Assignment 2

anonymous

2023-09-13

General information

I used AI to search some related information (How to calculate the likelihood the posterior and so on) in this assignment.

(a)

1.The likelihood function is:$p(y|\pi) = \binom{n}{y}\pi^{y}(1-\pi)^{n-y}$ So we can put our data in: $p(44|\pi) = \binom{274}{44}\pi^{44}(1-\pi)^{274-44}$

where$\binom{274}{44} = \frac{274!}{44!(274-44)!}$

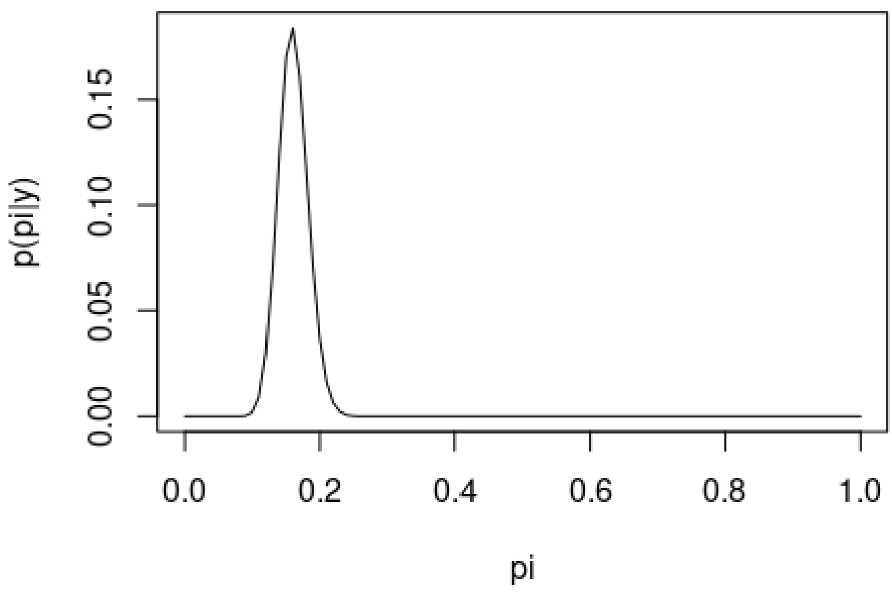
2.The prior is a beta distribution Beta(2,10), the probability density function of the beta distribution is: $p(\pi) = \frac{\pi^{2-1}(1-\pi)^{10-1}}{B(2,10)}$

where$B(\alpha,\beta) = \int_0^1 t^{\alpha-1}(1-t)^{\beta-1}dt$

3.The probability density function (pdf) for this beta distribution is:$p(\pi|y) = \frac{\pi^{2+y-1}(1-\pi)^{10+n-y-1}}{B(2+y,10+n-y)}$

```
posterior_alpha = 2
posterior_beta = 10
# These are not the actual values for the posterior!
# You will have to compute those from the data!
#posterior_alpha = 2
#posterior_beta = 10
count_algae_present<-sum(algae==1)
n <- 274  # number of monitoring station
y <- count_algae_present  # sucessful obversation time
#PI <- count_algae_present/274
PI <- seq(0, 1, by = 0.01)
# likelihood
likelihood <- function(y, n, PI) {
  dbinom(y, n, PI)
}
result_likelihood <- likelihood(y, n, PI)
```

```r
#cat("liklihood:",result_likelihood,"\n")
#prior
prior <- function(PI) {
  dbeta(PI, prior_alpha, prior_alpha)
}
result_prior_PI <- prior(PI)
#cat("prior p(PI):",result_prior_PI,"\n")

#posterior
posterior_alpha <- prior_alpha + y
posterior_beta <- prior_beta + n - y
posterior <- function(pi, posterior_beta, posterior_alpha) {
  return(dbeta(pi, posterior_alpha, posterior_beta))
}
#cat("posterior p(PI|y):",posterior(PI,posterior_beta,posterior_alpha),
"\n")

#test
# define pi range
pi_range <- seq(0, 1, by = 0.01)
# likelihood
likelihood <- dbinom(y, size = n, prob = pi_range)
# prior
prior <- dbeta(pi_range, 2, 10)
# posterior
unnormalized_posterior <- likelihood * prior
# normalized posterior
posterior <- unnormalized_posterior / sum(unnormalized_posterior)
plot(pi_range, posterior, type = "l", ylab = "p(pi|y)", xlab = "pi")
```



```
cat("Beta(",posterior_alpha,",",posterior_beta,")","\n")

## Beta(46,240)
```

