# Homework 3

Jessica Nguyen and Tristan Chen

**Due on February 13, 2022 at 11:59 pm**

**Note:** If you are working with a partner, please submit only one homework per group with both names and whether you are taking the course for graduate credit or not. Submit your Rmarkdown (.Rmd) and the compiled pdf on Gauchospace.

**1. Warmup: Posterior Predictive Distributions**

a. What is a posterior predictive distribution (i.e., what does it give probabilities for)? How is this different from the posterior distribution of a parameter?

The Posterior Predictive Distribution describes our uncertainty about a new observation after seeing n observations. The posterior predictive distribution refers to the distribution of future observations of data, unlike the posterior distribution which refers to the distribution of a parameter.

b. Is a posterior predictive model conditional on just the data, just the parameter, or on both the data and the parameter?

The predictive distribution does not depend on any parameters, only on the data.

c. Why do we need posterior predictive distributions? For example, if we wanted to predict new values of $Y$, why couldn't we just use the posterior mean of the parameter?

If we only use the posterior mean, we are not taking into account the different values of the parameter. We need posterior predictive distributions because with posterior predictive distributions, we're taking into account that the parameter can take a wide range of values.

**2. Posterior Credible Intervals.**

One way for us to learn more about posterior distributions is to find some credible intervals. Suppose we have a posterior density of a parameter $\lambda$, defined by $\lambda|y \sim \text{Gamma}(4, 1)$.

a. Plot this distribution. Construct the posterior middle 95% credible interval for $\lambda$. Save the lower and upper endpoints in a vector of length 2, called `middle_95`. Using either line segments or shading, display this interval on the plot.

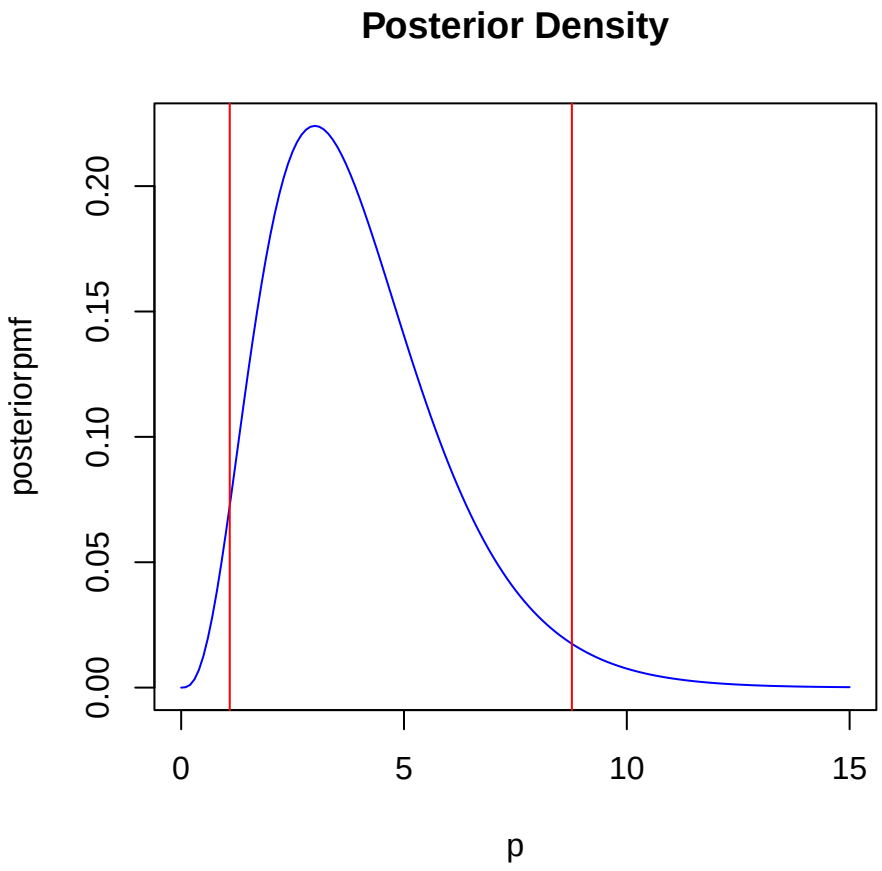
```
# create interval - middle_95 should be a vector containing the two endpoints of the interval, in order
middle_95 <- qgamma(c(0.025, 0.975), shape = 4, rate = 1)
middle_95
```

```
## [1] 1.089865 8.767273
```

```
. = ottr::check("tests/q2a1.R")
```

```
# make the plot
# YOUR CODE HERE
p=seq(0,15,.1)
posteriorpmf=dgamma(p,shape=4, rate=1)

plot(p, posteriorpmf,
    main="Posterior Density", col='blue', type="l")
```

```
abline(v = middle_95[1], col = 'red')
abline(v = middle_95[2], col = 'red')
```




b. Interpret the interval you found. How is it different from a frequentist confidence interval?

