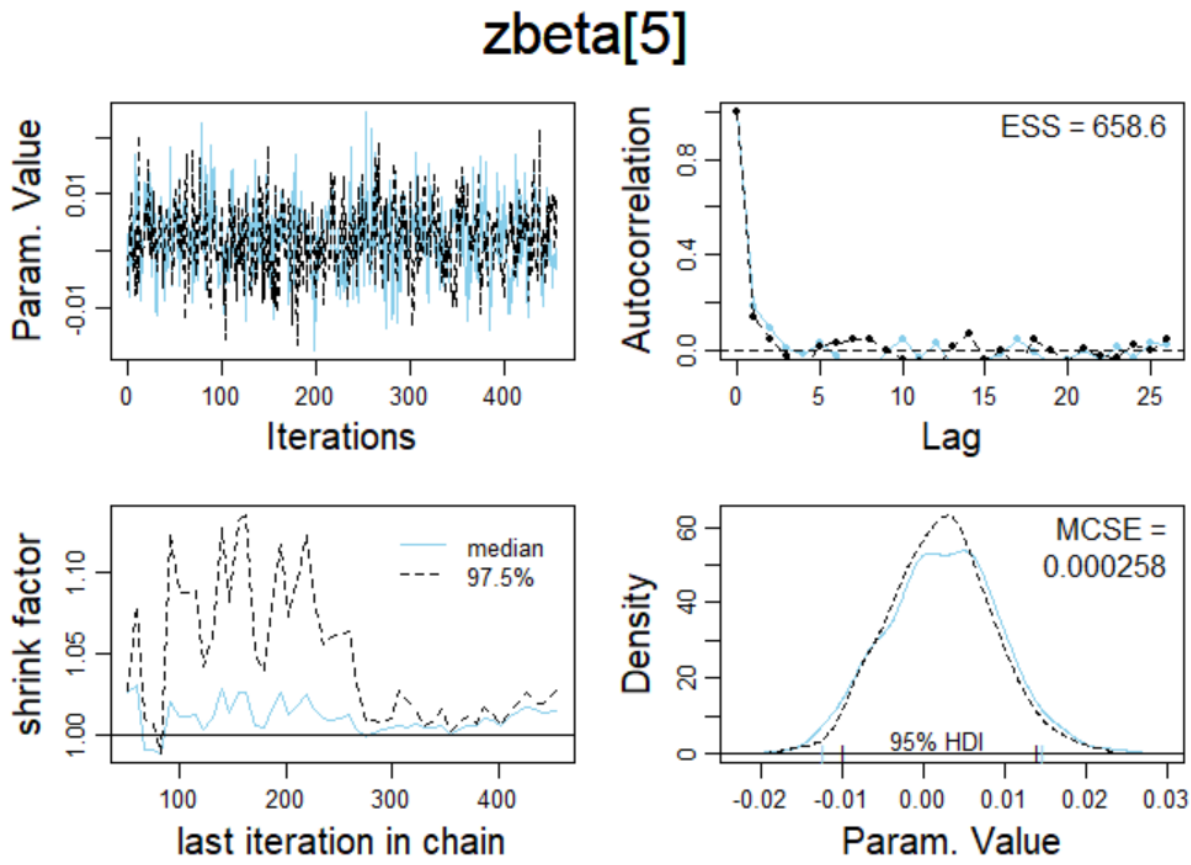

Figure 31:MCMC Diagnostics for zbeta[5]

Discussion

- The diagnostics show that for zbeta[5] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 0 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.000258 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 658.6 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

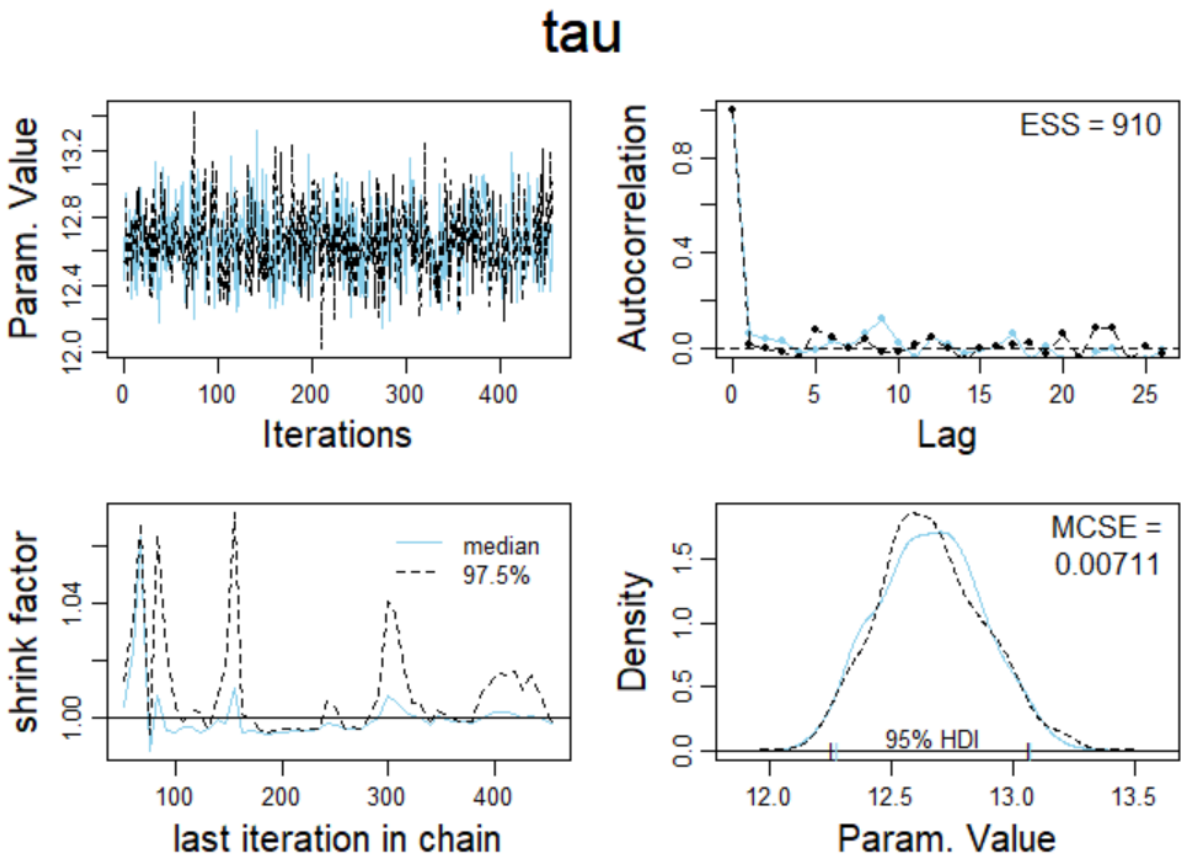

Figure 32:MCMC Diagnostics for tau

Discussion

- The diagnostics show that for tau the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 12.7 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00711 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 910 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

