Lab 05

Colton Irons and Griffin Shelor

March 22, 2024

1 Preliminaries

Question 1: Set the seed for random numbers in R with set.seed(1991)

See R code

Question 2: Simulate 100 data points from a normal distribution with mean $\mu =$ 100 and variance $\sigma^2 = 25$. You will use these data to test the conjugate functions that you will write in section IV below. Call the data set y. Be careful here. R requires the standard deviation, not the variance, as a parameter.

See R code

Question 3: Write a function called draw.mean

See R code

Question 4: Write a function called draw.var

See R code

2 Writing a Sampler

Question 5: Set up a matrix for storing samples from the posterior distribution of the mean. The number of columns should equal the number of chains (3) and number of rows should equal the number of iterations (10,000). Do the same thing for storing samples from the posterior distribution of the variance.

See R code

Question 6: Assign initial values to the first row of each matrix, a different value for each of the chains. These can be virtually any value within the support of the random variable.

See R code

Question 7: Set up nested for loops to iterate from two to the total number of iterations for each of the three chains for each parameter (recall we already selected values for the first row–these were our initial values). Use the conjugate functions draw.mean and draw.var to draw a sample from the distribution of the mean using the value of the variance at the previous current iteration. Then make a draw from the variance using the value of the mean from the current iteration. Repeat. Assume vague priors for the mean and variance

See R code

3 Trace Plots and Plots of Marginal Posterior Distributions

Question 8: Discard the first 1000 iterations as burn-in. On the same figure, make three plots (one for each chain) with the value of the mean as a function of iteration number. This is called a trace plot.

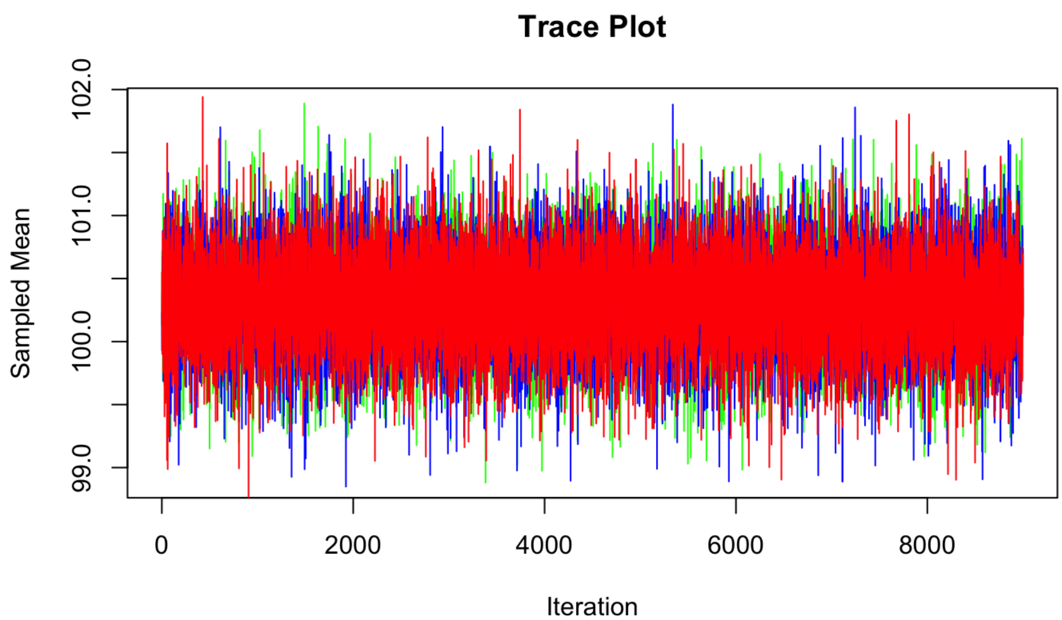

Figure 1: Trace plot of the μ chains

Question 9: For all chains combined, make a histogram of the samples of the mean retained after burn-in. Put a vertical line on the plot showing the generating value.

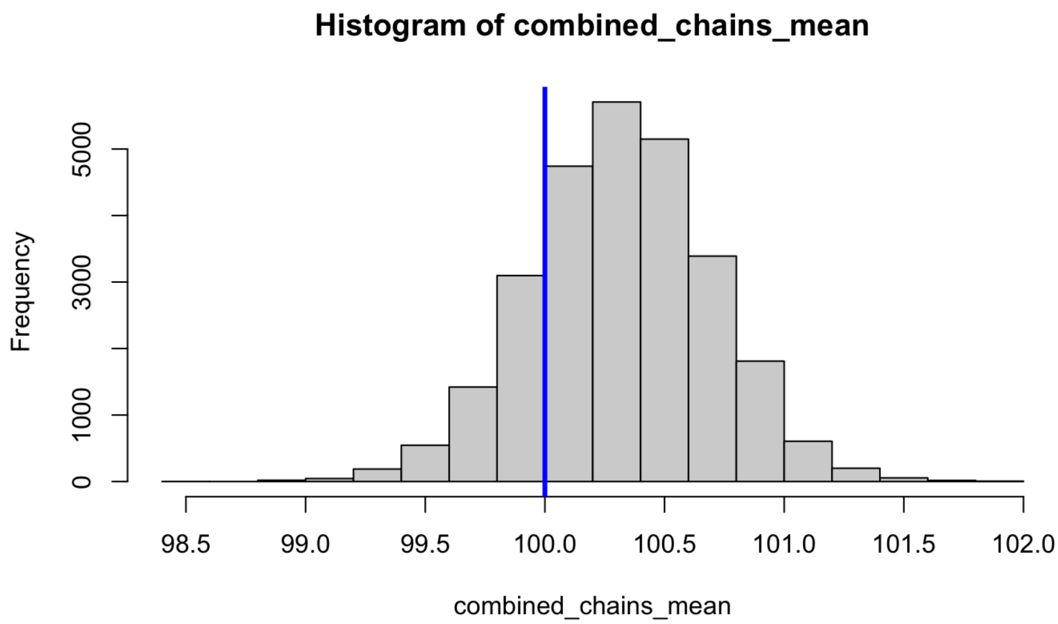

Figure 2: Histogram of the drawn values of μ for all chains

Question 10: Repeat steps 8-9 for the variance.

