STA365 Assignment 1

Consider the model:

$$y_i \sim N(\mu, 1)$$

where $i = 1, ..., n$ with prior $\mu \sim N(m, s)$, where m and s > 0 and may depend on the sample y.

Question 1:

What is the choice of m and s that will make $\mathbb{E}(\mu|y) = 1$? What will the variance do as $n \to \infty$?

Based on the distribution of y and μ, we know that the densities are as follows:

$$p(y_i|\mu) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(y_i - \mu)^2}{2\sigma^2})$$
$$p(\mu) = \frac{1}{\sqrt{2\pi s^2}} exp(-\frac{(\mu - m)^2}{2s^2})$$

Now, we know that

$$\begin{aligned}
p(\mu|\boldsymbol{y}) &= \frac{p(\boldsymbol{y}|\mu) * p(\mu)}{p(\boldsymbol{y})} \\
&\propto p(\boldsymbol{y}|\mu) * p(\mu) \\
&\propto p(y_1|\mu) * p(y_2|\mu) * ... * p(y_n|\mu) * p(\mu) \\
&\propto \prod_{i=1}^{n} p(y_i|\mu) * p(\mu) \\
&\propto (\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} exp(-\frac{(y_i - \mu)^2}{2\sigma^2})) * \frac{1}{\sqrt{2\pi s^2}} exp(-\frac{(\mu - m)^2}{2s^2}) \\
&\propto exp(-(\sum_{i=1}^{n} \frac{(y_i - \mu)^2}{2\sigma^2}) - \frac{(\mu - m)^2}{2s^2}) \\
&\propto exp(-\frac{\sum_{i=1}^{n} {y_i}^2}{2\sigma^2} - \frac{n\mu^2}{2\sigma^2} + 2\frac{\sum_{i=1}^{n} y_i\mu}{2\sigma^2} - \frac{\mu^2}{2s^2} - \frac{m^2}{2s^2} + 2\frac{\mu m}{2s^2}) \\
&\propto exp(-\frac{n\mu^2}{2\sigma^2} + 2\frac{\sum_{i=1}^{n} y_i\mu}{2\sigma^2} - \frac{\mu^2}{2s^2} + 2\frac{\mu m}{2s^2}) \\
&\propto exp(\frac{-n\mu^2 s^2 - \mu^2\sigma^2 + 2\mu s^2 \sum_{i=1}^{n} y_i + 2\mu m\sigma^2}{2\sigma^2 s^2}) \\
&\propto exp(\frac{-\mu^2(ns^2 + \sigma^2) + 2\mu(s^2 \sum_{i=1}^{n} y_i + m\sigma^2)}{2\sigma^2 s^2}) \\
&\propto exp(\frac{-\mu^2 + 2\mu\frac{s^2 \sum_{i=1}^{n} y_i + m\sigma^2}{ns^2+\sigma^2} - (\frac{s^2 \sum_{i=1}^{n} y_i + m\sigma^2}{ns^2+\sigma^2})^2}{\frac{2\sigma^2 s^2}{ns^2+\sigma^2}}) * exp(\frac{(\frac{s^2 \sum_{i=1}^{n} y_i + m\sigma^2}{ns^2+\sigma^2})^2}{\frac{2\sigma^2 s^2}{ns^2+\sigma^2}}) \\
&\propto exp(\frac{-\mu^2 + 2\mu\frac{s^2 \sum_{i=1}^{n} y_i + m\sigma^2}{ns^2+\sigma^2} - (\frac{s^2 \sum_{i=1}^{n} y_i + m\sigma^2}{ns^2+\sigma^2})^2}{\frac{2\sigma^2 s^2}{ns^2+\sigma^2}}) \\
&\propto exp(-\frac{1}{2(\frac{\sigma^2 s^2}{ns^2+\sigma^2})}(\mu - (\frac{s^2 \sum_{i=1}^{n} y_i + m\sigma^2}{ns^2 + \sigma^2}))^2
\end{aligned}$$

Based on this, we can say that the posterior distribution is

$$\mu|\boldsymbol{y} \sim N(\frac{s^2\sum_{i=1}^n y_i + m\sigma^2}{ns^2+\sigma^2}, \frac{\sigma^2 s^2}{ns^2+\sigma^2})$$

where the distribution is of the form N(mean, variance).