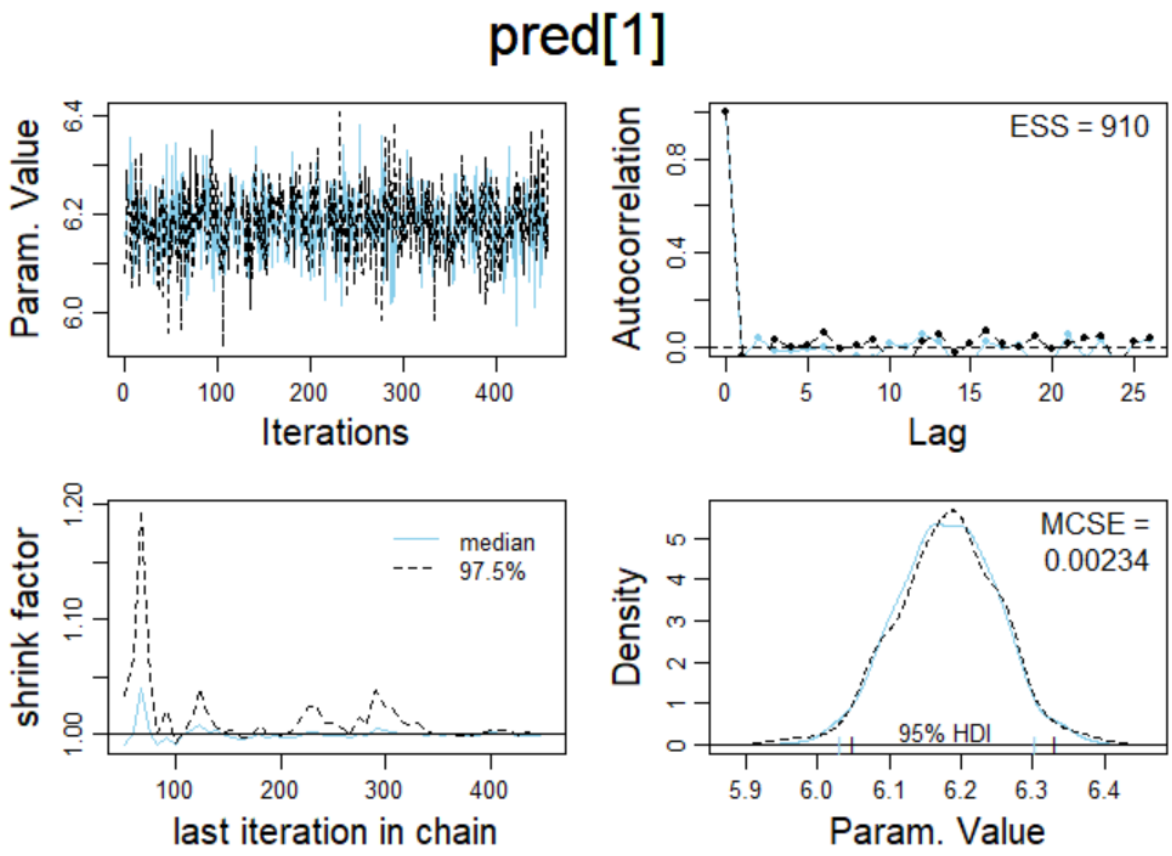

Figure 33:MCMC Diagnostics for pred[1]

Discussion

- The diagnostics show that for pred[1] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 6.2 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00234 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 910 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

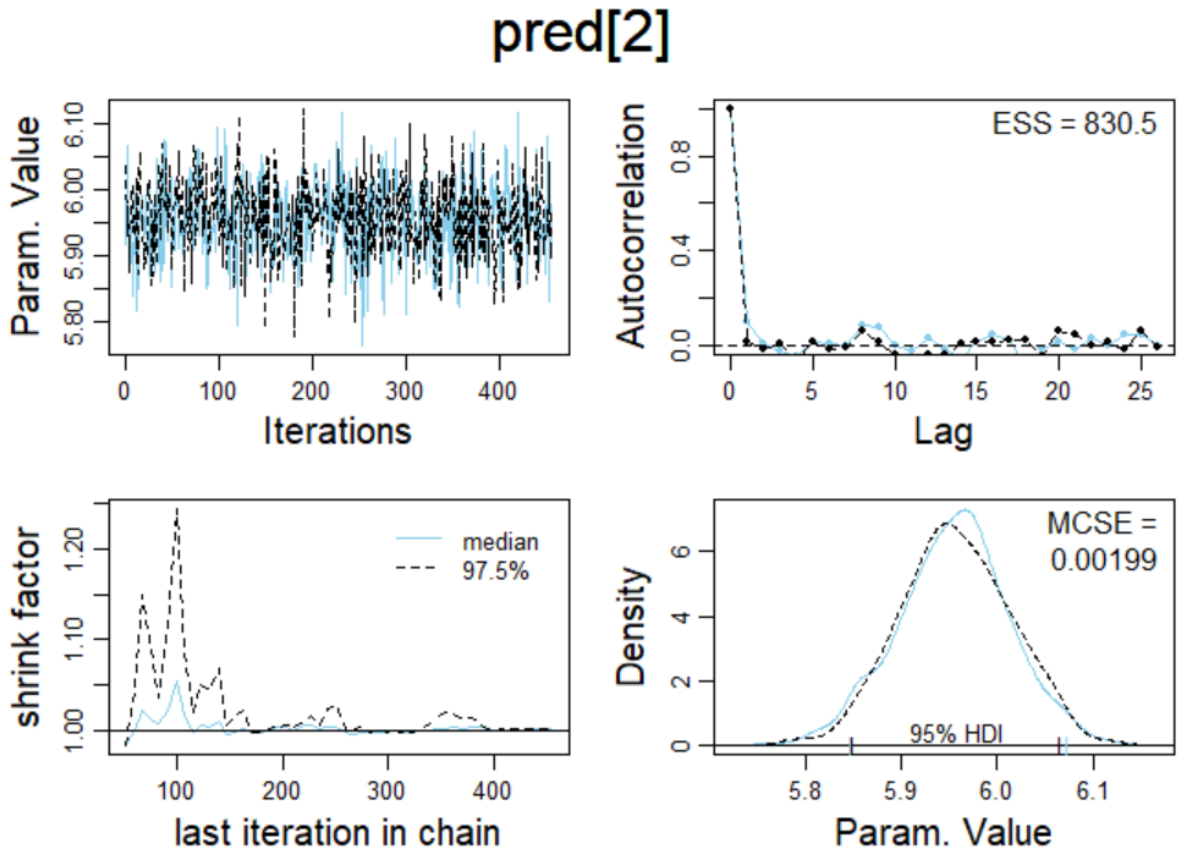

Figure 34:MCMC Diagnostics for pred[2]

Discussion

- The diagnostics show that for pred[2] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 5.95 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00199 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 830.5 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

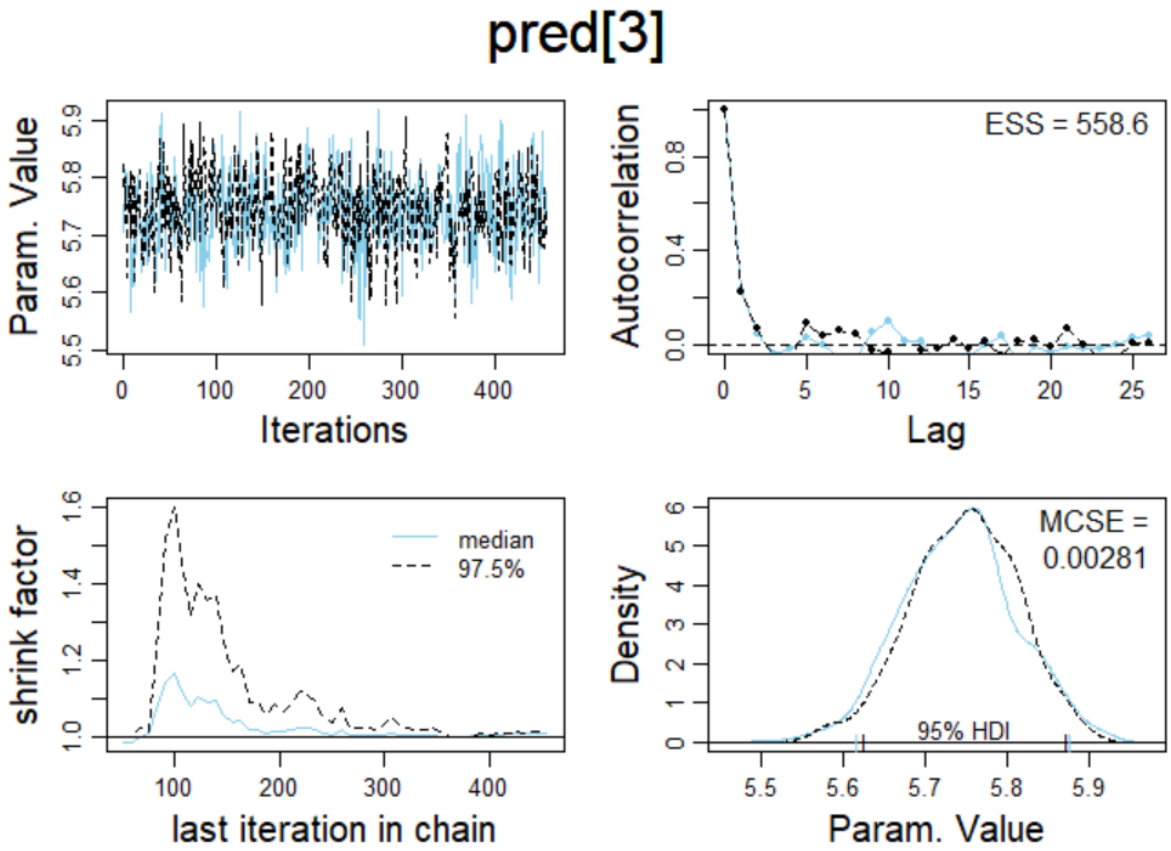

Figure 35:MCMC Diagnostics for pred[3]