(b)

```
# Useful function: qbeta()

beta_point_est <- function(prior_alpha, prior_beta, data) {
    # Do computation here, and return as below.
    alpha_prime <- prior_alpha + sum(data==1)
    beta_prime <- prior_beta + length(data) - sum(data==1)
    E_pi_given_y <- alpha_prime / (alpha_prime + beta_prime)
    return(E_pi_given_y)
}

#test
cat("The value of E(PI|y):",(beta_point_est(prior_alpha, prior_beta, al
gae)),"\n")

## The value of E(PI|y): 0.1608392

beta_interval <- function(prior_alpha, prior_beta, data, prob=0.9) {
   lower_bound <- qbeta(0.05, prior_alpha + sum(data==1), prior_beta +
length(data) - sum(data==1)) # 5th   percentile
   upper_bound <- qbeta(0.95, prior_alpha + sum(data==1), prior_beta +
length(data) - sum(data==1)) # 95th percentile
   return(c(lower_bound,upper_bound))
}

#test
cat("90% posterior interval:",beta_interval(prior_alpha, prior_beta, al
gae, prob=0.9),"\n")
```

```
## The value of E(PI|y): 0.1608392
```

```
## 90% posterior interval: 0.1265607 0.1978177
```

The prior is Beta(2,10) Beta(2,10). This implies that, before seeing any actual data, our belief is that it is more likely for the probability PI to be closer to 0. The mean of this distribution is $2/(10 + 2)$, suggesting that we initially believe there's roughly a 16.67% chance of a site having detectable blue-green algae levels. From the example, y=44 sites with detectable levels out of n=274 total sites. This is an observed rate of 44/274.

(c)

```
# Useful function: pbeta()

beta_low <- function(prior_alpha, prior_beta, data, pi_0=0.2) {
    # Do computation here, and return as below.
    # This is the correct return value for the test data provided above,
    # combined with the correct prior.
    num_station <- length(data)  # The number of monitoring stations
    x <- sum(data==1)     # Number of monitoring stations available to
detect algae levels
    p_value <- pbeta(pi_0, x + prior_alpha, num_station - x + prior_bet
a)
    return(p_value)
}
cat("probability that the proportion of monitoring sites with detectabl
e algae levels PI is smaller than PI_0=0.2: ",beta_low(prior_alpha, pri
or_beta, algae,pi_0=0.2))
```

```
## probability that the proportion of monitoring sites with detectable
algae levels PI is smaller than PI_0=0.2:  0.9586136
```

(d)

1.Binomial Distribution Result: The data we observe should come from what's called a "binomial process." This just means that for every observation or test we make, there are only two possible outcomes (like success/failure, 1/0, detectable/not detectable, etc.). Plus, each time we test or observe, it doesn't depend on or affect any other tests.

2.Set Number of Tests: We should decide in advance how many times we're going to test or observe something.

3.Consistent Success Rate: The chance of success, represented by ‘p’, should always stay the same, no matter how many times we test.

4.Beta Prior: We use something called the “Beta distribution” when we want to make guesses based on prior knowledge or beliefs about certain proportions. This is handy for data that falls between 0 and 1 (like percentages or probabilities).

5.What We Already Believe: The shape of the Beta distribution is determined by two parameters, α and β. These parameters reflect our prior knowledge or what we already believe. If we have no strong feelings or guesses about what the outcome might be, we can set both α and β to 1. This means we’re equally open to all possibilities between 0 and 1. But if we have strong historical data or beliefs, we can adjust these values accordingly.