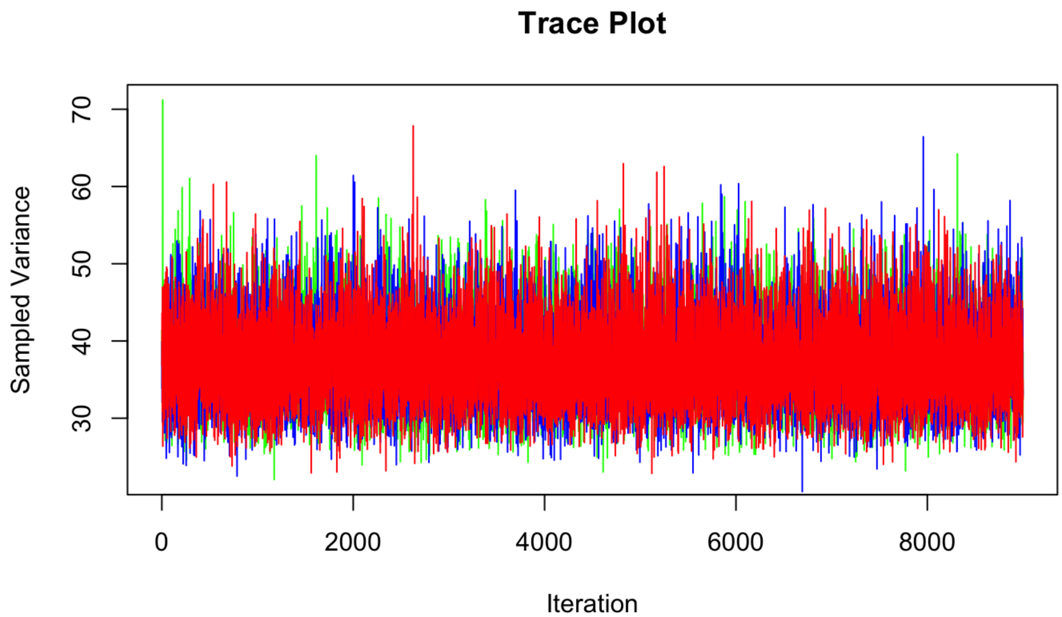

Figure 3: Trace plot of the σ^2 chains

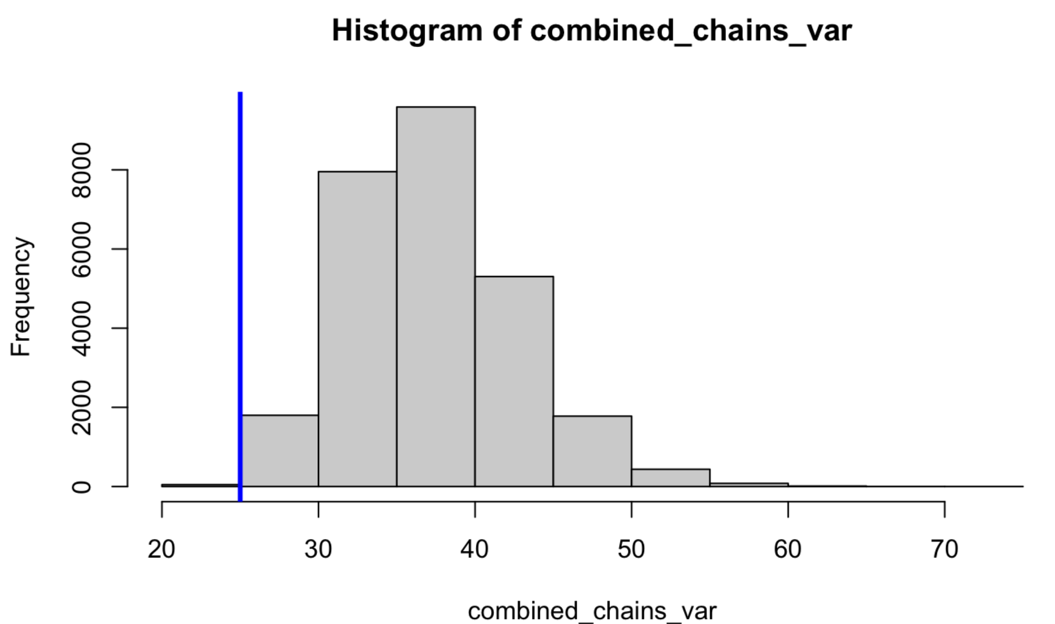

Figure 4: Histogram of all drawn values for σ^2 for all chains

Question 11: For both μ and σ^2, calculate the mean of all the chains combined and its standard deviation. Interpret these quantities.

The mean of the μ combined chain was 100.3186 while the SD was 0.3781. The mean of the σ^2 combined chain was 37.2907 while the SD was 5.3945.

Given that the mean of the μ chain is very similar to the mean of our simulated dataset, we can say that this iterative process of sampling from the posterior distribution of μ recovers the mean of our simulated dataset very well. We can also say that we can be very confident in knowing where the true value of μ is given that the SD of the combined μ chain is very low. However, the mean of our combined σ^2 is far apart from the variance of our simulated dataset, which may mean that we either need more iterations to recover the true value of σ^2 or that we need to test our MCMC algorithm on multiple simulated datasets to get a better understanding of the true value of σ^2. The standard deviation of our combined σ^2 chain is also high, indicating that we cannot have a high level of confidence in knowing the true value of σ^2.

Question 12: Compare the standard deviation of the posterior distribution of μ with an approximation using the standard deviation of the data divided by the square root of the sample size. What is this approximation called in the frequentist world?

The value of the standard deviation of the data divided by the square root of n is 0.6065, which is reasonably similar to our standard deviation of our posterior distribution. In frequentist statistics, this is referred to as the standard error.

Question 13: Vary the number of values in the simulated data set, e.g., n = 10, 100, 1000. What happens to the mean and variance of the posterior distributions as n gets large? Explain why you think this happens.

As n gets larger and larger, our mean and variance of our posterior distributions gradually shift closer and closer to the mean and variance of our simulated dataset. We believe this happens because as we increase our n, the random numbers in our simulated dataset become less meaningful individually in terms of their effect on the variance of the dataset. Because of this, an iterative chain in an MCMC algorithm will also be less subjected to variance and will be more likely to recover the true value from the posterior distributions of our parameters.

Question 14: Make the burn-in = 1 instead of 1000. Does this change your results? Why or why not?

Yes, this changes our results, but only slightly. While our mean is mostly still recovered by the MCMC algorithm except when $n = 10$, the variance is recovered but with a relatively high amount of uncertainty even when $n = 100$. Only when $n = 1000$ does the variance begin to be recovered with lower levels of uncertainty. This is likely because keeping more of the initial samples allows for potentially higher levels of variability before the MCMC algorithm narrows in on more likely range of values containing the true variance.