We know that $\sigma^2 = 1$. So, we can rewrite this distribution as

$$\mu|\boldsymbol{y} \sim N(\frac{s^2\sum_{i=1}^n y_i + m}{ns^2+1}, \frac{s^2}{ns^2+1})$$

Part 1:

Based on this posterior distribution, we want to find m and s^2 such that $\mathbb{E}(\mu|\boldsymbol{y}) = 1$. Based on that and the above posterior, we can say that:

$$\frac{s^2\sum_{i=1}^n y_i + m}{ns^2+1} = 1$$

. Therefore,

$$s^2\sum_{i=1}^n y_i + m = ns^2 + 1$$

$$m = ns^2 - s^2\sum_{i=1}^n y_i + 1$$

$$\boxed{\therefore m = s^2(n - \sum_{i=1}^n y_i) + 1}$$

So, in this case, there are multiple choices of m and s^2 that will satisfy the relation we want.

Part 2:

Also, based on the above posterior distribution, we want to know what will the variance do as $n \to \infty$.

Based on the posterior, we know that variance of posterior is $\frac{s^2}{ns^2+1}$. So, if $n \to \infty$, then we can say that the variance goes to 0 as the denominator goes to ∞.

Question 2:

Here, we first want to simulate datasets of size n = 10,100,1000 with μ = 1, 0, -1, -10. So, we do that with the following code.

(Generally, the code is in appendix. However, just because the question asked to simulate datasets, to show how the data is simulated, the code is presented here and not in appendix for this particular case.)

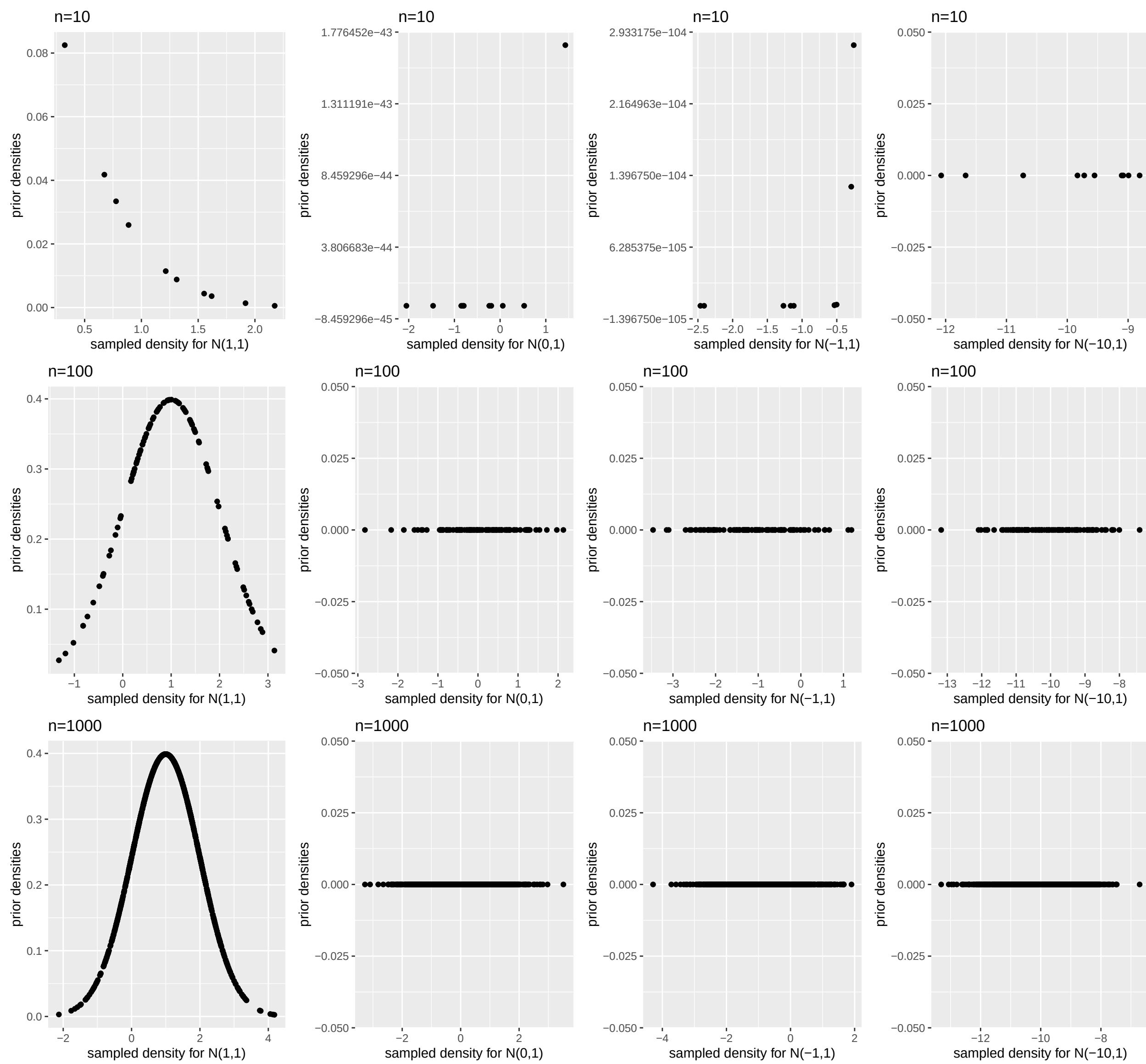
```
set.seed(101) # setting seed for consistency

# Simulating data sets:
d1_10p <- rnorm(10, mean = 1, sd = 1)
d2_10p <- rnorm(10, mean = 0, sd = 1)
d3_10p <- rnorm(10, mean = -1, sd = 1)
d4_10p <- rnorm(10, mean = -10, sd = 1)
d5_100p <- rnorm(100, mean = 1, sd = 1)
d6_100p <- rnorm(100, mean = 0, sd = 1)
```

```r
d7_100p <- rnorm(100, mean = -1, sd = 1)
d8_100p <- rnorm(100, mean = -10, sd = 1)
d9_1000p <- rnorm(1000, mean = 1, sd = 1)
d10_1000p <- rnorm(1000, mean = 0, sd = 1)
d11_1000p <- rnorm(1000, mean = -1, sd = 1)
d12_1000p <- rnorm(1000, mean = -10, sd = 1)
```

