HW 5

Key

Due Friday October 4, 2019

Please use D2L to turn in both the PDF output and your R Markdown file.

Q1.

After enjoying numerous Bridger Bowl powder days during your time at MSU, you have received a job offer in New Mexico. Your decision hinges on the quality of snow at your potential new home mountain, Taos.

According to https://www.onthesnow.com/new-mexico/taos-ski-valley/historical-snowfall.html annual snowfall totals can be obtained.

Below are the snowfall totals for Taos and Bridger Bowl for the last nine years.

```
library(dplyr)
library(ggplot2)
library(LearnBayes) # for rigamma
library(gridExtra)

taos <- c(78, 192, 169, 179, 191, 204, 197, 116, 195)
bridger <- c(271, 209, 228, 166, 316, 254, 344, 319, 247)
```

Specifically the BSF is interested in computing three probabilistic statements.

1. $Pr[\theta_{bridger} > 250]$
2. $Pr[\theta_{taos} > 180]$
3. $Pr[\theta_{bridger} > \theta_{taos}]$ where $\theta_{bridger}$ is the mean annual snow fall at Bridger Bowl and θ_{taos} is the mean annual snow fall at Taos.

a. (5 pts)

How would you go about addressing the questions in a classical framework? Would you be able to compute these probabilities?

You could compare the means from the two groups using a paired t-test, but in a classical framework you cannot make probabilistic statements about parameters, so we could not answer the three questions of interest.

b. (10 pts)

Using the prior structure where $p(\sigma^2_{bridger}, \theta_{bridger}) = p(\theta_{bridger}|\sigma^2_{bridger})p(\sigma^2_{bridger})$ and $p(\sigma^2_{taos}, \theta_{taos}) = p(\theta_{taos}|\sigma^2_{taos})p(\sigma^2_{taos})$ compute the marginal posterior distributions $p(\theta_{bridger}|y_{bridger,1}, \ldots, y_{bridger,9}, \sigma^2_{bridger})$ and $p(\theta_{taos}|y_{taos,1}, \ldots, y_{taos,9}, \sigma^2_{taos})$, where $\sigma^2_{bridger}$ and σ^2_{taos} are the variances for snowfall (in inches), and $y_{i,j}$ is the observed snowfall at location i for reading j. Then using posterior samples from each distribution compute the three values specified above.

Let the prior distributions be:

$$
\begin{aligned}
\theta_{bridger}|\sigma^2_{bridger} &\sim N(250, \sigma^2_{bridger}/0.01) \\
1/\sigma^2_{bridger} &\sim Gamma(0.01, 0.01) \\
\theta_{taos}|\sigma^2_{taos} &\sim N(250, \sigma^2_{taos}/0.01) \\
1/\sigma^2_{taos} &\sim Gamma(0.01, 0.01)
\end{aligned}
$$

Then we can compute the marginal posterior distributions and sample from them:

```r
#### Posterior Sampling with Normal Model
set.seed(10062019)
num.obs <- length(bridger)

# specify terms for priors
nu.0 <- .02
sigma.sq.0 <- 1
mu.0 <- 250
kappa.0 <- .01


# compute terms in posterior
kappa.n <- kappa.0 + num.obs
nu.n <- nu.0 + num.obs
s.sq.bridger <- var(bridger)
s.sq.taos <- var(taos)

sigma.sq.n.bridger <- (1 / nu.n) * (nu.0 * sigma.sq.0 + (num.obs - 1) * s.sq.bridger +
(kappa.0*num.obs)/kappa.n * (mean(bridger) - mu.0)^2)
sigma.sq.n.taos <- (1 / nu.n) * (nu.0 * sigma.sq.0 + (num.obs - 1) * s.sq.taos +
(kappa.0*num.obs)/kappa.n * (mean(taos) - mu.0)^2)

mu.n.bridger <- (kappa.0 * mu.0 + num.obs * mean(bridger)) / kappa.n
mu.n.taos <- (kappa.0 * mu.0 + num.obs * mean(taos)) / kappa.n


# simulate from posterior
#install.packages("LearnBayes")
num.sims <- 10000
sigmasq_bridger <- rigamma(num.sims ,nu.n/2,sigma.sq.n.bridger*nu.n/2)
theta_bridger <- rnorm(num.sims, mu.n.bridger, sqrt(sigmasq_bridger/kappa.n))

sigmasq_taos <- rigamma(num.sims ,nu.n/2,sigma.sq.n.taos*nu.n/2)
theta_taos <- rnorm(num.sims, mu.n.taos, sqrt(sigmasq_taos/kappa.n))


##question 1
prob1 <- mean(theta_bridger > 250)

##question 2
prob2 <- mean(theta_taos > 180)

##question 3
prob3 <- mean(theta_bridger > theta_taos)
```

The estimated probability that the true mean annual amount of snowfall at Bridger Bowl is greater than 250 inches is 0.72. The estimated probability that the true mean annual amount of snowfall at Taos is greater than 180 inches is 0.23. The estimated probability that the true mean annual amount of snowfall at Bridger Bowl is greater than the true mean annual amount of snowfall at Taos is 0.998.