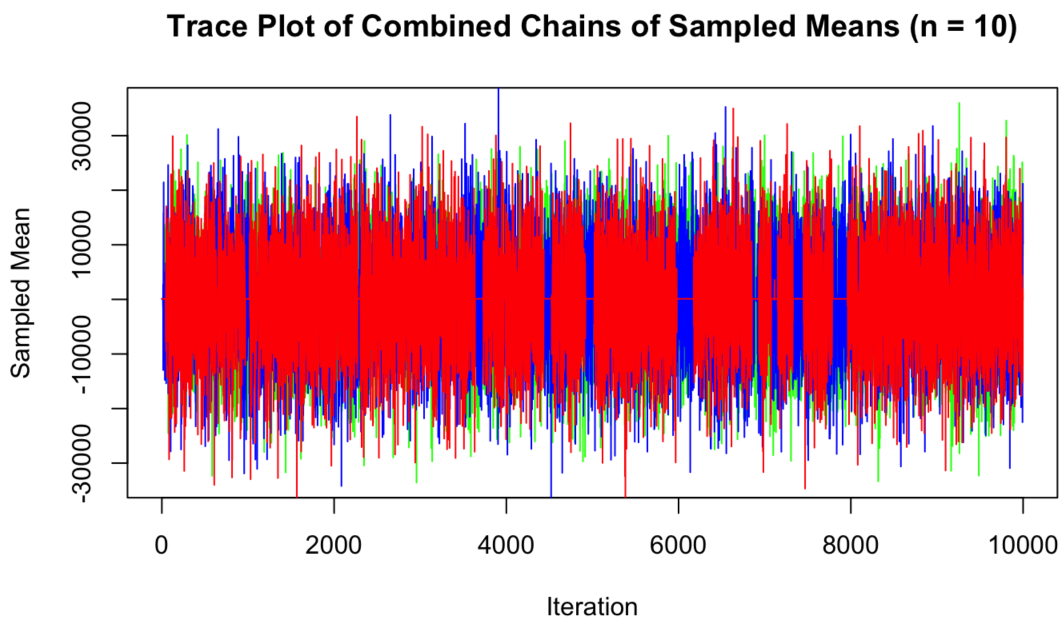

Figure 5: Trace plot when $n = 10$

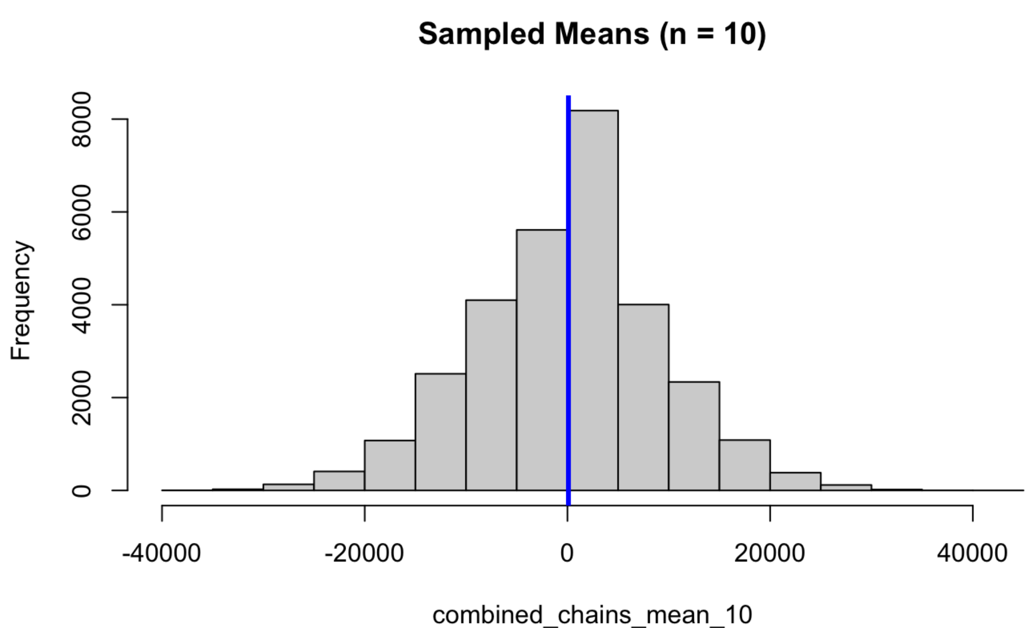

Figure 6: Our sampled means displayed high variance when $n = 10$

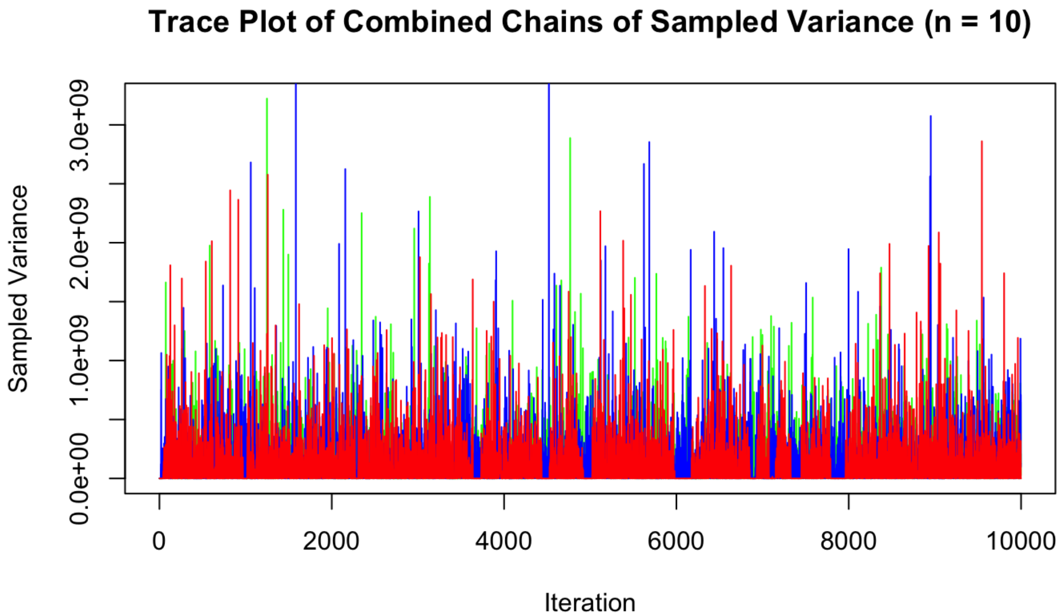

Figure 7: Trace plot of sampled variance when $n = 10$

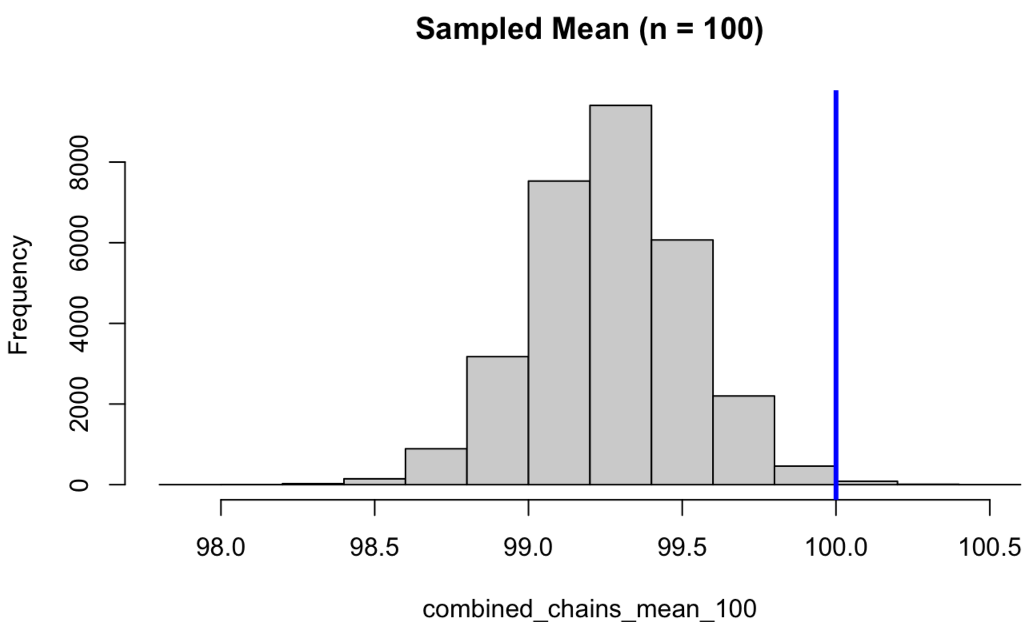

Figure 8: Sampled means when $n = 100$ displayed much less variance

Question 15: Reverse the order of the conjugate functions in step 7 so that the variance is drawn first followed by the mean. Be careful, this involves a bit more than simply reversing the order of the functions in the loop. Does this reordering have an effect on the posteriors? Why or why not?

It has very little effect on the posteriors, probably due to the large number of iterations in our chain and the vague priors we set in question 7. We also continue to sample within our sequence rather than sampling values out of sequence, so the order has at most a minimal impact.

4 Appendix: R Code

```
```{r}
library(pacman)
p_load(tidyverse, here, invgamma)
```

Section 1: Preliminaries
```{r}
## Question 1
set.seed(1991)
## Question 2
## simulating 100 data points with mean mu = 100 and variance = 25
y <- rnorm(100, 100, sqrt(25))

## Question 3
draw.mean <- function(data, mu_0, sigma_0, variance){
  mu_1 <- ((mu_0 / sigma_0^2) + (sum(data) / variance)) / ((1/sigma_0^2) + (
    length(data)/variance))
  sigma_sqd_1 <- 1 / ((1 / sigma_0^2) + length(data) / variance)
  random_meandraw <- rnorm(1, mu_1, sigma_sqd_1)
  return(random_meandraw)
}

## Question 4
draw.var <- function(data, alpha_0, beta_0, mu){
  alpha_1 <- alpha_0 + (length(data) / 2)
  beta_1 <- beta_0 + (sum((data - mu)^2) / 2)
  random_vardraw <- rinvgamma(1, shape = alpha_1, rate = beta_1)
  return(random_vardraw)
}
```


Section 2: Making a Sampler
```{r}
## Question 5
## creating matrices to store samples from posterior distributions of mean and
    variance
sample_mean_matrix <- matrix(nrow = 10000, ncol = 3)
sample_var_matrix <- matrix(nrow = 10000, ncol = 3)

## Question 6
sample_mean_matrix[1,] <- 20
```