There is a 95% probability that the true estimate will be between 1.089865 and 8.767273. The difference between this and a frequentist confidence interval is that frequentist intervals are random. Frequentists intervals are random because Y is random, however for us the parameter is the random variable and y is observed, so the intervals are not random.

c. Besides the middle 95% credible interval, we could also find the 95% highest posterior density (HPD) region. This region contains the 95% of posterior values with the highest posterior densities. The HPD region will always be the shortest credible interval for a given probability, since it by definition contains the values of $\lambda$ with the highest probability of occurring. Use `HDInterval::hdi()` to construct the HPD region. Save the lower and upper endpoints of this region in a variable called `hdi_region`. Add this interval to the plot you made in part (a), making sure that both intervals are distinguishable on the plot.

```
# create interval
set.seed(9)
hd_region <- HDInterval::hdi(rgamma(p, shape = 4, rate = 1), ci = 0.95)
hd_region
```

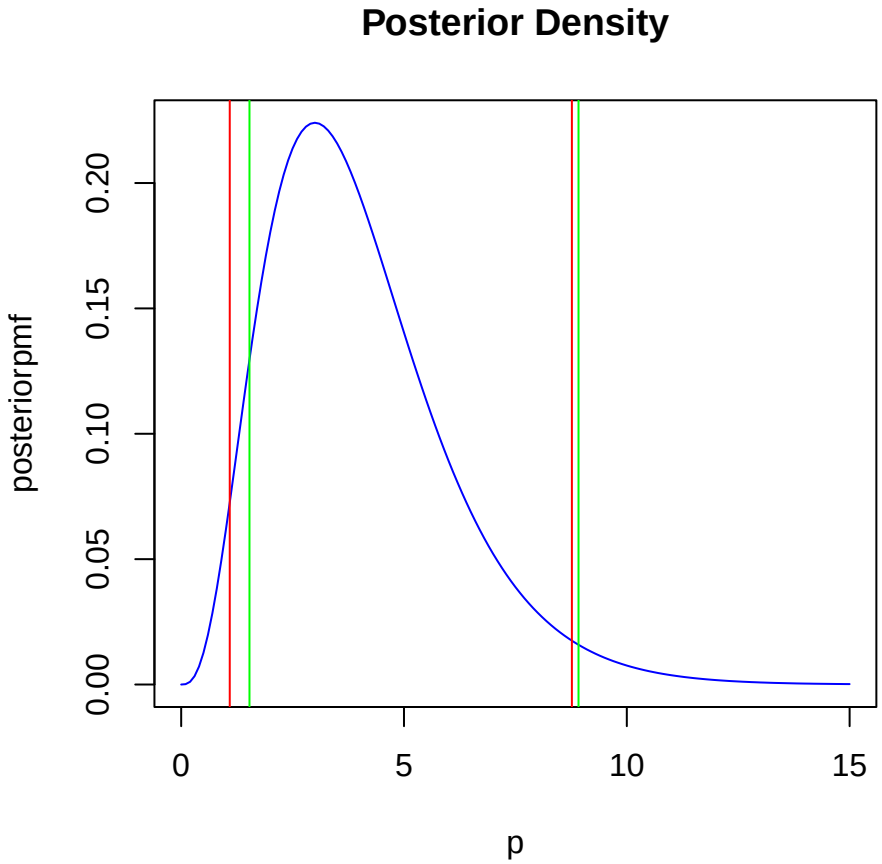
```
##    lower    upper
## 1.532638 8.916484
```

```
## attr(,"credMass")
## [1] 0.95
```

```
. = ottr::check("tests/q2c1.R")
```

```
##
## All tests passed!
```

```
# make the plot
# YOUR CODE HERE
plot(p, posteriorpmf,
    main="Posterior Density", col='blue', type="l")
abline(v = middle_95[1], col = 'red')
abline(v = middle_95[2], col = 'red')
abline(v = hd_region[1], col = 'green')
abline(v = hd_region[2], col = 'green')
```




d. Based on your plot, how do the two kinds of 95% credible intervals differ? How long is the middle interval? The HDI interval?

