ST540 Take Home Exam

Jeremy Ash

April 8, 2018

Statistical Model

I used a Bayesian hierarchical model with three components: the prior layer, process layer and data layer. The data layer consisted of the likelihood of the data from the three satellites. I could assume that all data were independent given θ, so I modeled the likelihood of each data set separately. Because the $Y1$ data is unbiased, but potentially noisy the model was:

$$Y^1_{i,j} = \theta_{i,j} + \epsilon_{i,j}, \quad \epsilon_{i,j} \sim N(0, \tau_1^2), \quad \epsilon_{i,j} \sim N(0, \tau_1^2), \quad i = 1, ..., 365, \quad j = 1, ..., 6$$

The $Y2$ data was a potentially biased and noisy measurement of pixel 1, so the model was:

$$Y^2_i = \theta_{i,1} + \beta_2 + \epsilon_i, \quad \epsilon_i \sim N(0, \tau_2^2), \quad \beta_2 \sim N(0, 100), \quad i = 1, ..., 365$$

No information about the bias was provided so a highly uninformative prior for the bias was assumed. The $Y3$ data was a potentially biased and noisy measurement of the average of all 6 pixels for each day, so the model was:

$$Y^3_i = \frac{\sum_{j=1}^{6} \theta_{i,j}}{6} + \beta_3 + \epsilon_i, \quad \epsilon_i \sim N(0, \tau_3^2), \quad \beta_3 \sim N(0, 100), \quad i = 1, ..., 365$$

In the process layer, the latent θ variables that generated these data are modeled, I simply used the information that was provided in the exam on how the latent variables were distributed:

$$\theta_1 \sim Normal(\mu_1, \Sigma_1), \quad \theta_t | \theta_{t-1} \sim Normal(\mu_2 + \rho\theta_{t-1}, \Sigma_2)$$

I assumed an uninformative Whishart prior for the covariance matrices, and an uninformative beta prior for ρ. I assumed that the elements of μ_1 and μ_2 were iid $Normal(0, 100)$:

$$\Sigma_1, \Sigma_2 \sim InvWishart(6.01, .1I_{6X6}), \quad \rho \sim Beta(1, 1)$$

Where I_{6X6} is a 6X6 indentity matrix.
I assumed that the elements of μ_1 and μ_2 were iid $Normal(0, 100)$.

JAGS also will generate all of the missing data in $Y1, Y2$, and $Y3$ from the posterior as additional model parameters.

JAGS Code

Sorry, for the sake of readability of this code, I distributed it across two pages.

```
setwd("C:/Users/jrash/ubuntu_share/Google Drive/courses/grad_classes/ST540_Bayes/take_home_exam/")
load("E2.RData")

all_y_dat <- list(Y1=Y1, Y2 = Y2, Y3 = Y3, p = 6, N = 365)

RE_model <- "model{

  ## Prior Layer

  # Construct R matrix for the wishart prior distributions
  k<-p+0.1

  for(j1 in 1:p){
    for(j2 in 1:p){
      R[j1,j2]<-0.1*(j1==j2)
    }
  }

  # Hyperparameters for theta 1
  for (i in 1:6) {
    mu1[i] ~ dnorm(0, .01)
  }

  s1_prec[1:p,1:p]~dwish(R[,],k)
  s1_cov[1:p,1:p]<-inverse(s1_prec[,])

  # Hyperparameters for theta 2-365
  for (i in 1:6) {
    mu2[i] ~ dnorm(0, .01)
  }

  rho ~ dbeta(1,1)

  s2_prec[1:p,1:p]~dwish(R[,],k)
  s2_cov[1:p,1:p]<-inverse(s2_prec[,])

  ## Process Layer

  # Theta 1 generates the data on day 1
  theta[1, 1:p]~dmnorm(mu1, s1_prec[,])

  # Theta 2-365 generates the data on day 2-365
  for(i in 2:365) {
    theta[i, 1:p]~dmnorm(mu2 + rho*theta[i-1, 1:p], s2_prec[,])
  }

  # Error term which models the noise in the Y data
  tau1  ~ dgamma(0.1,0.1)
  tau2  ~ dgamma(0.1,0.1)
  tau3  ~ dgamma(0.1,0.1)
```