```

```
sample_var_matrix[1,] <- 15

## Question 7
for (chain in 1:ncol(sample_mean_matrix)){
  for (iter in 2:nrow(sample_mean_matrix)){
    prev_var <- sample_var_matrix[iter-1,chain]
    sample_mean_matrix[iter,chain] <- draw.mean(y, 0, sqrt(10000), prev_var)
    prev_mean <- sample_mean_matrix[iter,chain]
    sample_var_matrix[iter, chain] <- draw.var(y, 0.01, 0.01, prev_mean)
  }
}
```


Section 3: Trace Plots and Plots of Marginal Posterior Distributions
```{r}
## Question 8
# discarding first 1000 rows as burn-in/warmup
post_burn_mean <- sample_mean_matrix[1001:nrow(sample_mean_matrix),]
# hist(post_burn_mean)

# making trace plots
plot(seq(1, nrow(post_burn_mean)), post_burn_mean[,1], xlab = "Iteration", ylab = "Sampled Mean", type = 'l', main = 'Trace Plot', col = 'green')
lines(post_burn_mean[,2], col = 'blue')
lines(post_burn_mean[,3], col = 'red')

## Question 9
# combining chains
combined_chains_mean <- c(post_burn_mean[,1], post_burn_mean[,2], post_burn_mean[,3])
hist(combined_chains_mean)
abline(v = 100, col = 'blue', lwd = 3)

## Question 10
# repeating 8 and 9 for the variance
post_burn_var <- sample_var_matrix[1001:nrow(sample_var_matrix),]

# making trace plots
plot(seq(1, nrow(post_burn_var)), post_burn_var[,1], xlab = "Iteration", ylab = "Sampled Variance", type = 'l', main = 'Trace Plot', col = 'green')
lines(post_burn_var[,2], col = 'blue')
lines(post_burn_var[,3], col = 'red')

# combining chains
combined_chains_var <- c(post_burn_var[,1], post_burn_var[,2], post_burn_var[,3])
hist(combined_chains_var)
abline(v = 25, col = 'blue', lwd = 3)

## Question 11
# mean and SD of combined chains for mu
mean_combined_chains_mean <- mean(combined_chains_mean)
sd_combined_chains_mean <- sd(combined_chains_mean)
```

````
# mean and SD of combined chains for sigma^2
mean_combined_chains_var <- mean(combined_chains_var)
sd_combined_chains_var <- sd(combined_chains_var)
# printing values
mean_combined_chains_mean
sd_combined_chains_mean
mean_combined_chains_var
sd_combined_chains_var

## Question 12
se_data <- sd(y) / sqrt(length(y))
se_data
```