The two kinds of 95% credible intervals differ in which the HDI interval is shorter. The interval for the HPD is [1.532638, 8.916484] while the other interval is [1.089865, 8.767273]. It is clear from looking at the actual interval and the plot that the HDI interval is shorter than the other credible interval, which is how it is supposed to be.

**3. Cancer Research in Laboratory Mice**

As a reminder from homework 2, a laboratory is estimating the rate of tumorigenesis (the formation of tumors) in two strains of mice, A and B. They have tumor count data for 10 mice in strain A and 13 mice in strain B. Type A mice have been well studied, and information from other laboratories suggests that type A mice have tumor counts that are approximately Poisson-distributed. Tumor count rates for type B mice are unknown, but type B mice are related to type A mice. Assuming a Poisson sampling distribution for each group with rates $\theta_A$ and $\theta_B$. We assume $\theta_A \sim \text{gamma}(120, 10)$ and $\theta_B \sim \text{gamma}(12, 1)$. We observe $y_A = (12, 9, 12, 14, 13, 13, 15, 8, 15, 6)$ and $y_B = (11, 11, 10, 9, 9, 8, 7, 10, 6, 8, 8, 9, 7)$. Now we will actually investigate evidence that Type A mice are have higher rates of tumor formation than Type B mice.

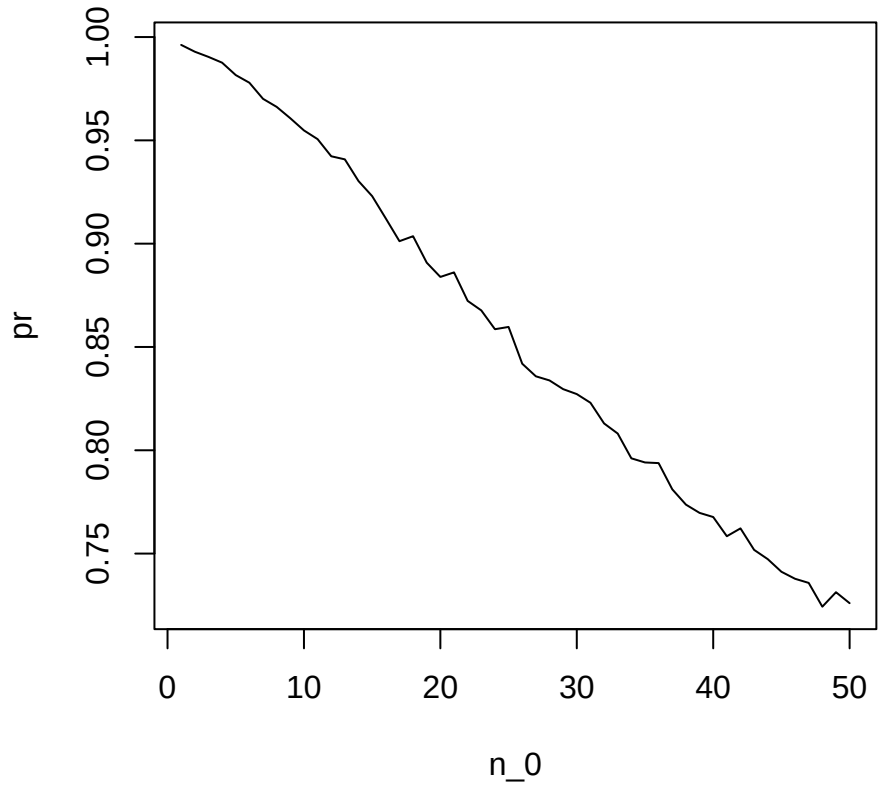
a. For $n_0 \in \{1, 2, ..., 50\}$, obtain $Pr(\theta_B < \theta_A \mid y_A, y_B)$ via Monte Carlo sampling for $\theta_A \sim \text{gamma}(120, 10)$ and $\theta_B \sim \text{gamma}(12 \times n_0, n_0)$. Make a line plot of $Pr(\theta_B < \theta_A \mid y_A, y_B)$ vs $n_0$. Describe how sensitive the conclusions about the event $\{\theta_B < \theta_A\}$ are to the prior distribution on $\theta_B$.

```
y_A <- c(12, 9, 12, 14, 13, 13, 15, 8, 15, 6)
y_B <- c(11, 11, 10, 9, 9, 8, 7, 10, 6, 8, 8, 9, 7)

# store your probabilities in a vector called "pr" for testing.
n_0 <- rep(1:50)
pr <- c()

for (n in n_0)
{
  theta_A <- rgamma(10000, 120 + sum(y_A), 10 + length(y_A))
  theta_B <- rgamma(10000, 12 * n + sum(y_B), n + length(y_B))
  pr[n] <- mean(theta_B<theta_A)
  }
```

```
. = ottr::check("tests/q3a1.R")
```

```
##
## All tests passed!
```

```
# create the plot
plot(n_0, pr, type = 'l')
```