Now that we have simulated the datasets, we want to plot the priors from part 1 for each of these 12 simulated datasets. In the plot, row 1 corresponds to datasets with n=10. While row 2 and 3 correspond to datasets with n=100 and 1000 respectively. Within each row, the first column, represents dataset with $\mu = 1$. The second, third and fourth represent dataset with $\mu = 0, -1, -10$ respectively.

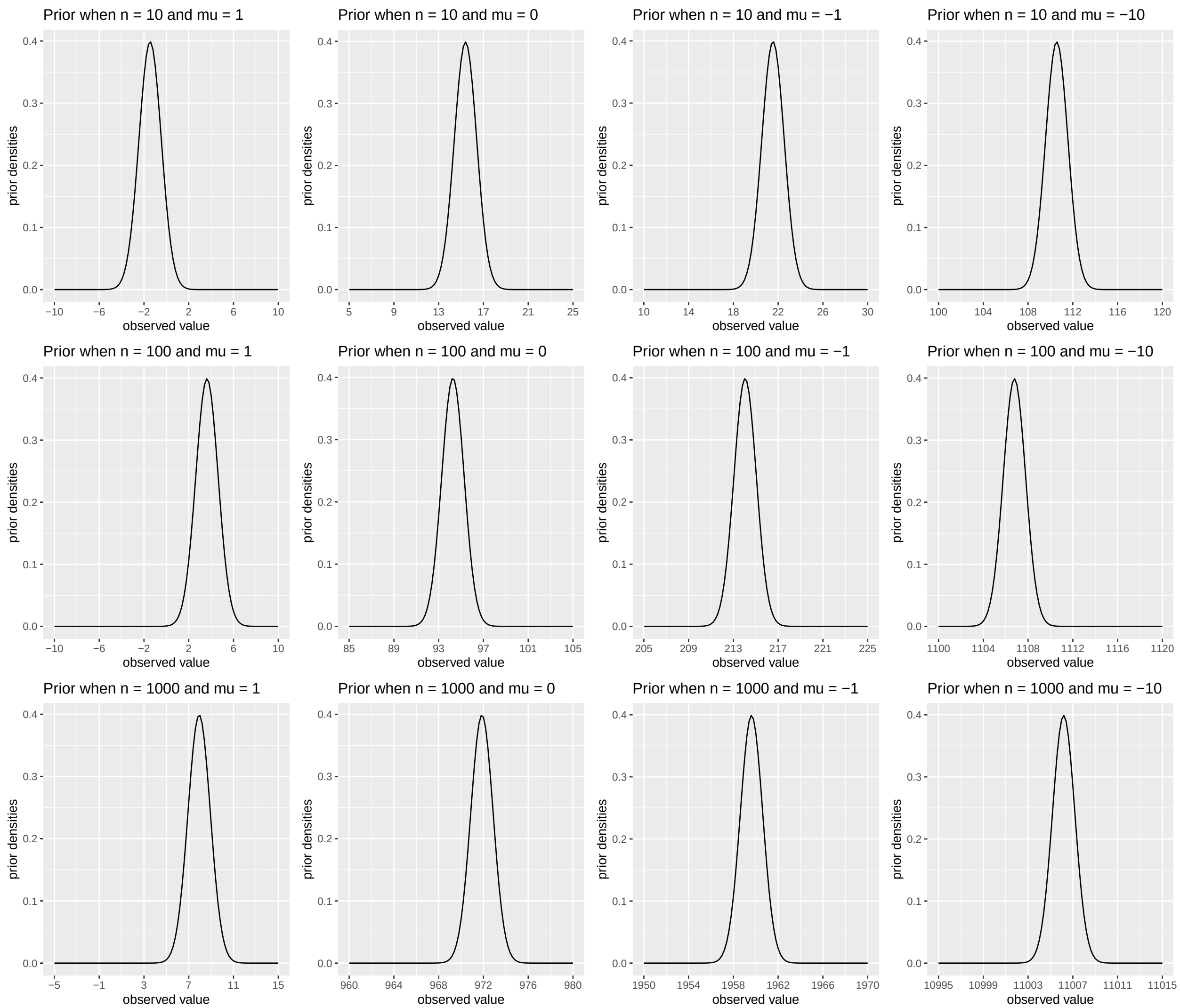
Initially, if we plot priors for each of these 12 simulated datasets, with x-axis ranging from minimum to maximum of these simulated datasets, we observe these plots. However, these do not provide us much information.

So, following these, we will plot the priors on a larger set of x-values so that we could observe the distribution of priors.



Now that we saw the plots of priors with x-axis ranging from minimum to maximum of these simulated datasets, we observe that these do not provide us much information.

So, following these, we will plot the priors on a larger set of x-values so that we could observe the distribution of priors.



- Based on the plots of priors, we first observe that all of them look very close to normal with each one centered at different places.
- For a specific value of μ (each column), we observe that as n increases, the parameter (and so, the distribution) for prior (i.e., m) shifts towards right.
- Similarly, for a specific value of n, the value of μ impacts parameter and thus the distribution quite a lot. The greater the μ, the more the distibution (including its parameter) shifts towards the right.

Question 3:

Here, we want to write a Stan program to sample from posterior distribution described in part 1. Once we write that, we want to use that program to sample from posterior for one of the datasets in part 2. Finally, we want to plot a histogram of the samples overlaid with the analytically derived posterior.

Firstly, we write a Stan program to sample from the posterior distribution described in part 1.

```
data {
  int<lower=0> n;
  vector[n] y;
  real posterior_expectation;
}
transformed data {
  real m = 2;
  real<lower=0> s = 1/sqrt(n - sum(y));
}
parameters {
real mu; }
model {
mu ~ normal(m,s);
y ~ normal(mu, 1);
}
```