## Question 13, n = 10
```{r}
# simulating new dataset
y_10 <- rnorm(10, 100, sqrt(25))


# creating matrices to store samples from posterior distributions of mean and variance
sample_mean_matrix_10 <- matrix(nrow = 10000, ncol = 3)
sample_var_matrix_10 <- matrix(nrow = 10000, ncol = 3)

# initializing values
sample_mean_matrix_10[1,] <- 20
sample_var_matrix_10[1,] <- 15

# sampling from posterior distributions
for (chain in 1:ncol(sample_mean_matrix_10)){
  for (iter in 2:nrow(sample_mean_matrix_10)){
    prev_var <- sample_var_matrix_10[iter-1,chain]
    sample_mean_matrix_10[iter,chain] <- draw.mean(y_10, 0, sqrt(10000), prev_var)
    prev_mean <- sample_mean_matrix_10[iter,chain]
    sample_var_matrix_10[iter, chain] <- draw.var(y_10, 0.01, 0.01, prev_mean)
  }
}

# discarding first 1000 rows as burn-in/warmup
post_burn_mean_10 <- sample_mean_matrix_10[1001:nrow(sample_mean_matrix_10),]
# hist(post_burn_mean_10)

# making trace plots
plot(seq(1, nrow(post_burn_mean_10)), post_burn_mean_10[,1], xlab = "Iteration", ylab = "Sampled Mean", type = 'l', main = 'Trace Plot', col = 'green')
lines(post_burn_mean_10[,2], col = 'blue')
lines(post_burn_mean_10[,3], col = 'red')

# combining chains
combined_chains_mean_10 <- c(post_burn_mean_10[,1], post_burn_mean_10[,2], post_burn_mean_10[,3])
hist(combined_chains_mean_10)
````

```
abline(v = 100, col = 'blue', lwd = 3)

# repeating for the variance
post_burn_var_10 <- sample_var_matrix_10[1001:nrow(sample_var_matrix_10),]

# making trace plots
plot(seq(1, nrow(post_burn_var_10)), post_burn_var_10[,1], xlab = "Iteration",
     ylab = "Sampled Variance", type = 'l', main = 'Trace Plot', col = 'green'
    )
lines(post_burn_var_10[,2], col = 'blue')
lines(post_burn_var_10[,3], col = 'red')

# combining chains
combined_chains_var_10 <- c(post_burn_var_10[,1], post_burn_var_10[,2], post_
    burn_var_10[,3])
hist(combined_chains_var_10)
abline(v = 25, col = 'blue', lwd = 3)


# mean and SD of combined chains for mu
mean_combined_chains_mean_10 <- mean(combined_chains_mean_10)
sd_combined_chains_mean_10 <- sd(combined_chains_mean_10)
# mean and SD of combined chains for sigma^2
mean_combined_chains_var_10 <- mean(combined_chains_var_10)
sd_combined_chains_var_10 <- sd(combined_chains_var_10)
# printing values
mean_combined_chains_mean_10
sd_combined_chains_mean_10
mean_combined_chains_var_10
sd_combined_chains_var_10
```

Question 13, n = 100
```{r}
# creating new simulated data
y_100 <- rnorm(100, 100, sqrt(25))

# creating matrices to store samples from posterior distributions of mean and
    variance
sample_mean_matrix_100 <- matrix(nrow = 10000, ncol = 3)
sample_var_matrix_100 <- matrix(nrow = 10000, ncol = 3)

# initializing values
sample_mean_matrix_100[1,] <- 20
sample_var_matrix_100[1,] <- 15

# sampling from posterior distributions
for (chain in 1:ncol(sample_mean_matrix_100)){
  for (iter in 2:nrow(sample_mean_matrix_100)){
    prev_var <- sample_var_matrix_100[iter-1,chain]
    sample_mean_matrix_100[iter,chain] <- draw.mean(y_100, 0, sqrt(10000),
    prev_var)
    prev_mean <- sample_mean_matrix_100[iter,chain]
```

````
    sample_var_matrix_100[iter, chain] <- draw.var(y_100, 0.01, 0.01, prev_mean)
  }
}

# discarding first 1000 rows as burn-in/warmup
post_burn_mean_100 <- sample_mean_matrix_100[1001:nrow(sample_mean_matrix_100),]
# hist(post_burn_mean_100)

# making trace plots
plot(seq(1, nrow(post_burn_mean_100)), post_burn_mean_100[,1], xlab = "Iteration", ylab = "Sampled Mean", type = 'l', main = 'Trace Plot', col = 'green')
lines(post_burn_mean_100[,2], col = 'blue')
lines(post_burn_mean_100[,3], col = 'red')

# combining chains
combined_chains_mean_100 <- c(post_burn_mean_100[,1], post_burn_mean_100[,2], post_burn_mean_100[,3])
hist(combined_chains_mean_100)
abline(v = 100, col = 'blue', lwd = 3)

# repeating for the variance
post_burn_var_100 <- sample_var_matrix_100[1001:nrow(sample_var_matrix_100),]

# making trace plots
plot(seq(1, nrow(post_burn_var_100)), post_burn_var_100[,1], xlab = "Iteration", ylab = "Sampled Variance", type = 'l', main = 'Trace Plot', col = 'green')
lines(post_burn_var_100[,2], col = 'blue')
lines(post_burn_var_100[,3], col = 'red')

# combining chains
combined_chains_var_100 <- c(post_burn_var_100[,1], post_burn_var_100[,2], post_burn_var_100[,3])
hist(combined_chains_var_100)
abline(v = 25, col = 'blue', lwd = 3)

# mean and SD of combined chains for mu
mean_combined_chains_mean_100 <- mean(combined_chains_mean_100)
sd_combined_chains_mean_100 <- sd(combined_chains_mean_100)
# mean and SD of combined chains for sigma^2
mean_combined_chains_var_100 <- mean(combined_chains_var_100)
sd_combined_chains_var_100 <- sd(combined_chains_var_100)
# printing values
mean_combined_chains_mean_100
sd_combined_chains_mean_100
mean_combined_chains_var_100
sd_combined_chains_var_100
```

## Question 13, n = 1000
```{r}
````

```r
# creating new simulated data
y_1000 <- rnorm(1000, 100, sqrt(25))

# creating matrices to store samples from posterior distributions of mean and variance
sample_mean_matrix_1000 <- matrix(nrow = 10000, ncol = 3)
sample_var_matrix_1000 <- matrix(nrow = 10000, ncol = 3)

# initializing values
sample_mean_matrix_1000[1,] <- 20
sample_var_matrix_1000[1,] <- 15

# sampling from posterior distributions
for (chain in 1:ncol(sample_mean_matrix_1000)){
  for (iter in 2:nrow(sample_mean_matrix_1000)){
    prev_var <- sample_var_matrix_1000[iter-1,chain]
    sample_mean_matrix_1000[iter,chain] <- draw.mean(y_1000, 0, sqrt(10000), prev_var)
    prev_mean <- sample_mean_matrix_1000[iter,chain]
    sample_var_matrix_1000[iter, chain] <- draw.var(y_1000, 0.01, 0.01, prev_mean)
  }
}

# discarding first 1000 rows as burn-in/warmup
post_burn_mean_1000 <- sample_mean_matrix_1000[1001:nrow(sample_mean_matrix_1000),]
# hist(post_burn_mean_100)

# making trace plots
plot(seq(1, nrow(post_burn_mean_1000)), post_burn_mean_1000[,1], xlab = "Iteration", ylab = "Sampled Mean", type = 'l', main = 'Trace Plot', col = 'green')
lines(post_burn_mean_1000[,2], col = 'blue')
lines(post_burn_mean_1000[,3], col = 'red')

# combining chains
combined_chains_mean_1000 <- c(post_burn_mean_1000[,1], post_burn_mean_1000[,2], post_burn_mean_1000[,3])
hist(combined_chains_mean_1000)
abline(v = 100, col = 'blue', lwd = 3)

# repeating for the variance
post_burn_var_1000 <- sample_var_matrix_1000[1001:nrow(sample_var_matrix_1000),]

# making trace plots
plot(seq(1, nrow(post_burn_var_1000)), post_burn_var_1000[,1], xlab = "Iteration", ylab = "Sampled Variance", type = 'l', main = 'Trace Plot', col = 'green')
lines(post_burn_var_1000[,2], col = 'blue')
lines(post_burn_var_1000[,3], col = 'red')

# combining chains
```

````
combined_chains_var_1000 <- c(post_burn_var_1000[,1], post_burn_var_1000[,2],
    post_burn_var_1000[,3])
hist(combined_chains_var_1000)
abline(v = 25, col = 'blue', lwd = 3)

# mean and SD of combined chains for mu
mean_combined_chains_mean_1000 <- mean(combined_chains_mean_1000)
sd_combined_chains_mean_1000 <- sd(combined_chains_mean_1000)
# mean and SD of combined chains for sigma^2
mean_combined_chains_var_1000 <- mean(combined_chains_var_1000)
sd_combined_chains_var_1000 <- sd(combined_chains_var_1000)
# printing values
mean_combined_chains_mean_1000
sd_combined_chains_mean_1000
mean_combined_chains_var_1000
sd_combined_chains_var_1000
```