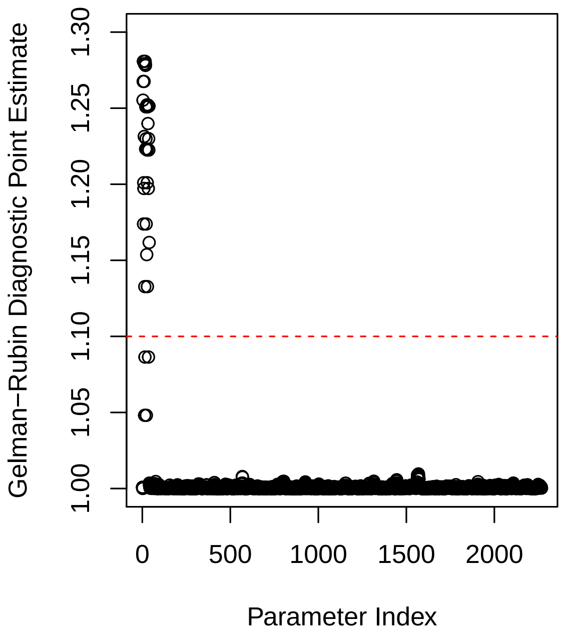
```r
  #Bias terms for Y2 and Y3
  delta2 ~ dnorm(0, .01)
  delta3 ~ dnorm(0, .01)

  ## Data layer

  # Compute the likelihood of Y1 data if it is not missing
  # If it is missing, then sample from the posterior distribution
  for(i in 1:N){
    for(j in 1:p){
      Y1[i, j] ~ dnorm(theta[i, j], tau1)
    }
  }
  # Compute the likelihood of Y2 data if it is not missing
  for(i in 1:N){
    Y2[i] ~ dnorm(theta[i, 1] + delta2, tau2)
  }
  # Compute the likelihood of Y3 data if it is not missing
  for(i in 1:N){
    Y3[i] ~ dnorm(mean(theta[i, ]) + delta3, tau3)
  }
}"

# Spreading the computation across four cores
cl <- makeCluster(4)
registerDoParallel(cl)

model1 <- jags.model(textConnection(RE_model),data = all_y_dat, n.chains=3)

# 40000 burn-in samples
update(model1, 20000, progress.bar="none")

# 200000 samples from the posterior
samp   <- coda.samples(model1,
                       variable.names=c("theta", "rho", "tau1", "tau2", "tau3",
                                        "delta2", "delta3", "s1_cov", "s2_cov"),
                       n.iter=40000, thin=10, progress.bar="none")
```

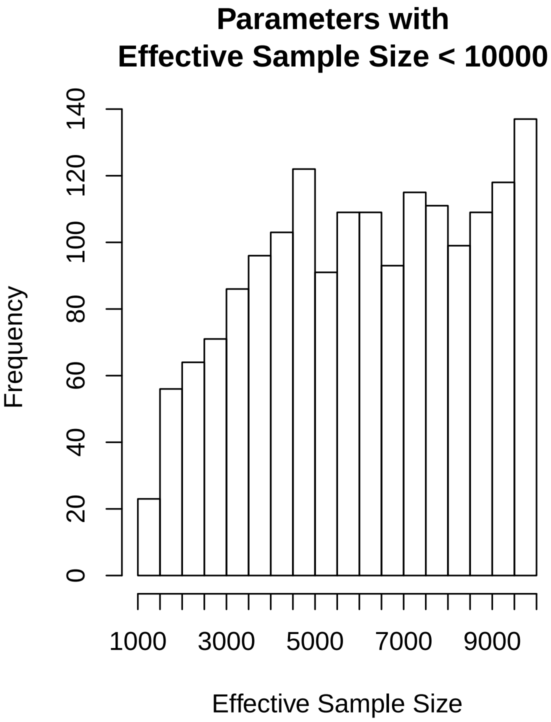
Convergence

Only model parameters of interest were monitored and used for convergence diagnostics. The convergence of the missing observations was inconsequential, as these “parameters” were not used to compute the likelihood of the observed data.

To assess convergence, I ran three MCMC chains from different initial values. I allowed JAGS to select the initial values, though I may have gotten faster convergence if I used better estimates for the initial values. I then used the Gelman Rubin statistic to determine if the chains were similar enough to conclude there is convergence. For nearly all parameters in the model, the estimate of the Gelman Rubin statistic is below 1.1. The only exception were a few parameters that were just above 1.1. These were all of the elements of the θ_1 covariance matrix. However, these parameters were below 1.3, so I concluded there was still decent convergence. I made a change to my model at the last moment that led to improvments in the prediction performance, but I did not have enough time to run the chains to convergence. I am confident that if I were to run the chains for longer, I would acheive convergence in all of the model parameters, as this was the case before I made a change to the model I concluded that for the parameters of interest, the MCMC chains had converged to the posterior distribution of the model parameters.