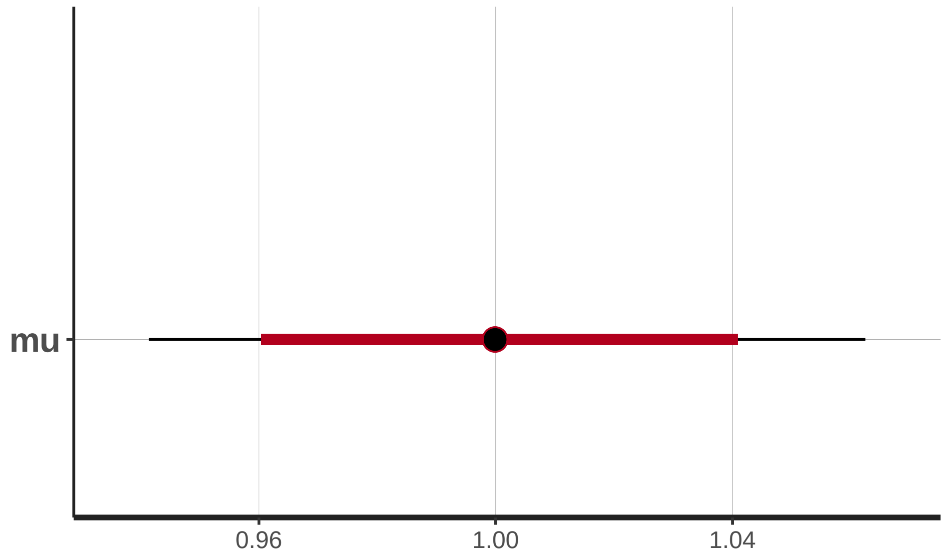
Now that we have written the Stan program, let us sample from one of the sample created in part 2. Here, we sample from dataset 9, which correponds to data with n=1000 and mean and standard deviation being 1 respectively.

Once we have the plot, let us observe the summary statistics of parameter estimates and sampler diagnostics and then see the plot of mean parameter estimate and confidence interval.

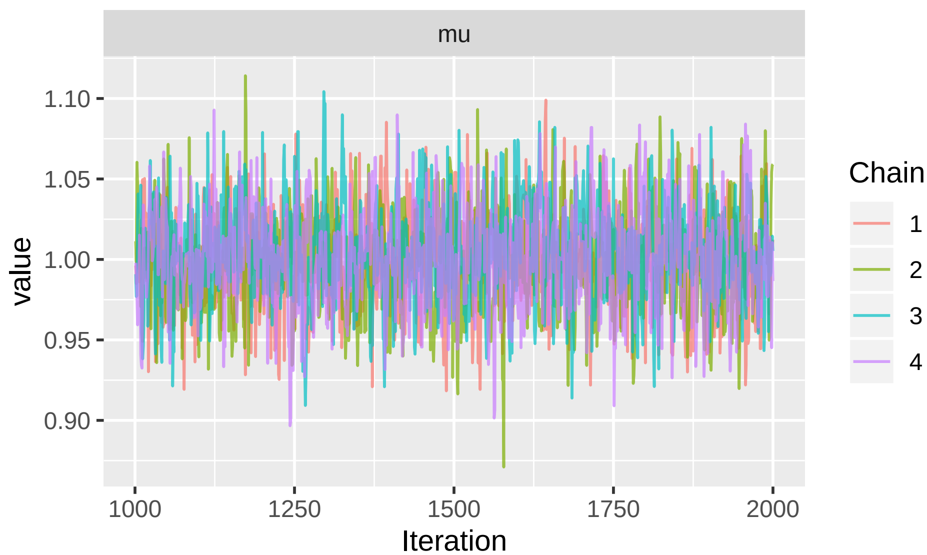
```
## Inference for Stan model: 0acda7b252cf51e114ace041170d3dc2.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##         mean se_mean   sd    2.5%     25%     50%     75%   97.5% n_eff Rhat
## mu      1.00    0.00 0.03    0.94    0.98    1.00    1.02    1.06  1484    1
## lp__ -500.34    0.02 0.68 -502.19 -500.51 -500.08 -499.90 -499.86  1936    1
##
## Samples were drawn using NUTS(diag_e) at Tue Jan 21 11:09:53 2020.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