## Question 13, n = 100000
```{r}
# creating new simulated data
y_100000 <- rnorm(100000, 100, sqrt(25))

# creating matrices to store samples from posterior distributions of mean and
    variance
sample_mean_matrix_100000 <- matrix(nrow = 10000, ncol = 3)
sample_var_matrix_100000 <- matrix(nrow = 10000, ncol = 3)

# initializing values
sample_mean_matrix_100000[1,] <- 20
sample_var_matrix_100000[1,] <- 15

# sampling from posterior distributions
for (chain in 1:ncol(sample_mean_matrix_100000)){
  for (iter in 2:nrow(sample_mean_matrix_100000)){
    prev_var <- sample_var_matrix_100000[iter-1,chain]
    sample_mean_matrix_100000[iter,chain] <- draw.mean(y_100000, 0, sqrt
    (10000), prev_var)
    prev_mean <- sample_mean_matrix_100000[iter,chain]
    sample_var_matrix_100000[iter, chain] <- draw.var(y_100000, 0.01, 0.01,
    prev_mean)
  }
}

# discarding first 1000 rows as burn-in/warmup
post_burn_mean_100000 <- sample_mean_matrix_100000[1001:nrow(sample_mean_
    matrix_100000),]
# hist(post_burn_mean_100)

# making trace plots
plot(seq(1, nrow(post_burn_mean_100000)), post_burn_mean_100000[,1], xlab = "
    Iteration", ylab = "Sampled Mean", type = 'l', main = 'Trace Plot', col =
    'green')
lines(post_burn_mean_100000[,2], col = 'blue')
````

```
lines(post_burn_mean_100000[,3], col = 'red')

# combining chains
combined_chains_mean_100000 <- c(post_burn_mean_100000[,1], post_burn_mean_100000[,2], post_burn_mean_100000[,3])
hist(combined_chains_mean_100000)
abline(v = 100, col = 'blue', lwd = 3)

# repeating for the variance
post_burn_var_100000 <- sample_var_matrix_100000[1001:nrow(sample_var_matrix_100000),]

# making trace plots
plot(seq(1, nrow(post_burn_var_100000)), post_burn_var_100000[,1], xlab = "Iteration", ylab = "Sampled Variance", type = 'l', main = 'Trace Plot', col = 'green')
lines(post_burn_var_100000[,2], col = 'blue')
lines(post_burn_var_100000[,3], col = 'red')

# combining chains
combined_chains_var_100000 <- c(post_burn_var_100000[,1], post_burn_var_100000[,2], post_burn_var_100000[,3])
hist(combined_chains_var_100000)
abline(v = 25, col = 'blue', lwd = 3)

# mean and SD of combined chains for mu
mean_combined_chains_mean_100000 <- mean(combined_chains_mean_100000)
sd_combined_chains_mean_100000 <- sd(combined_chains_mean_100000)
# mean and SD of combined chains for sigma^2
mean_combined_chains_var_100000 <- mean(combined_chains_var_100000)
sd_combined_chains_var_100000 <- sd(combined_chains_var_100000)
# printing values
mean_combined_chains_mean_100000
sd_combined_chains_mean_100000
mean_combined_chains_var_100000
sd_combined_chains_var_100000
```

Question 14
```{r}
##### n = 10 #####
# simulating new dataset
y_10 <- rnorm(10, 100, sqrt(25))


# creating matrices to store samples from posterior distributions of mean and variance
sample_mean_matrix_10 <- matrix(nrow = 10000, ncol = 3)
sample_var_matrix_10 <- matrix(nrow = 10000, ncol = 3)

# initializing values
sample_mean_matrix_10[1,] <- 20
sample_var_matrix_10[1,] <- 15