To determine whether the chains had been run for long enough after convergence to estimate the posterior distribution of the parameters, the effective sample size of the parameters was computed. The effective sample size was at least a thousand for all parameters. In the following histogram, only the parameters with effective sample size below 10000 is shown to make the histogram more readable. Overall, I concluded that for the parameters of interest, my chains had converged, and that they had been run long enough to accurately estimate the posterior distributions of the model parameters.

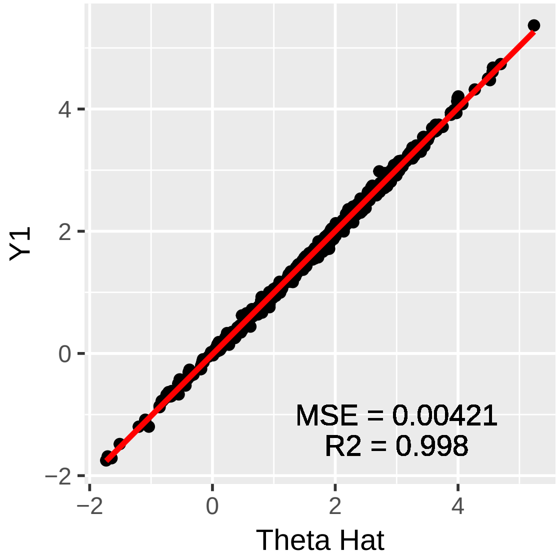


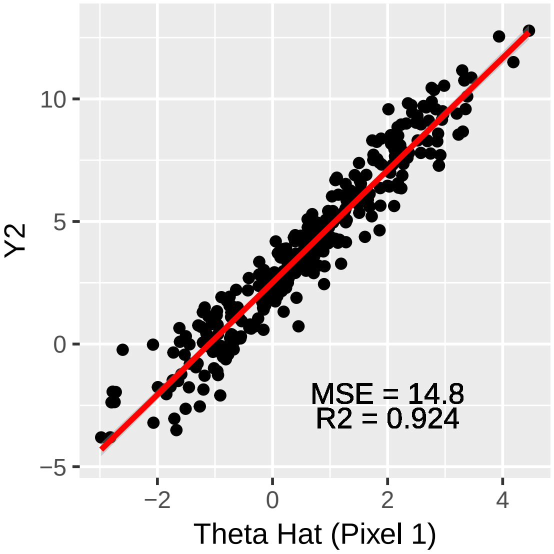


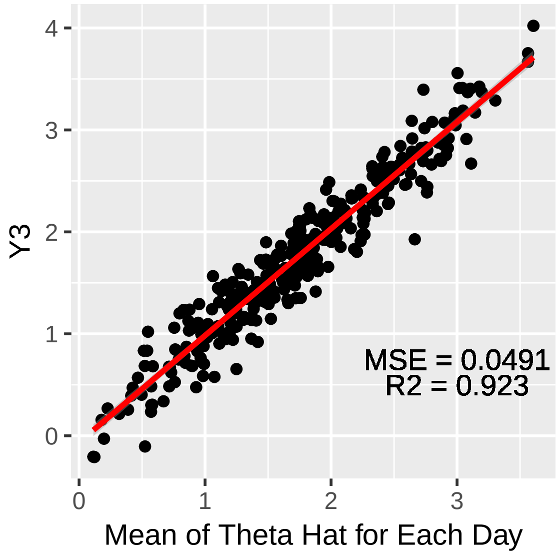
Final Results

Since $Y1$ is the "gold standard" data with the least amount of noise and bias, this data is ideal for estimating my prediction accuracy of the true θ. To further support that this data is indeed the gold standard. The posterior mean of the error variance for Y1 was 0.026.

For each dataset, I removed all NA's and computed the posterior means of the corresponding functions of $\hat{\theta}$ that were being measured. I used the posterior means of θ because this is what I will use for my final estimate of θ.







The following plot shows that there is a very high correlation between $\hat{\theta}$ and $Y1$. There is also an extremely low MSE (0.049). There seems to be very constant the error variance seems to be constant across theta hat and is small, suggesting that the error that is observed may be explained by the noise in $Y1$. The y intercept is almost 0 supporting that this data is unbiased.

The same plot was created for the $Y2$ data. For these data, there is a high correlation as expected, but much larger MSE. However, we expect the error variance to be larger for this noisier data, and the error variance is constant, which is consisten with the way these data were generated. The y intercept is much larger than 0, supporting the bias in these data.

The $Y3$ data appears to have very little bias. The error variance is also substantially lower than what was observed in the $Y2$ data. This is expected because this data is average of 6 data points, which will reduce the error variance.