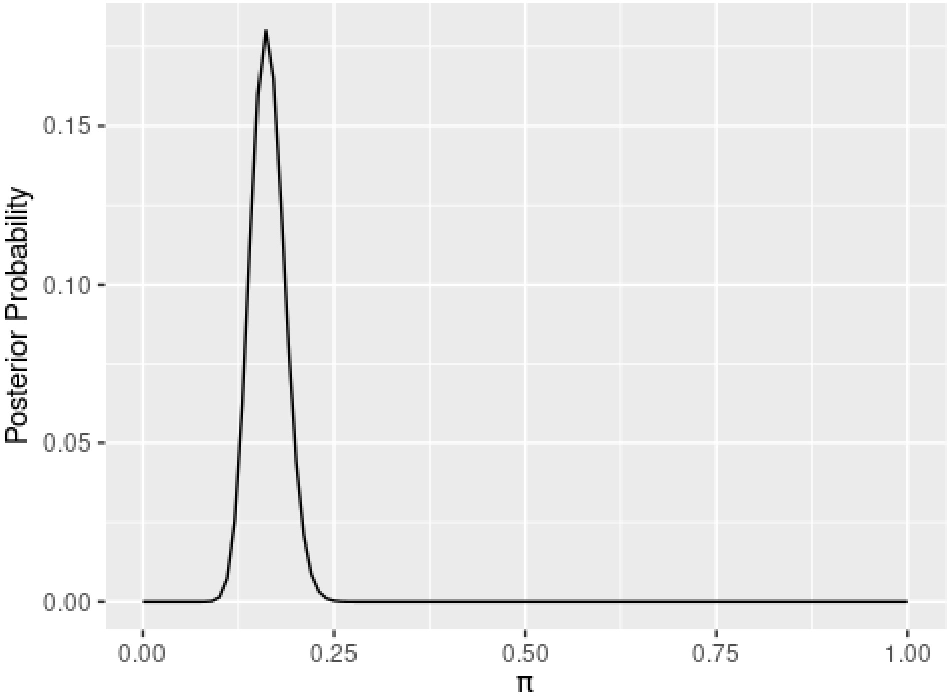
6.Likelihood and Prior Independence: We assume that our observed data which is modeled by the binomial distribution doesn’t depend on our prior beliefs. In simple terms, the way we gather our data is separate from our previous beliefs or expectations.

(e)

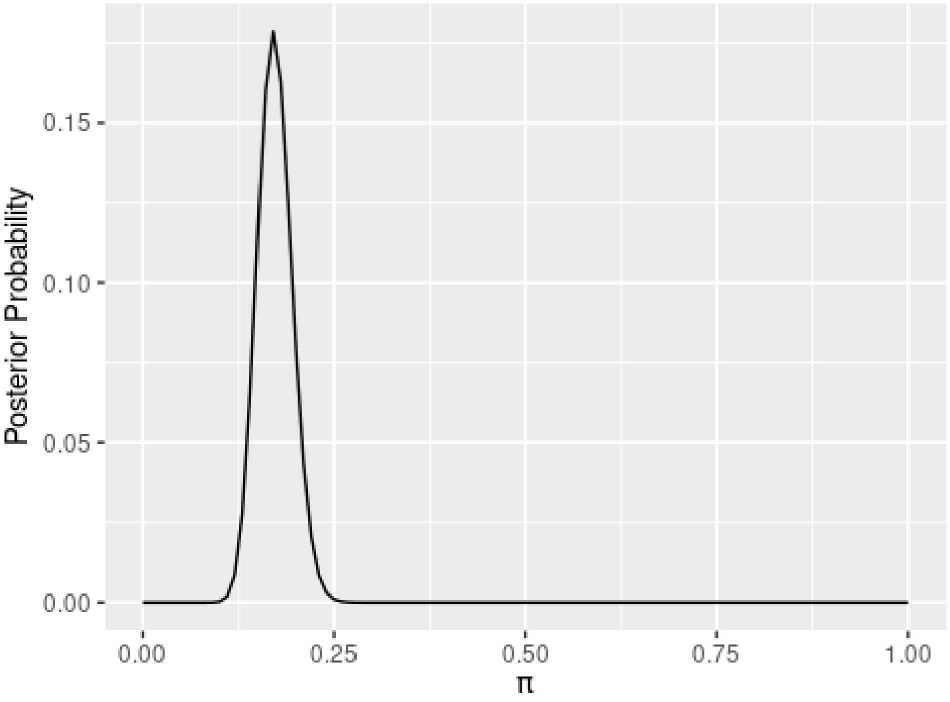
I use the code below to conduct a Bayesian inference sensitivity analysis on various prior distributions. It accomplishes this by defining observed data, creating a range of possible success probabilities, and developing a function to calculate posterior distributions based on different prior distribution parameters. Then I tests different combinations of prior distributions, calculates their posterior distributions, and generates plots to compare them.