Discussion

- The diagnostics show that for pred[3] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 5.7 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00281 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 558.6 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

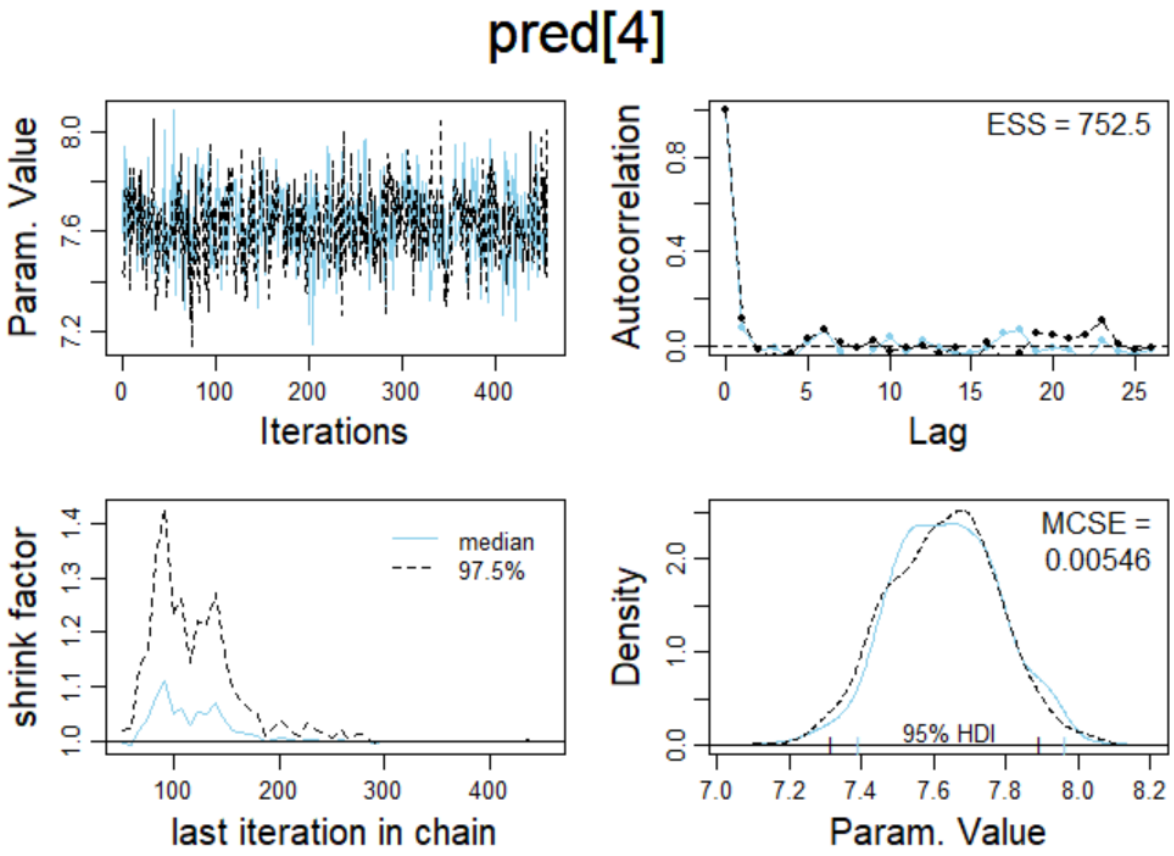

Figure 36:MCMC Diagnostics for pred[4]

Discussion

- The diagnostics show that for pred[4] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 7.6 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00546 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 752.5 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

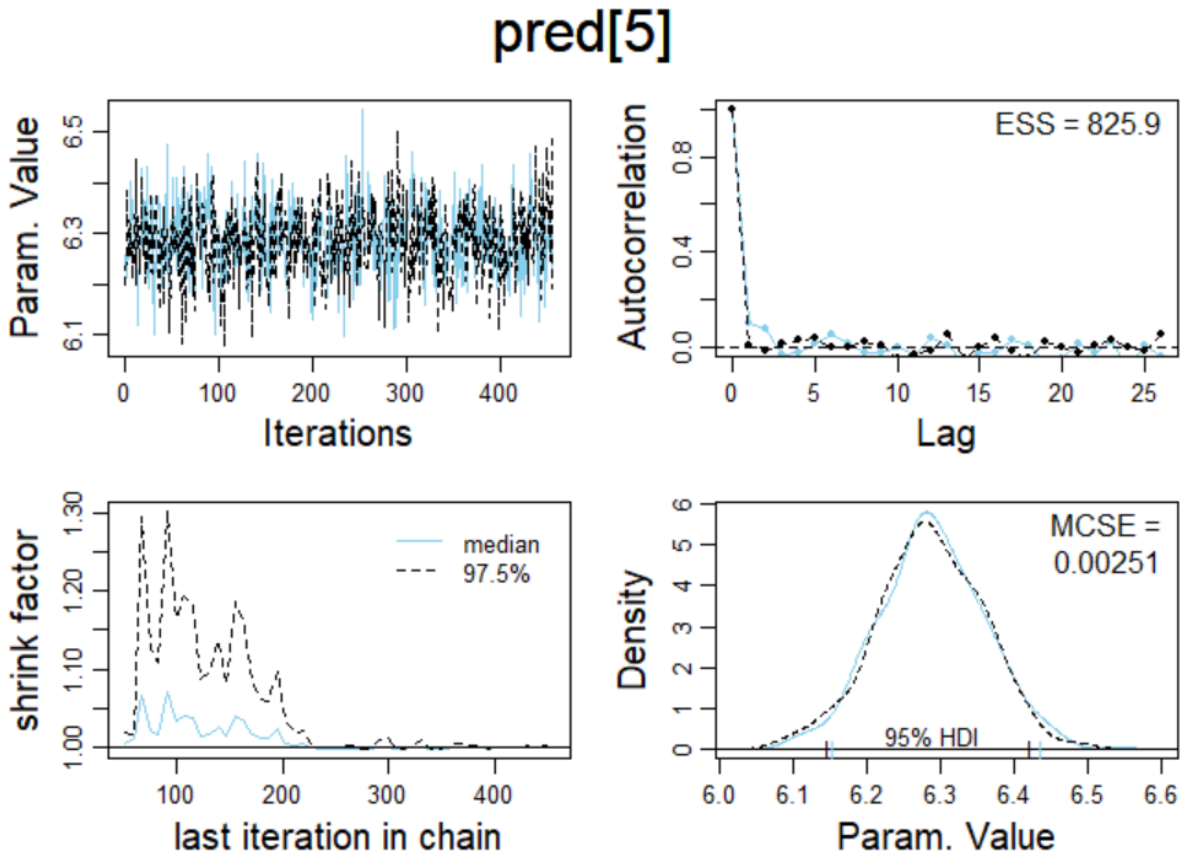

Figure 37:MCMC Diagnostics for pred[5]

Discussion

- The diagnostics show that for pred[5] the mean with burn in steps of 10000 for the 2 chains converged to a mean value of 6.3 in the trace plot and thus it is representative of the posterior.
- The density plot showed that the two chains converged effectively along with a very small Monte Carlo Standard error (MCSE) values at 0.00251 which is nearly zero which is desirable and confirms that it is representative of the posterior.
- The shrink factor is below the value 1.1 which is represented by median line of the shrink factor(Gelman-Rubin statistic) plot beyond the burn in period which is desirable, and it can be confirmed that the chains have converged effectively and is representative of the posterior.
- The further thinning value of 11 shows that the autocorrelation is effectively near at the zero value as seen in the autocorrelation plot and thus there is no significant autocorrelation. The effective sample size(ESS) is at 825.9 which is a high value and is desriable and confirms that the posterior estimates are accurate and stable.