```

```
# sampling from posterior distributions
for (chain in 1:ncol(sample_mean_matrix_10)){
  for (iter in 2:nrow(sample_mean_matrix_10)){
    prev_var <- sample_var_matrix_10[iter-1,chain]
    sample_mean_matrix_10[iter,chain] <- draw.mean(y_10, 0, sqrt(10000), prev_
    var)
    prev_mean <- sample_mean_matrix_10[iter,chain]
    sample_var_matrix_10[iter, chain] <- draw.var(y_10, 0.01, 0.01, prev_mean)
  }
}

# discarding first 1000 rows as burn-in/warmup
post_burn_mean_10 <- sample_mean_matrix_10[2:nrow(sample_mean_matrix_10),]
# hist(post_burn_mean_10)

# making trace plots
plot(seq(1, nrow(post_burn_mean_10)), post_burn_mean_10[,1], xlab = "Iteration
    ", ylab = "Sampled Mean", type = 'l', main = 'Trace Plot of Combined
    Chains of Sampled Means (n = 10)', col = 'green')
lines(post_burn_mean_10[,2], col = 'blue')
lines(post_burn_mean_10[,3], col = 'red')

# combining chains
combined_chains_mean_10 <- c(post_burn_mean_10[,1], post_burn_mean_10[,2],
    post_burn_mean_10[,3])
hist(combined_chains_mean_10, main = "Sampled Means (n = 10)")
abline(v = 100, col = 'blue', lwd = 3)

# repeating for the variance
post_burn_var_10 <- sample_var_matrix_10[2:nrow(sample_var_matrix_10),]

# making trace plots
plot(seq(1, nrow(post_burn_var_10)), post_burn_var_10[,1], xlab = "Iteration",
     ylab = "Sampled Variance", type = 'l', main = 'Trace Plot of Combined
    Chains of Sampled Variance (n = 10)', col = 'green')
lines(post_burn_var_10[,2], col = 'blue')
lines(post_burn_var_10[,3], col = 'red')

# combining chains
combined_chains_var_10 <- c(post_burn_var_10[,1], post_burn_var_10[,2], post_
    burn_var_10[,3])
hist(combined_chains_var_10, main = "Sampled Variance (n = 10)")
abline(v = 25, col = 'blue', lwd = 3)

# mean and SD of combined chains for mu
mean_combined_chains_mean_10 <- mean(combined_chains_mean_10)
sd_combined_chains_mean_10 <- sd(combined_chains_mean_10)
# mean and SD of combined chains for sigma^2
mean_combined_chains_var_10 <- mean(combined_chains_var_10)
sd_combined_chains_var_10 <- sd(combined_chains_var_10)
# printing values
mean_combined_chains_mean_10
sd_combined_chains_mean_10
mean_combined_chains_var_10
```

```r
sd_combined_chains_var_10


##### n = 100 #####
# creating new simulated data
y_100 <- rnorm(100, 100, sqrt(25))

# creating matrices to store samples from posterior distributions of mean and
  variance
sample_mean_matrix_100 <- matrix(nrow = 10000, ncol = 3)
sample_var_matrix_100 <- matrix(nrow = 10000, ncol = 3)

# initializing values
sample_mean_matrix_100[1,] <- 20
sample_var_matrix_100[1,] <- 15

# sampling from posterior distributions
for (chain in 1:ncol(sample_mean_matrix_100)){
  for (iter in 2:nrow(sample_mean_matrix_100)){
    prev_var <- sample_var_matrix_100[iter-1,chain]
    sample_mean_matrix_100[iter,chain] <- draw.mean(y_100, 0, sqrt(10000),
    prev_var)
    prev_mean <- sample_mean_matrix_100[iter,chain]
    sample_var_matrix_100[iter, chain] <- draw.var(y_100, 0.01, 0.01, prev_
    mean)
  }
}

# discarding first 1000 rows as burn-in/warmup
post_burn_mean_100 <- sample_mean_matrix_100[2:nrow(sample_mean_matrix_100),]
# hist(post_burn_mean_100)

# making trace plots
plot(seq(1, nrow(post_burn_mean_100)), post_burn_mean_100[,1], xlab = "
  Iteration", ylab = "Sampled Mean", type = 'l', main = 'Trace Plot (n =
  100)', col = 'green')
lines(post_burn_mean_100[,2], col = 'blue')
lines(post_burn_mean_100[,3], col = 'red')

# combining chains
combined_chains_mean_100 <- c(post_burn_mean_100[,1], post_burn_mean_100[,2],
  post_burn_mean_100[,3])
hist(combined_chains_mean_100, main = "Sampled Mean (n = 100)")
abline(v = 100, col = 'blue', lwd = 3)

# repeating for the variance
post_burn_var_100 <- sample_var_matrix_100[2:nrow(sample_var_matrix_100),]

# making trace plots
plot(seq(1, nrow(post_burn_var_100)), post_burn_var_100[,1], xlab = "Iteration
  ", ylab = "Sampled Variance", type = 'l', main = 'Trace Plot', col = '
  green')
lines(post_burn_var_100[,2], col = 'blue')
lines(post_burn_var_100[,3], col = 'red')
```

```
# combining chains
combined_chains_var_100 <- c(post_burn_var_100[,1], post_burn_var_100[,2], post_burn_var_100[,3])
hist(combined_chains_var_100, main = "Sampled Variance (n = 100)")
abline(v = 25, col = 'blue', lwd = 3)

# mean and SD of combined chains for mu
mean_combined_chains_mean_100 <- mean(combined_chains_mean_100)
sd_combined_chains_mean_100 <- sd(combined_chains_mean_100)
# mean and SD of combined chains for sigma^2
mean_combined_chains_var_100 <- mean(combined_chains_var_100)
sd_combined_chains_var_100 <- sd(combined_chains_var_100)
# printing values
mean_combined_chains_mean_100
sd_combined_chains_mean_100
mean_combined_chains_var_100
sd_combined_chains_var_100


##### n = 1000 #####
# creating new simulated data
y_1000 <- rnorm(1000, 100, sqrt(25))

# creating matrices to store samples from posterior distributions of mean and variance
sample_mean_matrix_1000 <- matrix(nrow = 10000, ncol = 3)
sample_var_matrix_1000 <- matrix(nrow = 10000, ncol = 3)

# initializing values
sample_mean_matrix_1000[1,] <- 20
sample_var_matrix_1000[1,] <- 15

# sampling from posterior distributions
for (chain in 1:ncol(sample_mean_matrix_1000)){
  for (iter in 2:nrow(sample_mean_matrix_1000)){
    prev_var <- sample_var_matrix_1000[iter-1,chain]
    sample_mean_matrix_1000[iter,chain] <- draw.mean(y_1000, 0, sqrt(10000), prev_var)
    prev_mean <- sample_mean_matrix_1000[iter,chain]
    sample_var_matrix_1000[iter, chain] <- draw.var(y_1000, 0.01, 0.01, prev_mean)
  }
}

# discarding first 1000 rows as burn-in/warmup
post_burn_mean_1000 <- sample_mean_matrix_1000[2:nrow(sample_mean_matrix_1000),]
# hist(post_burn_mean_100)

# making trace plots
plot(seq(1, nrow(post_burn_mean_1000)), post_burn_mean_1000[,1], xlab = "Iteration", ylab = "Sampled Mean", type = 'l', main = 'Trace Plot', col = 'green')
```

```
lines(post_burn_mean_1000[,2], col = 'blue')
lines(post_burn_mean_1000[,3], col = 'red')

# combining chains
combined_chains_mean_1000 <- c(post_burn_mean_1000[,1], post_burn_mean_
    1000[,2], post_burn_mean_1000[,3])
hist(combined_chains_mean_1000)
abline(v = 100, col = 'blue', lwd = 3)

# repeating for the variance
post_burn_var_1000 <- sample_var_matrix_1000[2:nrow(sample_var_matrix_1000),]

# making trace plots
plot(seq(1, nrow(post_burn_var_1000)), post_burn_var_1000[,1], xlab = "
    Iteration", ylab = "Sampled Variance", type = 'l', main = 'Trace Plot',
    col = 'green')
lines(post_burn_var_1000[,2], col = 'blue')
lines(post_burn_var_1000[,3], col = 'red')

# combining chains
combined_chains_var_1000 <- c(post_burn_var_1000[,1], post_burn_var_1000[,2],
    post_burn_var_1000[,3])
hist(combined_chains_var_1000)
abline(v = 25, col = 'blue', lwd = 3)

# mean and SD of combined chains for mu
mean_combined_chains_mean_1000 <- mean(combined_chains_mean_1000)
sd_combined_chains_mean_1000 <- sd(combined_chains_mean_1000)
# mean and SD of combined chains for sigma^2
mean_combined_chains_var_1000 <- mean(combined_chains_var_1000)
sd_combined_chains_var_1000 <- sd(combined_chains_var_1000)
# printing values
mean_combined_chains_mean_1000
sd_combined_chains_mean_1000
mean_combined_chains_var_1000
sd_combined_chains_var_1000


##### n = 100000 #####
# creating new simulated data
y_100000 <- rnorm(100000, 100, sqrt(25))

# creating matrices to store samples from posterior distributions of mean and
    variance
sample_mean_matrix_100000 <- matrix(nrow = 10000, ncol = 3)
sample_var_matrix_100000 <- matrix(nrow = 10000, ncol = 3)

# initializing values
sample_mean_matrix_100000[1,] <- 20
sample_var_matrix_100000[1,] <- 15

# sampling from posterior distributions
for (chain in 1:ncol(sample_mean_matrix_100000)){
  for (iter in 2:nrow(sample_mean_matrix_100000)){
```

```r
    prev_var <- sample_var_matrix_100000[iter-1,chain]
    sample_mean_matrix_100000[iter,chain] <- draw.mean(y_100000, 0, sqrt(10000), prev_var)
    prev_mean <- sample_mean_matrix_100000[iter,chain]
    sample_var_matrix_100000[iter, chain] <- draw.var(y_100000, 0.01, 0.01, prev_mean)
  }
}

# discarding first 1000 rows as burn-in/warmup
post_burn_mean_100000 <- sample_mean_matrix_100000[2:nrow(sample_mean_matrix_100000),]
# hist(post_burn_mean_100)

# making trace plots
plot(seq(1, nrow(post_burn_mean_100000)), post_burn_mean_100000[,1], xlab = "Iteration", ylab = "Sampled Mean", type = 'l', main = 'Trace Plot of Combined Chains of Sampled Means (n = 100000)', col = 'green')
lines(post_burn_mean_100000[,2], col = 'blue')
lines(post_burn_mean_100000[,3], col = 'red')

# combining chains
combined_chains_mean_100000 <- c(post_burn_mean_100000[,1], post_burn_mean_100000[,2], post_burn_mean_100000[,3])
hist(combined_chains_mean_100000, main = "Sampled Means (n = 100000)")
abline(v = 100, col = 'blue', lwd = 3)

# repeating for the variance
post_burn_var_100000 <- sample_var_matrix_100000[2:nrow(sample_var_matrix_100000),]

# making trace plots
plot(seq(1, nrow(post_burn_var_100000)), post_burn_var_100000[,1], xlab = "Iteration", ylab = "Sampled Variance", type = 'l', main = 'Trace Plot of Combined Chains of Sampled Variance (n = 100000)', col = 'green')
lines(post_burn_var_100000[,2], col = 'blue')
lines(post_burn_var_100000[,3], col = 'red')

# combining chains
combined_chains_var_100000 <- c(post_burn_var_100000[,1], post_burn_var_100000[,2], post_burn_var_100000[,3])
hist(combined_chains_var_100000, main = "Sampled Variance (n = 100000)")
abline(v = 25, col = 'blue', lwd = 3)

# mean and SD of combined chains for mu
mean_combined_chains_mean_100000 <- mean(combined_chains_mean_100000)
sd_combined_chains_mean_100000 <- sd(combined_chains_mean_100000)
# mean and SD of combined chains for sigma^2
mean_combined_chains_var_100000 <- mean(combined_chains_var_100000)
sd_combined_chains_var_100000 <- sd(combined_chains_var_100000)
# printing values
mean_combined_chains_mean_100000
sd_combined_chains_mean_100000
mean_combined_chains_var_100000
```

```
sd_combined_chains_var_100000
```