The conclusions about $\theta_B < \theta_A$ are sensitive to the prior distribution on $\theta_B$. When n is 0, the probability that $\theta_B < \theta_A$ is almost 1. As n increases, the probability decreases in general.

b. Repeat the previous part replacing the event $\{\theta_B < \theta_A\}$ with the event $\{\tilde{Y}_B < \tilde{Y}_A\}$, where $\tilde{Y}_A$ and $\tilde{Y}_B$ are samples from the posterior predictive distribution.

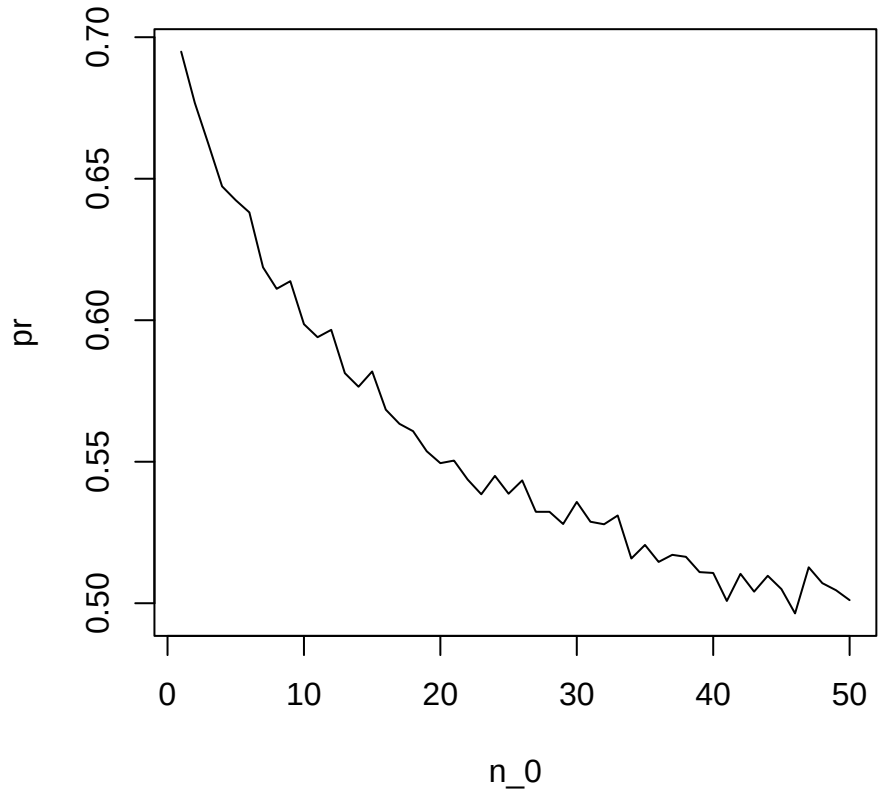
```
y_A = c(12, 9, 12, 14, 13, 13, 15, 8, 15, 6)
y_B = c(11, 11, 10, 9, 9, 8, 7, 10, 6, 8, 8, 9, 7)

# store your probabilities in a vector called "pr" for testing.
n_0 <- rep(1:50)
pr <- c()

for (n in n_0)
{
  theta_A <- rgamma(10000, 120 + sum(y_A), 10 + length(y_A))
  theta_B <- rgamma(10000, 12 * n + sum(y_B), n + length(y_B))
  ytilde_A <- rpois(10000, theta_A)
  ytilde_B <- rpois(10000, theta_B)
  pr[n] <- mean(ytilde_B < ytilde_A)
}
```

```
. = ottr::check("tests/q3b1.R")
```

```
##
```

```
## All tests passed!
```

```
# create the plot
plot(n_0, pr, type = 'l')
```




The conclusions about $\tilde{Y}_B < \tilde{Y}_A$ are also sensitive to the prior distribution on $\theta_B$. As n increases from 0 to 50, $Pr(\tilde{Y}_B < \tilde{Y}_A \mid y_A, y_B)$ decreases from around 0.7 to 0.5.

c. In the context of this problem, describe the meaning of the events $\{\theta_B < \theta_A\}$ and $\{\tilde{Y}_B < \tilde{Y}_A\}$. How are they different?