Predictive Check

```
summaryInfo <- smryMCMC_HD( codaSamples = newCodaSamples , compVal = compVal )
print(summaryInfo)
```

	Mean	Median	Mode	ESS	HDImass	HDIlow	HDIhigh	CompVal	PcntGtCompVal	ROPElow	ROPEhigh	PcntLtROPE	PcntInROPE
CHAIN	1.500000e+00	1.500000e+00	1.998182e+00	1.5	0.95	1.00000e+00	2.00000e+00	NA	NA	NA	NA	NA	NA
zbeta0	9.769143e-01	9.783315e-01	9.850009e-01	487.6	0.95	9.27260e-01	1.03221e+00	NA	NA	NA	NA	NA	NA
zbeta[1]	2.806584e-04	2.809140e-04	2.829803e-04	910.0	0.95	-7.63027e-04	1.40451e-03	NA	NA	NA	NA	NA	NA
zbeta[2]	1.882294e-02	1.841155e-02	1.710121e-02	517.3	0.95	3.40505e-03	3.39350e-02	NA	NA	NA	NA	NA	NA
zbeta[3]	4.911713e-02	4.910445e-02	5.134232e-02	768.8	0.95	3.48650e-02	6.29667e-02	NA	NA	NA	NA	NA	NA
zbeta[4]	1.707364e-02	1.709320e-02	1.718934e-02	814.2	0.95	4.85224e-03	3.02419e-02	NA	NA	NA	NA	NA	NA
zbeta[5]	2.225325e-03	2.234500e-03	3.486410e-03	655.3	0.95	-1.11318e-02	1.46759e-02	NA	NA	NA	NA	NA	NA
beta0	5.004965e+00	5.012225e+00	5.046393e+00	487.6	0.95	4.75057e+00	5.28824e+00	NA	NA	NA	NA	NA	NA
beta[1]	2.545459e-06	2.547780e-06	2.566476e-06	910.0	0.95	-6.92035e-06	1.27383e-05	NA	NA	NA	NA	NA	NA
beta[2]	1.075326e-01	1.051825e-01	9.769677e-02	517.3	0.95	1.94525e-02	1.93865e-01	NA	NA	NA	NA	NA	NA
beta[3]	4.148146e-01	4.147075e-01	4.336074e-01	768.8	0.95	2.94450e-01	5.31780e-01	NA	NA	NA	NA	NA	NA
beta[4]	1.055688e-01	1.056895e-01	1.062844e-01	814.2	0.95	3.00021e-02	1.86990e-01	NA	NA	NA	NA	NA	NA
beta[5]	2.451719e-02	2.461825e-02	3.841099e-02	655.3	0.95	-1.22643e-01	1.61689e-01	NA	NA	NA	NA	NA	NA
tau	1.266345e+01	1.265925e+01	1.259116e+01	910.0	0.95	1.22584e+01	1.30584e+01	NA	NA	NA	NA	NA	NA
zVar	4.824619e-01	4.823010e-01	4.797070e-01	910.0	0.95	4.67031e-01	4.97509e-01	NA	NA	NA	NA	NA	NA
pred[1]	6.181272e+00	6.183585e+00	6.192381e+00	910.0	0.95	6.04937e+00	6.32359e+00	NA	NA	NA	NA	NA	NA
pred[2]	5.955551e+00	5.955680e+00	5.958129e+00	817.6	0.95	5.84896e+00	6.06809e+00	NA	NA	NA	NA	NA	NA
pred[3]	5.744232e+00	5.746735e+00	5.757151e+00	565.8	0.95	5.62702e+00	5.87943e+00	NA	NA	NA	NA	NA	NA
pred[4]	7.630525e+00	7.629925e+00	7.674365e+00	751.7	0.95	7.36578e+00	7.95066e+00	NA	NA	NA	NA	NA	NA
pred[5]	6.287914e+00	6.285945e+00	6.279023e+00	817.9	0.95	6.14277e+00	6.42655e+00	NA	NA	NA	NA	NA	NA

	PcntGtROPE
CHAIN	NA
zbeta0	NA
zbeta[1]	NA
zbeta[2]	NA
zbeta[3]	NA
zbeta[4]	NA
zbeta[5]	NA
beta0	NA
beta[1]	NA
beta[2]	NA
beta[3]	NA
beta[4]	NA
beta[5]	NA
tau	NA
zVar	NA
pred[1]	NA
pred[2]	NA
pred[3]	NA
pred[4]	NA
pred[5]	NA