Question 15
```{r}
## setting seed
set.seed(1991)

## new draw mean function
draw.mean_new <- function(data, mu_0, sigma_0, variance){
  mu_1 <- ((mu_0 / sigma_0^2) + (sum(data) / variance)) / ((1/sigma_0^2) + (length(data)/variance))
  sigma_sqd_1 <- 1 / ((1 / sigma_0^2) + length(data) / variance)
  random_meandraw <- rnorm(1, mu_1, sigma_sqd_1)
  return(random_meandraw)
}

## new draw var function
draw.var_new <- function(data, alpha_0, beta_0, mu){
  alpha_1 <- alpha_0 + (length(data) / 2)
  beta_1 <- beta_0 + (sum((data - mu)^2) / 2)
  random_vardraw <- rinvgamma(1, shape = alpha_1, rate = beta_1)
  return(random_vardraw)
}

## creating matrices to store samples from posterior distributions of mean and variance
sample_mean_matrix <- matrix(nrow = 10000, ncol = 3)
sample_var_matrix <- matrix(nrow = 10000, ncol = 3)

## initializing first mean and variance values
sample_mean_matrix[1,] <- 20
sample_var_matrix[1,] <- 15

## drawing variance first, then mean
for (chain in 1:ncol(sample_mean_matrix)){
  for (iter in 2:nrow(sample_mean_matrix)){
    prev_mean <- sample_mean_matrix[iter-1,chain]
    sample_var_matrix[iter, chain] <- draw.var_new(y, 0.01, 0.01, prev_mean)
    prev_var <- sample_var_matrix[iter,chain]
    sample_mean_matrix[iter, chain] <- draw.mean_new(y, 0, sqrt(10000), prev_var)
  }
}

# discarding first 1000 rows as burn-in/warmup
post_burn_mean <- sample_mean_matrix[1001:nrow(sample_mean_matrix),]
# hist(post_burn_mean)

# making trace plots
plot(seq(1, nrow(post_burn_mean)), post_burn_mean[,1], xlab = "Iteration", ylab = "Sampled Mean", type = 'l', main = 'Trace Plot', col = 'green')
lines(post_burn_mean[,2], col = 'blue')
lines(post_burn_mean[,3], col = 'red')
```

```
# combining chains
combined_chains_mean <- c(post_burn_mean[,1], post_burn_mean[,2], post_burn_mean[,3])
hist(combined_chains_mean)
abline(v = 100, col = 'blue', lwd = 3)

# repeating burn-in removal, plots for the variance
post_burn_var <- sample_var_matrix[1001:nrow(sample_var_matrix),]

# making trace plots
plot(seq(1, nrow(post_burn_var)), post_burn_var[,1], xlab = "Iteration", ylab = "Sampled Variance", type = 'l', main = 'Trace Plot', col = 'green')
lines(post_burn_var[,2], col = 'blue')
lines(post_burn_var[,3], col = 'red')

# combining chains
combined_chains_var <- c(post_burn_var[,1], post_burn_var[,2], post_burn_var[,3])
hist(combined_chains_var)
abline(v = 25, col = 'blue', lwd = 3)


# mean and SD of combined chains for mu
mean_combined_chains_mean <- mean(combined_chains_mean)
sd_combined_chains_mean <- sd(combined_chains_mean)
# mean and SD of combined chains for sigma^2
mean_combined_chains_var <- mean(combined_chains_var)
sd_combined_chains_var <- sd(combined_chains_var)
# printing values
mean_combined_chains_mean
sd_combined_chains_mean
mean_combined_chains_var
sd_combined_chains_var
```
```
```