Analysis Report on NVDI Data

Harika Malapaka **Unity Id : hsmalapa**

1. Statistical Model :

Normalized Difference Vegetation Index (NDVI) is a common measure for determining greenness. This measurements will be useful in fields like agriculture. Now, we use Bayesian statistics and develop a model to predict NVDI values for 365 days which was initially observed from 3 satellites Y1, Y2, Y3. The representation of the Data frames/ data structures is followed as given in the initial dataset E2.Rdata. Here, we combine all the Y's into a single response matrix Y, so that we can model easily. Since, there are many missing values in Y1 (80%), we use OpenBugs rather than Jags to model this NVDI data. To account for unknown parameters, we use Inverse Wishart matrix

The mathematical model can be described as follows :

1. Data Layer : Provide the likelihood

The $\theta1 \sim$ Normal($\mu1$, $\Sigma1$) and θt $|\theta t-1 \sim$ Normal($\mu2 + \rho\theta t-1$, $\Sigma2$)

Y1[i,1:p] ~ dmnorm(mpx[i-1,1:p],Omega2[1:p,1:p])

Y2[i] ~ dnorm(mpxy[i-1],tau)

Y3[i] ~ dnorm(mpxz[i-1],tau)

2. Process Layer : Provide the likelihood priors

Mpx -> mu2[i,j] + rho * Y1[i,j]] // i is in range 1 to 365 and j is in range 1 to 6

Mpxy -> mu2[i,1] + rho*Y2[i]] // i is in range 1 to 365

Mpxz -> mean(mu2[i,1:6]) + rho*Y3[i] // i is in range 1 to 365

3. Prior Layer : Provide the initial and missing value priors

Rho ~ dunif(0,1)

Tau ~ dgamma(0.1,0.1)

Omega 1 [1:6,1:6] ~ dwish(R1[,],k) // R1 and k are defined

Omega 2 [1:6,1:6] ~ dwish(R2[,],k)

Mu1[1:6] ~ dnorm(0,0.001)

Mu2 [1:365,1:6] ~ dnorm(0,0.001)

2.OpenBugs Code :

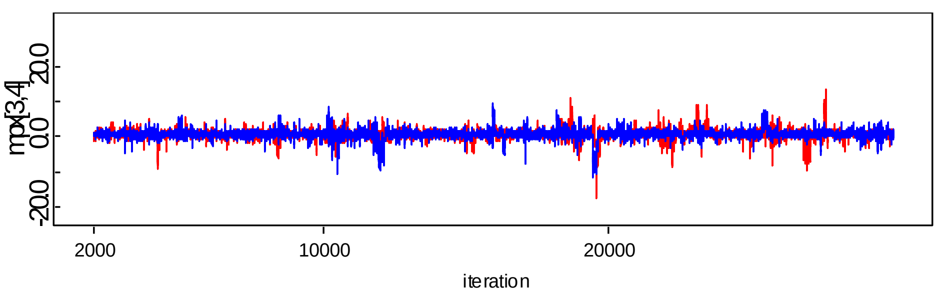
The code for model statement is given below.

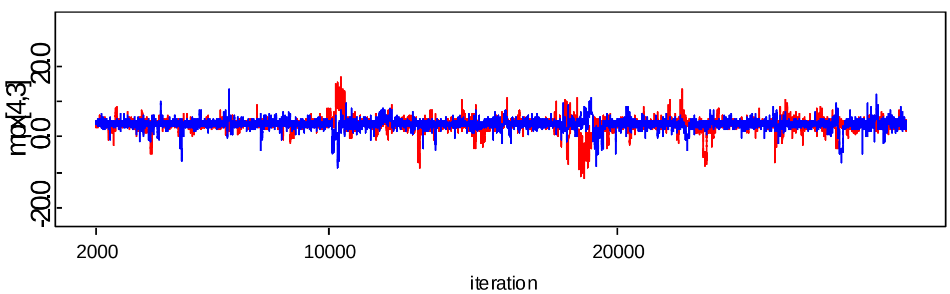
Since, it's not the regular JAGS code, everything is commented clearly