Figure 38:Summary

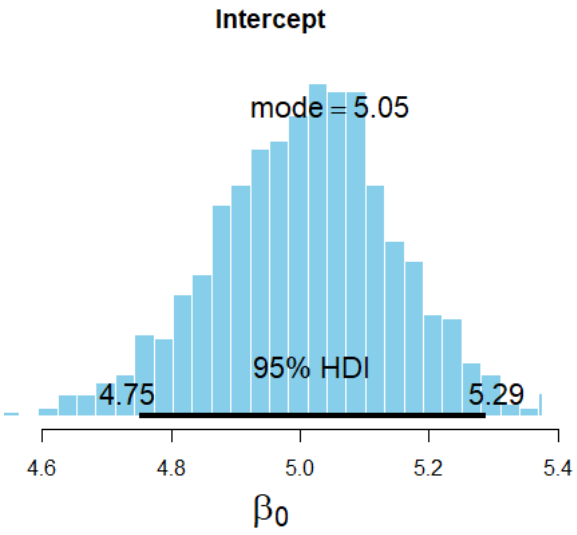

Figure 39:Posterior Distribution beta0

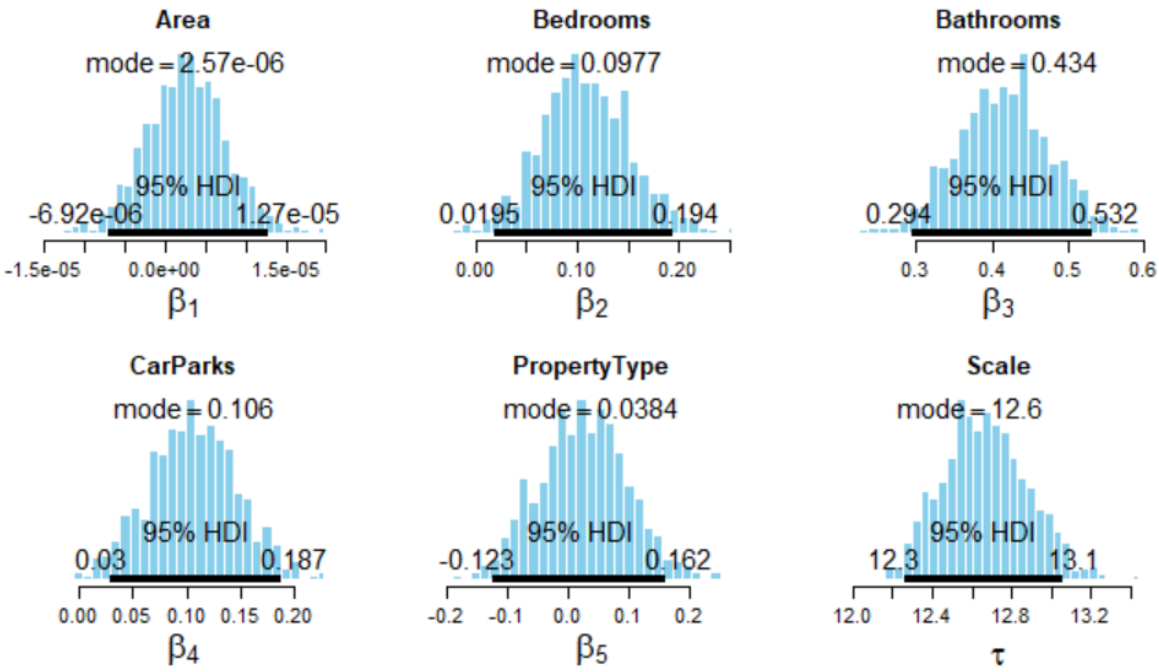

Figure 40: Posterior Distribution of coefficients

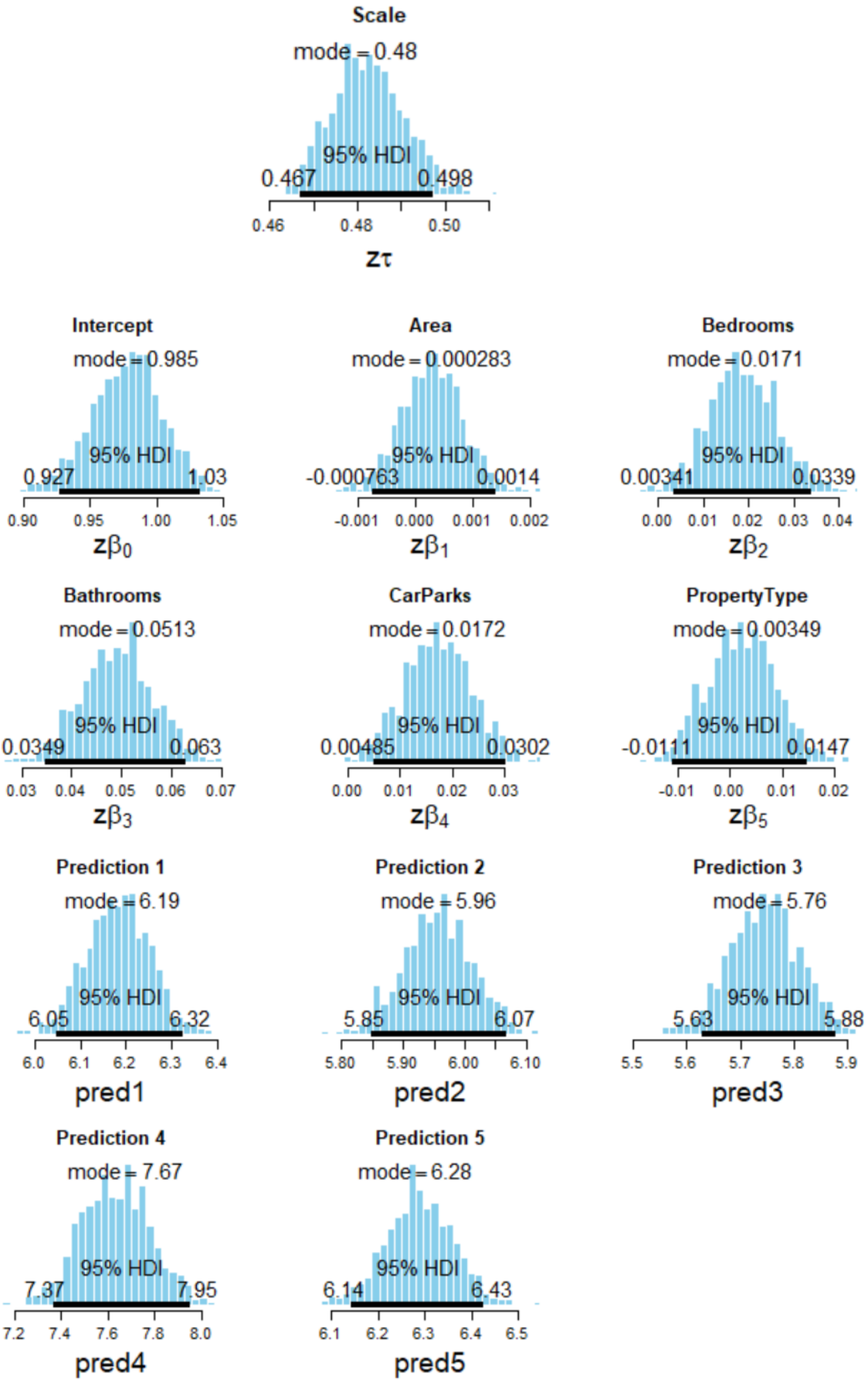

Figure 41:Posterior Distribution of coefficients and predictions

Discussion

- The interpretation of the coefficients of beta of the posterior distribution is that for **Area** for each unit increase in the area the SalePrice does not have much of a change and the Bayesian Point estimate is 2.57e-06 and interval is -6.92e-06 and 1.27e-05 but since zero is included in the 95% confidence interval it is not significant. The beta coefficient has a nearly symmetric posterior distribution.
- While the interpretation of the coefficients of beta of the posterior distribution is that for **Bedrooms** for each unit increase in the number of bedrooms there is an increase of 9770 Dollars in the SalesPrice and the Bayesian Point estimate is 0.0977 and interval is 0.0195 and 0.194 but since zero is not included in the 95% confidence interval it is significant. The beta coefficient has a nearly symmetric posterior distribution.
- While the interpretation of the coefficients of beta of the posterior distribution is that for **Bathrooms** for each unit increase in the number of bathrooms there is an increase of 43400 Dollars in the SalesPrice and the Bayesian Point estimate is 0.434 and interval is 0.294 and 0.532 but since zero is not included in the 95% confidence interval it is significant. The beta coefficient has a nearly symmetric posterior distribution.
- While the interpretation of the coefficients of beta of the posterior distribution is that for **CarParks** for each unit increase in the number of car parkings there is an increase of 10600 Dollars in the SalesPrice and the Bayesian Point estimate is 0.106 and interval is 0.03 and 0.187 but since zero is not included in the 95% confidence interval it is significant. The beta coefficient has a nearly symmetric posterior distribution.
- While the interpretation of the coefficients of beta of the posterior distribution is that for **PropertyType** if it is a Unit there is an increase of 3840 Dollars in the SalesPrice and the Bayesian Point estimate is 0.0384 and interval is -0.123 and 0.162 but since zero is included in the 95% confidence interval it is not significant. The beta coefficient has a nearly symmetric posterior distribution.

SalePrice Predictions for the 5 properties are as follows:

Tasks

- **xPred[1,] = c(600, 2, 2, 1, 1)**

For this property it can be seen in prediction 1 posterior distribution that the Bayesian Point estimate of the prediction of SalePrice of this property is 619,000 dollars and the SalePrice of this property will be between 605,000 dollars and 632,000 dollars with a probability of 0.95. The Prediction has a nearly symmetric posterior distribution.

- **xPred[2,] = c(800, 3, 1, 2, 0)**

For this property it can be seen in prediction 2 posterior distribution that the Bayesian Point estimate of the prediction of SalePrice of this property is 596,000 dollars and the SalePrice of

this property will be between 585,000 dollars and 607,000 dollars with a probability of 0.95. The Prediction has a nearly symmetric posterior distribution.

- **xPred[3,] = c(1500, 2, 1, 1, 0)**

For this property it can be seen in prediction 3 posterior distribution that the Bayesian Point estimate of the prediction of SalePrice of this property is 576,000 dollars and the SalePrice of this property will be between 563,000 dollars and 588,000 dollars with a probability of 0.95. The Prediction has a nearly symmetric posterior distribution.

- **xPred[4,] = c(2500, 5, 4, 4, 0)**

For this property it can be seen in prediction 4 posterior distribution that the Bayesian Point estimate of the prediction of SalePrice of this property is 767,000 dollars and the SalePrice of this property will be between 737,000 dollars and 795,000 dollars with a probability of 0.95. The Prediction has a nearly symmetric posterior distribution.

- **xPred[5,] = c(250, 3, 2, 1, 1)**

For this property it can be seen in prediction 5 posterior distribution that the Bayesian Point estimate of the prediction of SalePrice of this property is 628,000 dollars and the SalePrice of this property will be between 614,000 dollars and 643,000 dollars with a probability of 0.95. The Prediction has a nearly symmetric posterior distribution.

Posterior Predictive Check

Data is regenerated to perform posterior predictive check to see how the model captures the generated data to see how it performs with the raw data using the Bayesian Point estimates from the summaryInfo.