```
## ci_level: 0.8 (80% intervals)
```

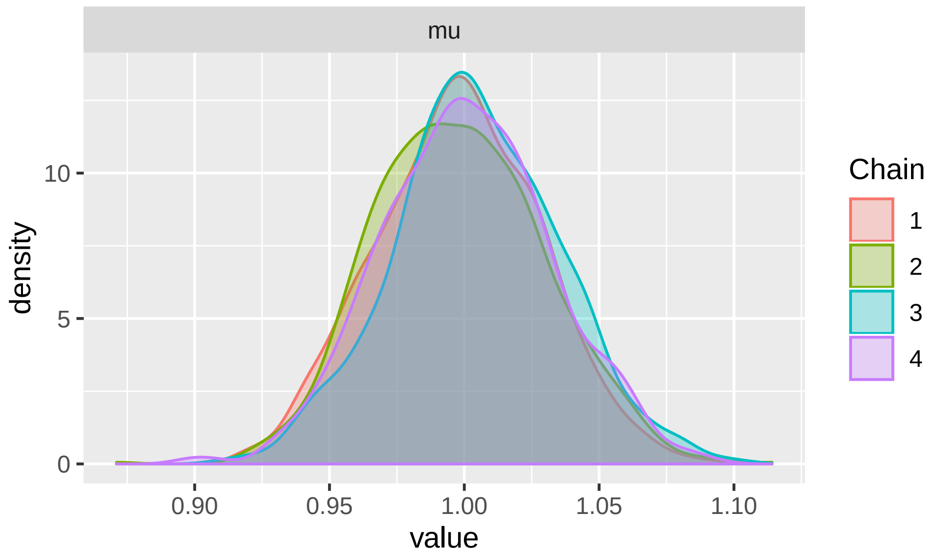
```
## outer_level: 0.95 (95% intervals)
```



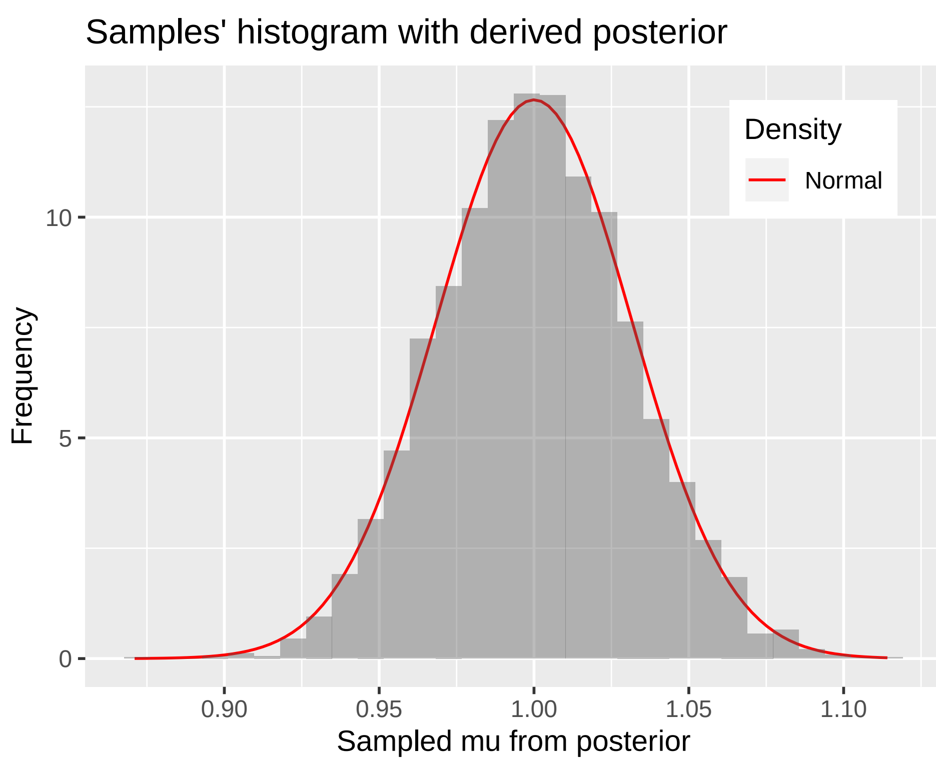
From the fit, we see that Rhat = 1, so we can say that there is convergence. To further confirm that, we now observe the traceplot of all chains. As the chains have converged, we have obtained the posterior distribution of our parameter.



Now that we have sampled from posterior, let us have a look at density of the parameter.



Now that we have that, we will plot the histogram of samples with analytically derived posterior.



Based on the above plot, we can say that the histogram of samples as well as the analytically derived posterior, both are quite similar distributionally and they seem to follow normal distribution centered around 1 which is something we expected. (This is becuase we allowed our priors to depend on our data)

Appendix:

```r
p1 <- qplot(d1_10p,dnorm(d1_10p, mean = 11 - sum(d1_10p), sd = 1),
            xlab = "sampled density for N(1,1)", ylab = "prior densities", main = "n=10")
p2 <- qplot(d2_10p,dnorm(d2_10p, mean = 11 - sum(d2_10p), sd = 1),
            xlab = "sampled density for N(0,1)", ylab = "prior densities", main = "n=10")
p3 <- qplot(d3_10p,dnorm(d3_10p, mean = 11 - sum(d3_10p), sd = 1),
            xlab = "sampled density for N(-1,1)",ylab = "prior densities", main = "n=10")
p4 <- qplot(d4_10p,dnorm(d4_10p, mean = 11 - sum(d4_10p), sd = 1),
            xlab = "sampled density for N(-10,1)",ylab = "prior densities", main = "n=10")
p5 <- qplot(d5_100p,dnorm(d5_100p, mean = 1, sd = 1),
            xlab = "sampled density for N(1,1)", ylab = "prior densities",
             main = "n=100")
p6 <- qplot(d6_100p,dnorm(d6_100p, mean = 101, sd = 1),
            xlab = "sampled density for N(0,1)", ylab = "prior densities",
             main = "n=100")
p7 <- qplot(d7_100p,dnorm(d7_100p, mean = 201, sd = 1),
            xlab = "sampled density for N(-1,1)", ylab = "prior densities",
             main = "n=100")
p8 <- qplot(d8_100p,dnorm(d8_100p, mean = 1101, sd = 1),
            xlab = "sampled density for N(-10,1)", ylab = "prior densities",
             main = "n=100")
p9 <- qplot(d9_1000p,dnorm(d9_1000p, mean = 1, sd = 1),
            xlab = "sampled density for N(1,1)", ylab = "prior densities",
             main = "n=1000")
p10 <- qplot(d10_1000p,dnorm(d10_1000p, mean = 1001, sd = 1),
             xlab = "sampled density for N(0,1)", ylab = "prior densities",
              main = "n=1000")
p11 <- qplot(d11_1000p,dnorm(d11_1000p, mean = 2001, sd = 1),
             xlab = "sampled density for N(-1,1)", ylab = "prior densities",
              main = "n=1000")
p12 <- qplot(d12_1000p,dnorm(d12_1000p, mean = 11001, sd = 1),
             xlab = "sampled density for N(-10,1)", ylab = "prior densities",
              main = "n=1000")

plot_grid(p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12, ncol=4, nrow = 3)

# Fixing standard deviation
s <- 1

# Finding mean for the priors with standard deviation fixed as 1

m1_10p <- 10 + 1 - sum(d1_10p)
m2_10p <- 10 + 1 - sum(d2_10p)
m3_10p <- 10 + 1 - sum(d3_10p)
m4_10p <- 10 + 1 - sum(d4_10p)
m5_100p <- 100 + 1 - sum(d5_100p)
m6_100p <- 100 + 1 - sum(d6_100p)
m7_100p <- 100 + 1 - sum(d7_100p)
m8_100p <- 100 + 1 - sum(d8_100p)
m9_1000p <- 1000 + 1 - sum(d9_1000p)
m10_1000p <- 1000 + 1 - sum(d10_1000p)
m11_1000p <- 1000 + 1 - sum(d11_1000p)
m12_1000p <- 1000 + 1 - sum(d12_1000p)
```