Q2.

a. (5 points)

Sketch out the steps for a Gibbs sampler algorithm.

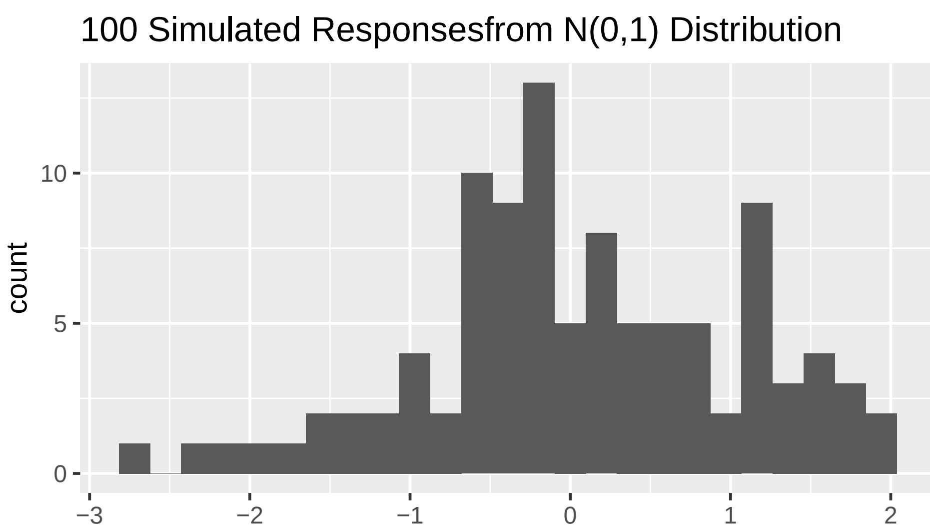
We need to start with the full conditional distributions: $p(\mu|y_1, ..., y_n, \sigma^2)$ and $p(\sigma^2|y_1, ..., y_n, \mu)$. We then sample $\mu^{(i+1)}$ from $p(\mu|y_1, ..., y_n, \sigma^{2(i)})$ and use that value to sample $\sigma^{2(i+1)}$ from $p(\sigma^2|y_1, ..., y_n, \mu^{(i+1)})$. We repeat this many times and let $\phi^{(i+1)} = \{\mu^{(i+1)}, \sigma^{2(i+1)}\}$.

b. (5 pts)

Simulating data is a key step in verifying your algorithms are working correctly. This will be more apparent as we start studying sophisticated hierarchical models.

Simulate 100 observations from a standard normal distribution and plot a histogram of your data.

```
set.seed(10062019)
num_obs <- 100
y <- rnorm(num_obs)
tibble(y = y) %>% ggplot(aes(y)) + geom_histogram(bins = 25) +
  ggtitle("100 Simulated Responsesfrom N(0,1) Distribution") + xlab('')
```



c. (5 pts)

Select and state prior distributions for θ the mean of the normal distribution and σ^2 the variance (or alternatively you may parameterize your model using the precision term).