$\theta_B < \theta_A$ is examining the rate given $Y_B$ is less than the rate given $Y_A$. $\tilde{Y}_B < \tilde{Y}_A$ is examining the replicated data from using $\theta_B$ is less than the replicated data from using $\theta_A$.

**4. Posterior Predictive Model Checking**

Model checking and refinement is an essential part of Bayesian data analysis. Let's investigate the adequacy of the Poisson model for the tumor count data. Consider strain A mice only for now, and generate posterior predictive datasets $y_A^{(1)}, ..., y_A^{(1000)}$. Each $y_A^{(s)}$ is a sample of size $n_A = 10$ from the Poisson distribution with parameter $\theta_A^{(s)}$, $\theta_A^{(s)}$ is itself a sample from the posterior distribution $p(\theta_A \mid y_A)$ and $y_A$ is the observed data. For each $s$, let $t^{(s)}$ be the sample average divided by the sample variance of $y_A^{(s)}$.

a. If the Poisson model was a reasonable one, what would a "typical" value $t^{(s)}$ be? Why?

```
y_A = c(12, 9, 12, 14, 13, 13, 15, 8, 15, 6)
```

```r
# generate posterior predictive datasets and find test statistic for each one
# store your test statistics in a vector called "t" for testing.

# YOUR CODE HERE
set.seed(555)
t <- rep(1, 1000)

for (i in rep(1:1000))
{
  theta_A <- rgamma(10000, 120 + sum(y_A), 10 + length(y_A))
  ytilde_A <- rpois(10, theta_A)
  test_stat <- mean(ytilde_A) / var(ytilde_A)
  t[i] <- test_stat
}
mean(t)
```

```
## [1] 1.247114
```

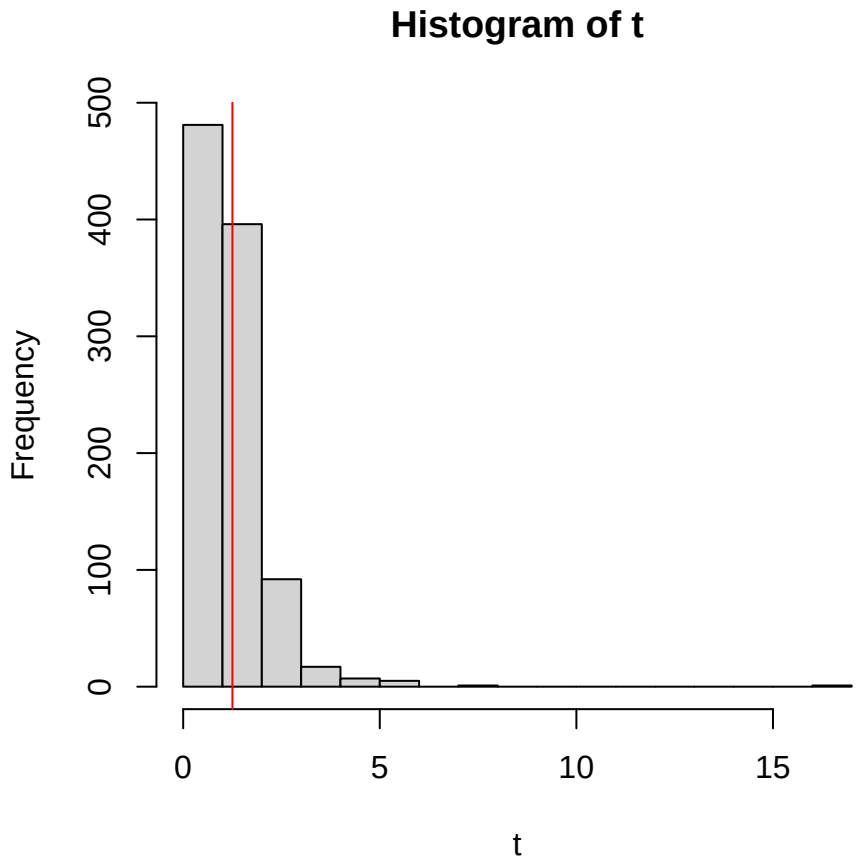
```r
. = ottr::check("tests/q4a1.R")
```

```
##
## All tests passed!
```

A typical value of $t^{(s)}$ should be 1 considering the mean and variance of a poisson distribution should be the same value. When we ran 1000 tests, our average for $t^{(s)}$ was about 1.25, which is close to the value of 1.