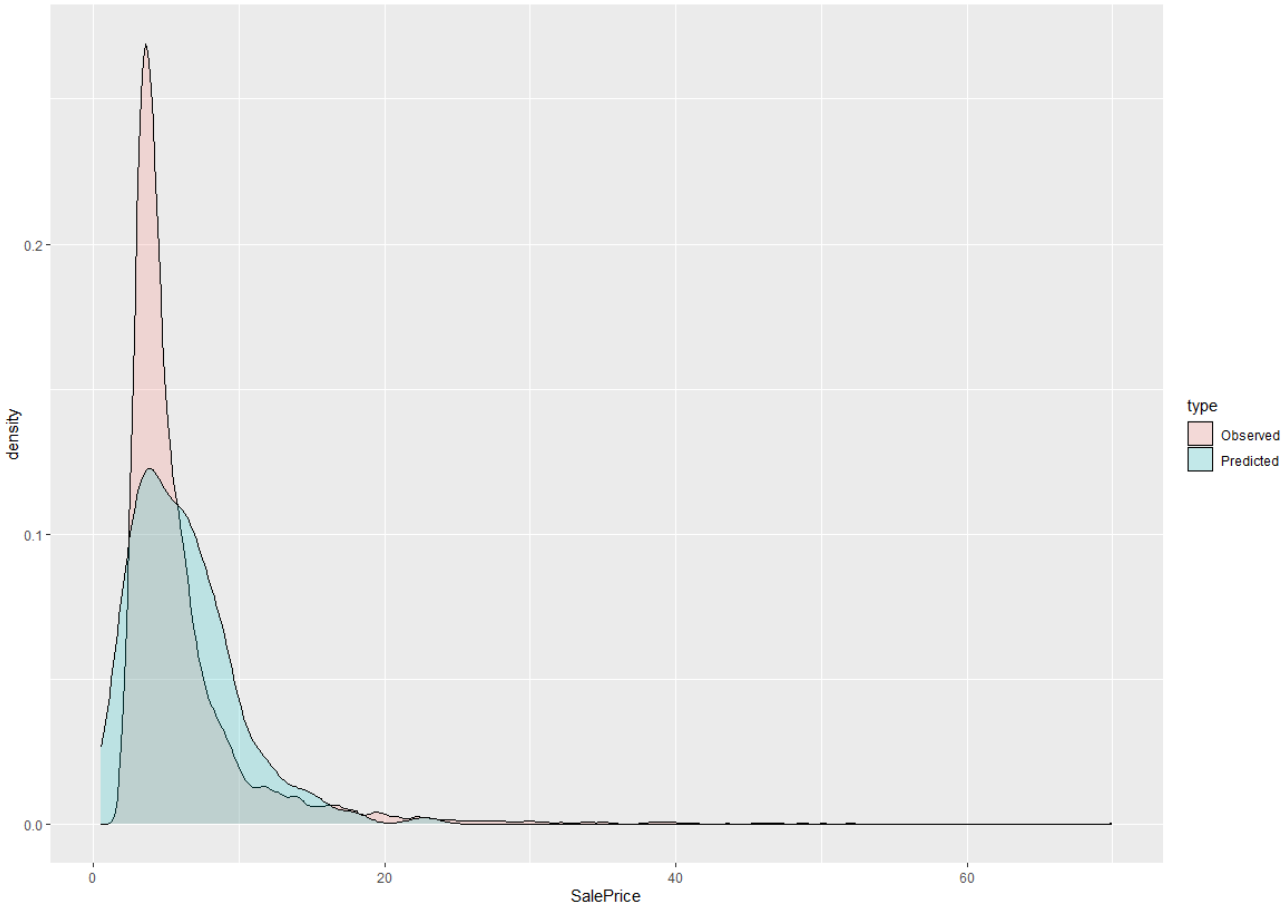

Figure 42:Posterior Predictive Check

Discussion

The model captures most of the random data but still requires improvement as there is a lot of observed data that is still not captured by the model as seen in the graph thus it will underestimate the SalePrice in some predictions as it some of the observed data is underestimated by the model as seen.

Conclusion

The JAGS model was created for the multiple linear regression setting for the normal gamma model taking into consideration the prior information. The MCMC diagnostic check showed that there was significant autocorrelation left with the initial values given and the parameters given so further thinning was performed to arrive to the final posterior distribution. The posterior distributions of the parameters were nearly symmetric, and the Bedrooms, Bathrooms and CarParks coefficient s were found to be significant, and the posterior predictive check showed that the model captures a lot of the observations and performance seemed good but still got a lot of observations that were underestimated.

References

- Demirhan, D. H. (2024a). *Module 1 - Burn-in* [Module 1 Notes, MATH2269]. RMIT University.
- Demirhan, D. H. (2024b). *Module 2 - Basics of Probability* [Module 2 Notes, MATH2269]. RMIT University.
- Demirhan, D. H. (2024c). *Module 3 - Bayes' Rule* [Module 3 Notes, MATH2269]. RMIT University.
- Demirhan, D. H. (2024d). *Module 4 - Markov Chain Monte Carlo - MCMC Methods* [Module 4 Notes, MATH2269]. RMIT University.
- Demirhan, D. H. (2024e). *Module 5 - Implementation of MCMC Methods with JAGS* [Module 5 Notes, MATH2269]. RMIT University.
- Demirhan, D. H. (2024f). *Module 6 - Bayesian Linear Regression* [Module 6 Notes, MATH2269]. RMIT University.