```r
plot1 <- ggplot(data.frame(x = c(-10, 10)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m1_10p, sd = 1)) +
scale_x_continuous(name = "mu",breaks = seq(-10, 10, 4), limits=c(-10, 10)) +
scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 10 and mu = 1")

plot2 <- ggplot(data.frame(x = c(5, 25)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m2_10p, sd = 1)) +
scale_x_continuous(name = "mu",breaks = seq(5, 25, 4), limits=c(5, 25)) +
scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 10 and mu = 0")

plot3 <- ggplot(data.frame(x = c(10,30)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m3_10p, sd = 1)) +
scale_x_continuous(name = "mu",breaks = seq(10, 30, 4), limits=c(10, 30)) +
scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 10 and mu = -1")

plot4 <- ggplot(data.frame(x = c(100, 120)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m4_10p, sd = 1)) +
scale_x_continuous(name = "mu",breaks = seq(100, 120, 4), limits=c(100, 120)) +
scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 10 and mu = -10")

plot5 <- ggplot(data.frame(x = c(-10, 10)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m5_100p, sd = 1)) +
scale_x_continuous(name = "mu",breaks = seq(-10, 10, 4),
                   limits=c(-10, 10)) + scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 100 and mu = 1")

plot6 <- ggplot(data.frame(x = c(85, 105)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m6_100p, sd = 1)) +
scale_x_continuous(name = "mu",breaks = seq(85, 105, 4), limits=c(85, 105)) +
  scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 100 and mu = 0")

plot7 <- ggplot(data.frame(x = c(205, 225)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m7_100p, sd = 1)) +
  scale_x_continuous(name = "mu",breaks = seq(205, 225, 4), limits=c(205, 225)) +
  scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 100 and mu = -1")

plot8 <- ggplot(data.frame(x = c(1100, 1120)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m8_100p, sd = 1)) +
  scale_x_continuous(name = "mu",breaks = seq(1100, 1120, 4), limits=c(1100, 1120)) +
  scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 100 and mu = -10")

plot9 <- ggplot(data.frame(x = c(-5, 15)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m9_1000p, sd = 1)) +
  scale_x_continuous(name = "mu",breaks = seq(-5, 15, 4), limits=c(-5, 15)) +
  scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 1000 and mu = 1")
```

```r
plot10 <- ggplot(data.frame(x = c(960, 980)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m10_1000p, sd = 1)) +
  scale_x_continuous(name = "mu",breaks = seq(960, 980, 4), limits=c(960, 980)) +
  scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 1000 and mu = 0")

plot11 <- ggplot(data.frame(x = c(1950, 1970)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m11_1000p, sd = 1)) +
  scale_x_continuous(name = "mu",breaks = seq(1950, 1970, 4), limits=c(1950, 1970)) +
  scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 1000 and mu = -1")

plot12 <- ggplot(data.frame(x = c(10995, 11015)), aes(x=x)) +
stat_function(fun = dnorm,args = list(mean = m12_1000p, sd = 1)) +
  scale_x_continuous(name = "mu",breaks = seq(10995, 11015, 4), limits=c(10995, 11015)) +
scale_y_continuous(name = "Density") +
ggtitle("Prior when n = 1000 and mu = -10")

figure <- plot_grid(plot1, plot2, plot3, plot4, plot5, plot6, plot7, plot8, plot9, plot10,
                    plot11, plot12, ncol = 4, nrow = 3)

figure

options(mc.cores = parallel::detectCores())

# sampling from posterior
y <- d9_1000p
data <- list(n = 1000, y = y, posterior_expectation = 1)
fit <- sampling(normal, data = data, chains = 4, iter = 2000)

# printing the summary statistics
print(fit)

# plotting mean parameter estimate and confidence interval
plot(fit)

# plotting traceplot of all chains
ggs_traceplot(ggs(fit))

# plotting density of parameter
ggs_density(ggs(fit))

# extracting values of sample
data_mat <- as.matrix(fit)
sampled_mu <- data_mat[1:4000,1]

# converting them to a data frame
data=data.frame(value=sampled_mu)

# finding true variance
s_2 <- 1/(1000-sum(d9_1000p))
variance_true <- s_2/(1000*s_2 + 1)
```

```r
#  plotting the histogram of samples with analytically derived posterior
qplot(sampled_mu, geom = 'blank', xlab ='Sampled mu from posterior', ylab='Frequency',
      main = "Histogram of samples overlaid with analytically derived posterior") +
   stat_function(fun = dnorm,args = list(mean = 1, sd = sqrt(variance_true)),
                 aes(colour = "Normal")) +
    geom_histogram(aes(y = ..density..), alpha = 0.4, bins = 30) +
      scale_colour_manual(name = 'Density', values = c('red')) +
        theme(legend.position = c(0.85, 0.85))
```