$$\theta \sim N(0, 100^2)$$
$$\sigma^2 \sim InvGamma(0.01, 0.01)$$

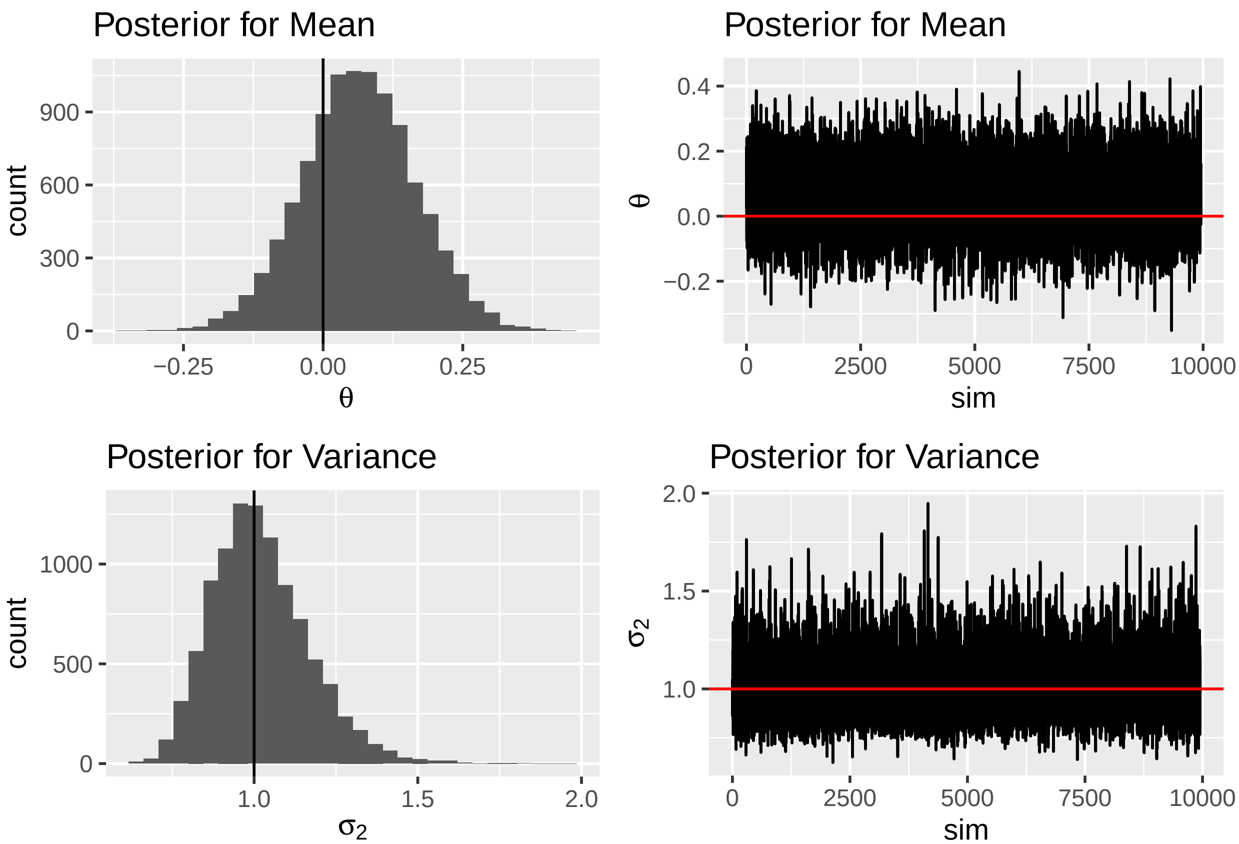
d. (5 pts)

Implement an MCMC algorithm (using a Gibbs sampler or JAGS/stan) to simulate from the joint posterior distribution $p(\theta, \sigma^2 | y_1, \ldots, y_{100})$. Plot trace plots and histograms of the marginal posterior distributions for θ and σ^2. Include the true values on these figures. Comment on the figures.

```r
mean_y <- mean(y)


### initialize vectors and set starting values and priors
num_sims <- 10000
theta_samples <- rep(1, num_sims)
sigmasq_samples <- rep(1, num_sims)
## Hyperparameters
# theta ~n (mu_0, tausq_0)
mu_0 <- 0
tausq_0 <- 10000

#sigmasq ~ IG(nu_0/2, nu_0 * sigmasq_0 / 2)
nu_0 <- .02
sigmasq_0 <- 1
for (i in 2:num_sims){
  # sample theta from full conditional
  mu_n <- (mu_0 / tausq_0 + num_obs * mean_y / sigmasq_samples[i-1]) /
    (1 / tausq_0 + num_obs / sigmasq_samples[i-1] )
  tausq_n <- 1 / (1/tausq_0 + num_obs / sigmasq_samples[i-1])
  theta_samples[i] <- rnorm(1,mu_n,sqrt(tausq_n))

  # sample (1/sigma.sq) from full conditional
  nu_n <- nu_0 + num_obs
  sigmasq_n <- 1/nu_n*(nu_0*sigmasq_0 + sum((y - theta_samples[i])^2))
  sigmasq_samples[i] <- rigamma(1, nu_n / 2,nu_n * sigmasq_n / 2)
}

#remove burnin
burn_in <- 50
theta_posterior <- theta_samples[-c(1:burn_in)]
sigmasq_posterior <- sigmasq_samples[-c(1:burn_in)]
```

```r
post <- tibble(vals = c(theta_posterior, sigmasq_posterior), type = rep(c('theta','sigmasq'), each = nu

hist_theta <- post %>% filter(type =='theta') %>% ggplot(aes(vals)) + geom_histogram() + xlab(expressio

trace_theta <- post %>% filter(type =='theta') %>% ggplot(aes(y = vals, x = sim)) + geom_line() + ylab(

hist_sigmasq <- post %>% filter(type =='sigmasq') %>% ggplot(aes(vals)) + geom_histogram() + xlab(expre

trace_sigmasq <- post %>% filter(type =='sigmasq') %>% ggplot(aes(y = vals, x = sim)) + geom_line() + y

grid.arrange(hist_theta, trace_theta, hist_sigmasq, trace_sigmasq, nrow = 2, ncol = 2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Based on the trace plots and histograms of the marginal posterior distributions for θ and σ^2, the algorithms appear to be working pretty well with the priors chosen. The true values of θ and σ^2 appear to be near the center of the marginal posterior histograms, and would certainly be contained in the 95% credible intervals created.