Appendix

```
graphics.off()

rm(list=ls())

library(ggplot2)

library(ggpubr)

library(ks)

library(rjags)

library(runjags)

library(egg)

source("DBDA2E-utilities.R")


#===============PRELIMINARY FUNCTIONS FOR POSTERIOR
INFERENCES====================


smryMCMC_HD = function(  codaSamples , compVal = NULL,  saveName=NULL) {

 summaryInfo = NULL

 mcmcMat = as.matrix(codaSamples,chains=TRUE)

 paramName = colnames(mcmcMat)

 for ( pName in paramName ) {

  if (pName %in% colnames(compVal)){

   if (!is.na(compVal[pName])) {

    summaryInfo = rbind( summaryInfo , summarizePost( paramSampleVec =
mcmcMat[,pName] ,

                                       compVal = as.numeric(compVal[pName]) ))

   }

   else {

    summaryInfo = rbind( summaryInfo , summarizePost( paramSampleVec =
mcmcMat[,pName] ) )

   }

  } else {
```

```
    summaryInfo = rbind( summaryInfo , summarizePost( paramSampleVec =
mcmcMat[,pName] ) )
   }
 }
 rownames(summaryInfo) = paramName

 # summaryInfo = rbind( summaryInfo ,
 #                 "tau" = summarizePost( mcmcMat[,"tau"] ) )
 if ( !is.null(saveName) ) {
   write.csv( summaryInfo , file=paste(saveName,"SummaryInfo.csv",sep="") )
 }
 return( summaryInfo )
}

#=============================================================================
====

plotMCMC_HD = function( codaSamples , data , xName="x" , yName="y" ,
                  showCurve=FALSE ,  pairsPlot=FALSE , compVal = NULL,
                  saveName=NULL , saveType="jpg" ) {
 # showCurve is TRUE or FALSE and indicates whether the posterior should
 #   be displayed as a histogram (by default) or by an approximate curve.
 # pairsPlot is TRUE or FALSE and indicates whether scatterplots of pairs
 #   of parameters should be displayed.
 #-----------------------------------------------------------------------
 y = data[,yName]
 x = as.matrix(data[,xName])
 mcmcMat = as.matrix(codaSamples,chains=TRUE)
 chainLength = NROW( mcmcMat )
 zbeta0 = mcmcMat[,"zbeta0"]
 zbeta  = mcmcMat[,grep("^zbeta$|^zbeta\\[",colnames(mcmcMat))]
 if ( ncol(x)==1 ) { zbeta = matrix( zbeta , ncol=1 ) }
```

```
zVar = mcmcMat[,"zVar"]
beta0 = mcmcMat[,"beta0"]
beta  = mcmcMat[,grep("^beta$|^beta\\[",colnames(mcmcMat))]
if ( ncol(x)==1 ) { beta = matrix( beta , ncol=1 ) }
tau = mcmcMat[,"tau"]
pred1 = mcmcMat[,"pred[1]"]
pred2 = mcmcMat[,"pred[2]"]
pred3 = mcmcMat[,"pred[3]"]
pred4 = mcmcMat[,"pred[4]"]
pred5 = mcmcMat[,"pred[5]"]
#------------------------------------------------------------------------------
# Compute R^2 for credible parameters:
YcorX = cor( y , x ) # correlation of y with each x predictor
Rsq = zbeta %*% matrix( YcorX , ncol=1 )
#------------------------------------------------------------------------------
if ( pairsPlot ) {
 # Plot the parameters pairwise, to see correlations:
 openGraph()
 nPtToPlot = 1000
 plotIdx = floor(seq(1,chainLength,by=chainLength/nPtToPlot))
 panel.cor = function(x, y, digits=2, prefix="", cex.cor, ...) {
  usr = par("usr"); on.exit(par(usr))
  par(usr = c(0, 1, 0, 1))
  r = (cor(x, y))
  txt = format(c(r, 0.123456789), digits=digits)[1]
  txt = paste(prefix, txt, sep="")
  if(missing(cex.cor)) cex.cor <- 0.8/strwidth(txt)
  text(0.5, 0.5, txt, cex=1.25 ) # was cex=cex.cor*r
 }
 pairs( cbind( beta0 , beta , tau )[plotIdx,] ,
      labels=c( "beta[0]" ,
```

```
            paste0("beta[",1:ncol(beta),"]\n",xName) ,
            expression(tau) ) ,
      lower.panel=panel.cor , col="skyblue" )
 if ( !is.null(saveName) ) {
  saveGraph( file=paste(saveName,"PostPairs",sep=""), type=saveType)
 }
}
#-------------------------------------------------------------------------------
# Marginal histograms:

decideOpenGraph = function( panelCount , saveName , finished=FALSE ,
                    nRow=2 , nCol=3 ) {
 # If finishing a set:
 if ( finished==TRUE ) {
  if ( !is.null(saveName) ) {
   saveGraph( file=paste0(saveName,ceiling((panelCount-1)/(nRow*nCol))),
           type=saveType)
  }
  panelCount = 1 # re-set panelCount
  return(panelCount)
 } else {
  # If this is first panel of a graph:
  if ( ( panelCount %% (nRow*nCol) ) == 1 ) {
   # If previous graph was open, save previous one:
   if ( panelCount>1 & !is.null(saveName) ) {
    saveGraph( file=paste0(saveName,(panelCount%/%(nRow*nCol))),
            type=saveType)
   }
   # Open new graph
   openGraph(width=nCol*7.0/3,height=nRow*2.0)
   layout( matrix( 1:(nRow*nCol) , nrow=nRow, byrow=TRUE ) )
```

```
    par( mar=c(4,4,2.5,0.5) , mgp=c(2.5,0.7,0) )
  }
  # Increment and return panel count:
  panelCount = panelCount+1
  return(panelCount)
 }
}

# Original scale:
panelCount = 1
if (!is.na(compVal["beta0"])){
 panelCount = decideOpenGraph( panelCount , saveName=paste0(saveName,"PostMarg") )
 histInfo = plotPost( beta0 , cex.lab = 1.75 , showCurve=showCurve ,
                xlab=bquote(beta[0]) , main="Intercept", compVal = as.numeric(compVal["beta0"] ))
} else {
 histInfo = plotPost( beta0 , cex.lab = 1.75 , showCurve=showCurve ,
                xlab=bquote(beta[0]) , main="Intercept")
}
for ( bIdx in 1:ncol(beta) ) {
 panelCount = decideOpenGraph( panelCount , saveName=paste0(saveName,"PostMarg") )
 if (!is.na(compVal[paste0("beta[",bIdx,"]")])) {
  histInfo = plotPost( beta[,bIdx] , cex.lab = 1.75 , showCurve=showCurve ,
                xlab=bquote(beta[.(bIdx)]) , main=xName[bIdx],
                compVal = as.numeric(compVal[paste0("beta[",bIdx,"]")]))
 } else{
  histInfo = plotPost( beta[,bIdx] , cex.lab = 1.75 , showCurve=showCurve ,
                xlab=bquote(beta[.(bIdx)]) , main=xName[bIdx])
 }
}
panelCount = decideOpenGraph( panelCount , saveName=paste0(saveName,"PostMarg") )
histInfo = plotPost( tau , cex.lab = 1.75 , showCurve=showCurve ,
```

```
              xlab=bquote(tau) , main=paste("Scale") )
# panelCount = decideOpenGraph( panelCount , saveName=paste0(saveName,"PostMarg") )
# histInfo = plotPost( Rsq , cex.lab = 1.75 , showCurve=showCurve ,
#                xlab=bquote(R^2) , main=paste("Prop Var Accntd") )
panelCount = decideOpenGraph( panelCount ,  saveName=paste0(saveName,"PostMarg") )
histInfo = plotPost( pred1 , cex.lab = 1.75 , showCurve=showCurve ,
              xlab="pred1" , main="Prediction 1" )
panelCount = decideOpenGraph( panelCount ,  saveName=paste0(saveName,"PostMarg") )
histInfo = plotPost( pred2 , cex.lab = 1.75 , showCurve=showCurve ,
              xlab="pred2" , main="Prediction 2" )
panelCount = decideOpenGraph( panelCount ,  saveName=paste0(saveName,"PostMarg") )
histInfo = plotPost( pred3 , cex.lab = 1.75 , showCurve=showCurve ,
              xlab="pred3" , main="Prediction 3" )
panelCount = decideOpenGraph( panelCount ,  saveName=paste0(saveName,"PostMarg") )
histInfo = plotPost( pred4 , cex.lab = 1.75 , showCurve=showCurve ,
              xlab="pred4" , main="Prediction 4" )
panelCount = decideOpenGraph( panelCount ,  saveName=paste0(saveName,"PostMarg") )
histInfo = plotPost( pred5 , cex.lab = 1.75 , showCurve=showCurve ,
              xlab="pred5" , main="Prediction 5" )

# Standardized scale:
panelCount = 1
panelCount = decideOpenGraph( panelCount , saveName=paste0(saveName,"PostMargZ") )
histInfo = plotPost( zbeta0 , cex.lab = 1.75 , showCurve=showCurve ,
              xlab=bquote(z*beta[0]) , main="Intercept" )
for ( bIdx in 1:ncol(beta) ) {
  panelCount = decideOpenGraph( panelCount , saveName=paste0(saveName,"PostMargZ") )
  histInfo = plotPost( zbeta[,bIdx] , cex.lab = 1.75 , showCurve=showCurve ,
                xlab=bquote(z*beta[.(bIdx)]) , main=xName[bIdx] )
}
panelCount = decideOpenGraph( panelCount , saveName=paste0(saveName,"PostMargZ") )
```

```
 histInfo = plotPost( zVar , cex.lab = 1.75 , showCurve=showCurve ,
                xlab=bquote(z*tau) , main=paste("Scale") )