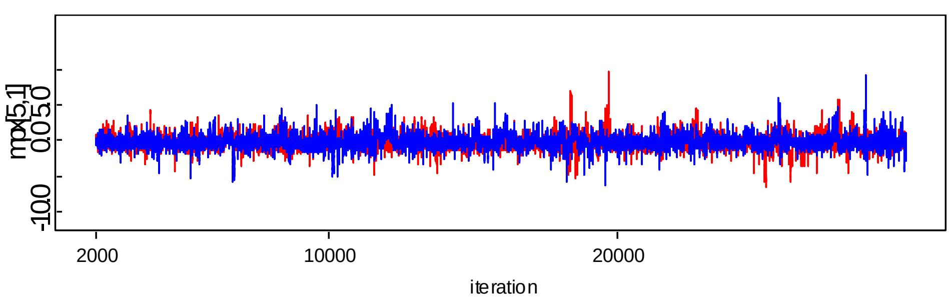
```
RE_model <- function(){
 # the liklihood of Y1 (satellite) for day 1 for all 6 pixels
 Y1[1,1:p] ~ dmnorm(mu1[1:p],Omega1[1:p,1:p])
 # liklihood for Y2 ( satellite 2)
 Y2[1] ~ dnorm(mu1[1],tau)
 # liklihood of Y3(satellite 3)
 #Y3[1] ~ dnorm(m3,tau)
 # The third satellite's prior which is given as per question - the mean of 6 pixels
# m3 <- mean(mu1[1:p])

 for(i in 2:n){
  # compute liklihood for the remaining 364 days for all 3 satellites
  Y1[i,1:p] ~ dmnorm(mpx[i-1,1:p],Omega2[1:p,1:p])
  Y2[i] ~ dnorm(mpxy[i-1],tau)
  #Y3[i] ~ dnorm(mpxz[i-1],tau)
 }

 # this is process layer which defines the function used in data layer
 for(i in 1:n-1){
  for(j in 1:p){
   # for Y1, the priors for liklihood
   mpx[i,j] <- mu2[i,j] + rho*Y1[i,j]
  }
  mpxy[i] <- mu2[i,1] + rho*Y2[i]
  #mpxz[i] <- mean(mu2[i,1:6]) + rho*Y3[i]
 }

 for(j in 1:p){
  # mean vector for alues of day 1
  mu1[j] ~ dnorm(0,0.01)
 }

 # priors for missing data model
 for(i in 1:n-1){
  # mean vector for values from day 2 onwards
  for(j in 1:p){
   mu2[i,j] ~ dnorm(0,0.01)
  }
 }
```

```
# priors for missing data model parameters
  # rho controls the temperature and belongs to (0,1)
  rho ~ dunif(0,1)
  tau ~ dgamma(0.01,0.01)
  # Since we do not use omega1/omega2 in futurw, we need not calculate it's inverse
  # since Bugs considers inverse of the variance in dnorm function
  # the second parameter for wishart function is 0.1 greater than no of rows in covariate matrix.
  k <- p+0.1
  # omega 1 and omega 2 are covariance matrices
  Omega1[1:p,1:p] ~ dwish(R1[,],k)
  for(j1 in 1:p){for(j2 in 1:p){R1[j1,j2]<-0.1*equals(j1,j2)}} #R is diagonal
  Omega2[1:p,1:p] ~ dwish(R2[,],k)
  for(j1 in 1:p){for(j2 in 1:p){R2[j1,j2]<-0.1*equals(j1,j2)}} #R is diagonal
  }
```

Few of the trace ploys have been given below.
They all show that it’s converging well.







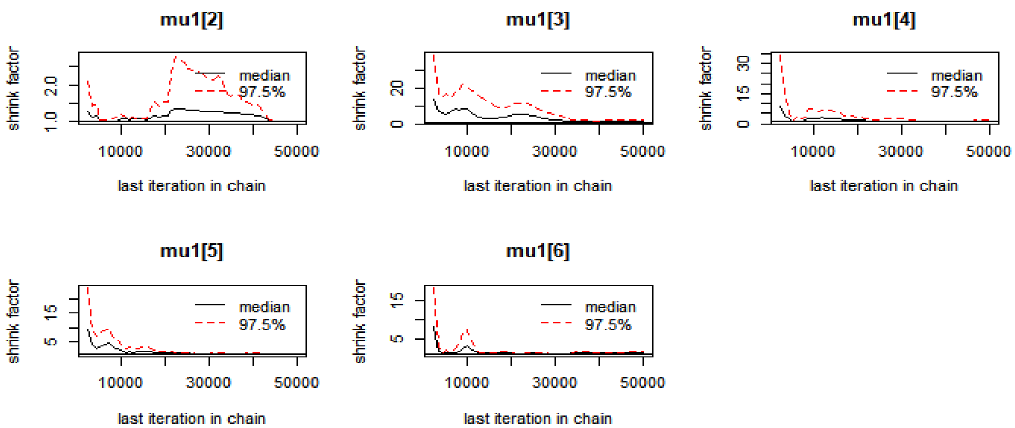
3. Convergence :

To make sure whether the values converged or not, gelman dialogue was used.

If the upper confidence interval is near to one, we can say that the value has converged.

So when we checked for 365*6 = 2190 plots, we found from gelman dialogue that 75% have converged by counting the number of values near to 1 in Upper confidence interval.

mu1[1:6] shows the gelman convergence plots of Y1 for day 1 – all 6 pixels



We observe tha, if Y1[i,j] from the response vector given to us, is not a null value, then generally it would converge at some point. The probability that it converges is high compared to when it’s value is null (NA).

4. Final Results :

We have 3 response matrices : Y1, Y2 and Y3. Using appropriate priors as stated above, we get the minimum difference when we compare θ_{tj} with the observed data. Since we have 365/6 matrix for Y1, we compute the θ_{tj} using values of even Y2 and Y3 apart from Y1 and then compare the values that our model has generated with observed values.

One of the observation is that if I do not include Y2 when developing a model, we get results which are not that convincing. We may conclude that Y2 is biased. The reason might be because in Y1, 80% of the times, the values are missing. So when we include Y2, since it's missing just 10% of the times, we get a better informative model which predicts better.

When we do not include Y3, the results of Y1 estimate are almost same as when we include it and thus good. But when we check again Y2 data, they aren't that convincing. This maybe because, Y3 wouldn't carry much of extra information for Y1, since it may be a average of 6 pixels from first satellite. But there is effect on Y2 prediction since it lacks information when we do not include Y3.

General points :

To see a convergence dialogue, I've used gelman dialogue. For that reason, I took 2 chains for the model. Regarding selecting prior values, I didn't differentiate between 6 pixel values of satellite 1. Anyhow there wasn't much difference even if I had used different mean and variance for the dnorm distribution of mean vectors.