b. In any given experiment, the realized value of $t^s$ will not be exactly the "typical value" due to sampling variability. Make a histogram of $t^{(s)}$ and compare to the observed value of this statistic, $\frac{\text{mean}(y_A)}{\text{var}(y_A)}$. Based on this statistic, make a comment on if the Poisson model seems reasonable for these data (at least by this one metric).

```r
# create the histogram, adding a vertical line at the observed value of the test statistic
# YOUR CODE HERE
# create the histogram, adding a vertical line at the observed value of the test statistic
hist(t, breaks = 20)
abline(v = mean(y_A) / var(y_A), col = "red")
```




It appears that the Poisson model is reasonable for these data, as the observed test statistic is close to the majority of the posterior predictive dataset test statistic values.

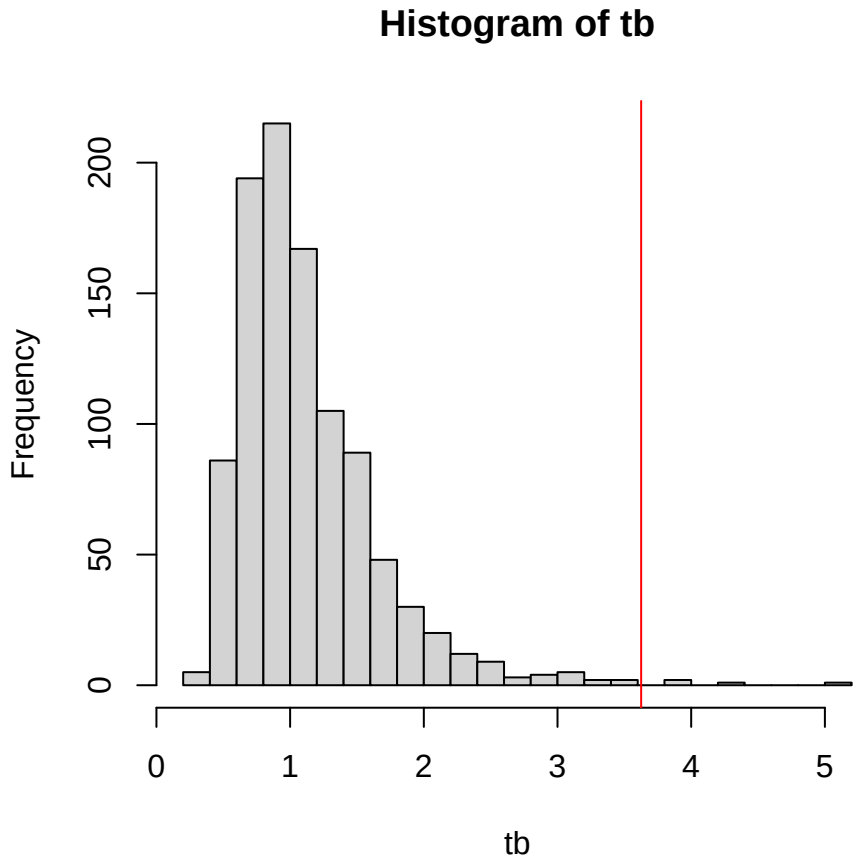
c. Repeat the part b) above for strain B mice, using $Y_B$ and $n_B = 13$ to generate the samples. Assume the prior distribution $p(\theta_B) \sim \text{Gamma}(12, 1)$. Again make a comment on the Poisson model fit.

```
y_B = c(11, 11, 10, 9, 9, 8, 7, 10, 6, 8, 8, 9, 7)

# generate posterior predictive datasets and find test statistic for each one
# store your test statistics in a vector called "tb" for testing

# YOUR CODE HERE
n_B <- 13
tb <- rep(1, 1000)

for (i in rep(1:1000))
{
  theta_B <- rgamma(10000, 12 + sum(y_B), 1 + length(y_B))
  ytilde_B <- rpois(n_B, theta_B)
  test_stat <- mean(ytilde_B) / var(ytilde_B)
  tb[i] <- test_stat
}
```

```
. = ottr::check("tests/q4c1.R")
```

```
##
## All tests passed!
```

```
# create the histogram, adding a vertical line at the observed value of the test statistic
# YOUR CODE HERE
hist(tb, breaks = 20)
abline(v = mean(y_B) / var(y_B), col = "red")
```




It appears that the Poisson model is not a good fit for the data, as the observed test statistic is not close to the majority of the posterior predictive dataset test statistics.