 # panelCount = decideOpenGraph( panelCount , saveName=paste0(saveName,"PostMargZ") )
 # histInfo = plotPost( Rsq , cex.lab = 1.75 , showCurve=showCurve ,
 #                xlab=bquote(R^2) , main=paste("Prop Var Accntd") )
 panelCount = decideOpenGraph( panelCount , finished=TRUE ,
saveName=paste0(saveName,"PostMargZ") )

 #-------------------------------------------------------------------------
}

#===============PRELIMINARY FUNCTIONS FOR POSTERIOR
INFERENCES====================


propPricesAus <- read.csv("Assignment2PropertyPrices.csv")
head(propPricesAus)



# Scatter plots
p1 <- ggplot(propPricesAus, aes(x=Area, y=SalePrice.100K.)) +
 geom_point() +
 xlab("Area") +
 ylab("Sale Price(100K)")

p2 <- ggplot(propPricesAus, aes(x=Bedrooms, y=SalePrice.100K.)) +
 geom_point() +
 xlab("Bedrooms") +
 ylab("Sale Price(100K)")
```

```
p3 <- ggplot(propPricesAus, aes(x=Bathrooms, y=SalePrice.100K.)) +
 geom_point() +
 xlab("Bathrooms") +
 ylab("Sale Price(100K)")


p4 <- ggplot(propPricesAus, aes(x=CarParks, y=SalePrice.100K.)) +
 geom_point() +
 xlab("CarParks") +
 ylab("Sale Price(100K)")

p5 <- ggplot(propPricesAus, aes(x=PropertyType, y=SalePrice.100K.)) +
 geom_point() +
 xlab("PropertyType") +
 ylab("Sale Price(100K)")

figure <- ggarrange(p1, p2, p3, p4,p5, nrow = 3, ncol = 2)
figure

# Histogram
hist(propPricesAus$SalePrice.100K., main= " Histogram of the dependent variable", xlab = "Sale Price(100K)")

# Kernel density estimation
plot(kde(propPricesAus$SalePrice.100K.), xlab = "Sale Price(100K)") # with default settings


# Data
y = propPricesAus[,"SalePrice.100K."]
x = as.matrix(propPricesAus[,c("Area","Bedrooms","Bathrooms","CarParks","PropertyType")])
```

```
summary(propPricesAus)



cat("\nCORRELATION MATRIX OF PREDICTORS:\n ")
show( round(cor(x),3) )
cat("\n")

xPred = array(NA, dim = c(5,5))
xPred[1,] = c(600, 2, 2, 1, 1)
xPred[2,] = c(800, 3, 1, 2, 0)
xPred[3,] = c(1500, 2, 1, 1, 0)
xPred[4,] = c(2500, 5, 4, 4, 0)
xPred[5,] = c(250, 3, 2, 1, 1)

# Specifying the data in a list, for usage in JAGS:
dataList <- list(
 x = x ,
 y = y ,
 xPred = xPred ,
 Nx = dim(x)[2] ,
 Ntotal = dim(x)[1]
)

#Initial List
initsList <- list(
 zbeta0 = 6.25,
 zbeta = c(1.5, 1.3, 1.4, 1.2,1),
 Var = 26.3
)
```

```
# THE MODEL.
modelString = "
# Standardize the data:
data {
 ysd <- sd(y)
 for ( i in 1:Ntotal ) {
  zy[i] <- y[i] / ysd
 }
 for ( j in 1:Nx ) {
  xsd[j] <-  sd(x[,j])
  for ( i in 1:Ntotal ) {
   zx[i,j] <- x[i,j] / xsd[j]
  }
 }
}
# Specify the model for scaled data:
model {
 for ( i in 1:Ntotal ) {
  zy[i] ~ dgamma( (mu[i]^2)/zVar , mu[i]/zVar )
  mu[i] <- zbeta0 + sum( zbeta[1:Nx] * zx[i,1:Nx] )
 }
 # Priors on standardized scale:
 zbeta0 ~ dnorm( 0 , 1/2^2 )
 zbeta[1] ~ dnorm( (90/100000)/xsd[1] , 1/(0.1/xsd[1]^2) )
 zbeta[2] ~ dnorm( 1/xsd[2] , 1/(4/xsd[2]^2) )
 zbeta[3] ~ dnorm( 0 , 1/4 )
 zbeta[4] ~ dnorm( 1.2/xsd[4] , 1/(1/xsd[4]^2) )
 zbeta[5] ~ dnorm( (-1.5)/xsd[5] , 1/(0.1/xsd[5]^2) )

 zVar ~ dgamma( 0.01 , 0.01 )
```

```
 # Transform to original scale:
 beta[1:Nx] <- ( zbeta[1:Nx] / xsd[1:Nx] ) * ysd
 beta0 <- zbeta0*ysd
 tau <- zVar * (ysd)^2

 # Compute predictions at every step of the MCMC
 for ( i in 1:5){
   pred[i] <- beta0 + beta[1] * xPred[i,1] + beta[2] * xPred[i,2] + beta[3] * xPred[i,3] + beta[4] *
xPred[i,4]+ beta[5] * xPred[i,5]
 }



}
"
# Writing out modelString to a text file
writeLines( modelString , con="TEMPmodel.txt" )



parameters = c( "zbeta0" ,  "zbeta" , "beta0" ,  "beta" ,  "tau", "zVar")

adaptSteps = 500
burnInSteps = 10000
nChains = 2
thinSteps = 3
numSavedSteps = 5000
nIter = ceiling( ( numSavedSteps * thinSteps ) / nChains )

# Parallel run
startTime = proc.time()
runJagsOut <- run.jags( method="parallel" ,
                    model="TEMPmodel.txt" ,
```

```
                monitor=c( "zbeta0" ,  "zbeta" , "beta0" ,  "beta" ,  "tau", "zVar", "pred")  ,
                data=dataList ,
                inits=initsList ,
                n.chains=nChains ,
                adapt=adaptSteps ,
                burnin=burnInSteps ,
                sample=numSavedSteps ,
                thin=thinSteps , summarise=FALSE , plots=FALSE )
codaSamples = as.mcmc.list( runJagsOut )
stopTime = proc.time()
elapsedTime = stopTime - startTime
show(elapsedTime)

save.image(file="absassignment2rprog1-prefinal-219241final.RData")
load(file="absassignment2rprog1-prefinal-219241final.RData")
nrow(codaSamples[[1]])

# Further thinning from the codaSamples

furtherThin <- 11
thiningSequence <- seq(1,nrow(codaSamples[[1]]), furtherThin)
newCodaSamples <- mcmc.list()
for ( i in 1:nChains){
  newCodaSamples[[i]] <- as.mcmc(codaSamples[[i]][thiningSequence,])
}

summary(codaSamples)

summary(newCodaSamples)
```

```
library(coda)
gelman.plot(codaSamples, confidence = 0.95, transform=FALSE, autoburnin=FALSE)

diagMCMC( codaSamples , parName="beta0" )
diagMCMC( codaSamples , parName="beta[1]" )
diagMCMC( codaSamples , parName="beta[2]" )
diagMCMC( codaSamples , parName="beta[3]" )
diagMCMC( codaSamples , parName="beta[4]" )
diagMCMC( codaSamples , parName="beta[5]" )
diagMCMC( codaSamples , parName="tau" )
diagMCMC( codaSamples , parName="pred[1]" )
diagMCMC( codaSamples , parName="pred[2]" )
diagMCMC( codaSamples , parName="pred[3]" )
diagMCMC( codaSamples , parName="pred[4]" )
diagMCMC( codaSamples , parName="pred[5]" )
diagMCMC( codaSamples , parName="zbeta0" )
diagMCMC( codaSamples , parName="zbeta[1]" )
diagMCMC( codaSamples , parName="zbeta[2]" )
diagMCMC( codaSamples , parName="zbeta[3]" )
diagMCMC( codaSamples , parName="zbeta[4]" )
diagMCMC( codaSamples , parName="zbeta[5]" )

#MCMC after further thinning
gelman.plot(newCodaSamples, confidence = 0.95, transform=FALSE, autoburnin=FALSE)

diagMCMC( newCodaSamples , parName="beta0" )
diagMCMC( newCodaSamples , parName="beta[1]" )
diagMCMC( newCodaSamples , parName="beta[2]" )
diagMCMC( newCodaSamples , parName="beta[3]" )
diagMCMC( newCodaSamples , parName="beta[4]" )
diagMCMC( newCodaSamples , parName="beta[5]" )
```

```
diagMCMC( newCodaSamples , parName="tau" )
diagMCMC( newCodaSamples , parName="pred[1]" )
diagMCMC( newCodaSamples , parName="pred[2]" )
diagMCMC( newCodaSamples , parName="pred[3]" )
diagMCMC( newCodaSamples , parName="pred[4]" )
diagMCMC( newCodaSamples , parName="pred[5]" )
diagMCMC( newCodaSamples , parName="zbeta0" )
diagMCMC( newCodaSamples , parName="zbeta[1]" )
diagMCMC( newCodaSamples , parName="zbeta[2]" )
diagMCMC( newCodaSamples , parName="zbeta[3]" )
diagMCMC( newCodaSamples , parName="zbeta[4]" )
diagMCMC( newCodaSamples , parName="zbeta[5]" )
graphics.off()

compVal <- data.frame("beta0" = NA, "beta[1]" = NA, "beta[2]" = NA, "beta[3]" = NA, "beta[4]" =
NA,"beta[5]" =  NA, "tau" = NA , check.names=FALSE)

summaryInfo <- smryMCMC_HD( codaSamples = newCodaSamples , compVal = compVal )
print(summaryInfo)

plotMCMC_HD( codaSamples = newCodaSamples , data = propPricesAus,
xName=c("Area","Bedrooms","Bathrooms","CarParks","PropertyType") ,
        yName="SalePrice.100K.", compVal = compVal)

# ============ Predictive check ============
coefficients <- summaryInfo[8:13,3]
Variance <- summaryInfo[14,3]

meanGamma <- as.matrix(cbind(rep(1,nrow(x)),  x)) %*% as.vector(coefficients)
# Generating random data from the posterior distribution.
randomData <- rgamma(n= 333,shape=meanGamma^2/Variance, rate =
meanGamma/Variance)
```

```
# Displaying the density plot of observed data and posterior distribution:
predicted <- data.frame(SalePrice = randomData)
observed <- data.frame(SalePrice = y)
predicted$type <- "Predicted"
observed$type <- "Observed"
dataPred <- rbind(predicted, observed)

ggplot(dataPred, aes(SalePrice, fill = type)) + geom_density(alpha = 0.2)
```