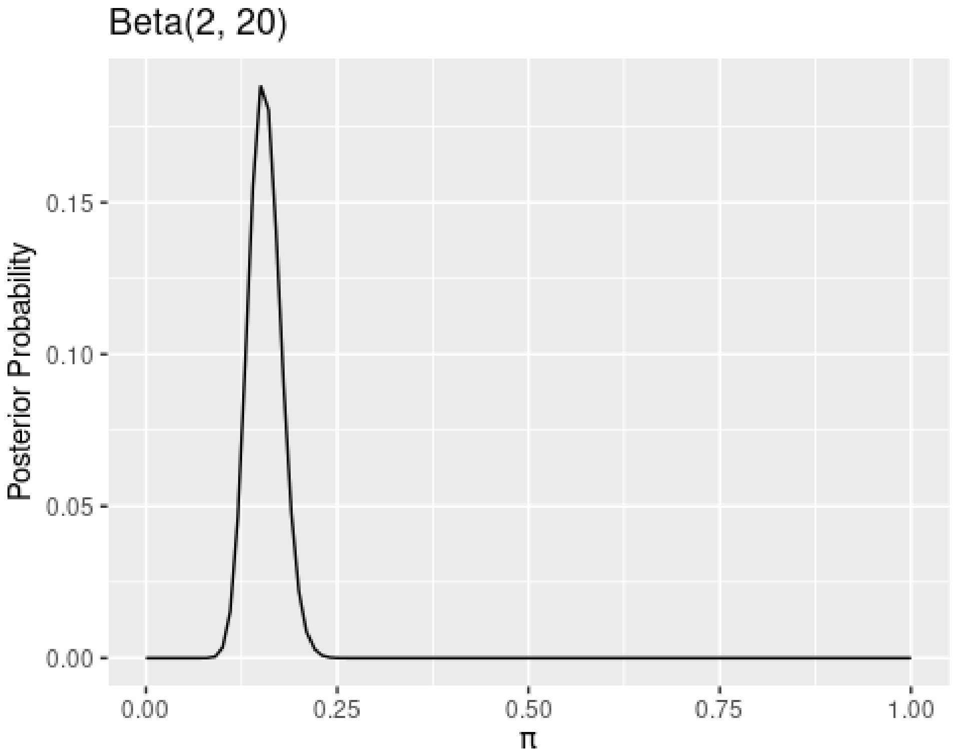
```r
library(ggplot2)

# Define the observed data
y <- 44 # Number of successes (sites with detectable algae)
n <- 274 # Total number of monitoring sites

pi_values <- seq(0, 1, by = 0.01)

# Create a function to calculate the posterior given a specific prior
calculate_posterior <- function(alpha, beta) {
  prior <- dbeta(pi_values, shape1 = alpha, shape2 = beta)
  likelihood <- dbinom(y, size = n, prob = pi_values)
  unnormalized_posterior <- likelihood * prior
  posterior <- unnormalized_posterior / sum(unnormalized_posterior)
  return(posterior)
}

# Different prior combinations to test
prior_combinations <- list(
  c(1, 1),
  c(5, 5),
  c(2, 20),
  c(100,10)
)

# Plot and compare posterior distributions
plot_titles <- c("Beta(1, 1)", "Beta(5, 5)", "Beta(2, 20)","Beta(100,10)
")

for (i in 1:length(prior_combinations)) {
  alpha <- prior_combinations[[i]][1]
  beta <- prior_combinations[[i]][2]
  posterior <- calculate_posterior(alpha, beta)

  # Create a plot
  plot_data <- data.frame(pi = pi_values, posterior = posterior)
  p <- ggplot(plot_data, aes(x = pi, y = posterior)) +
    geom_line() +
    labs(title = plot_titles[i], x = expression(pi), y = "Posterior Pro
bability")
  # Display the plot
  print(p)
}
```

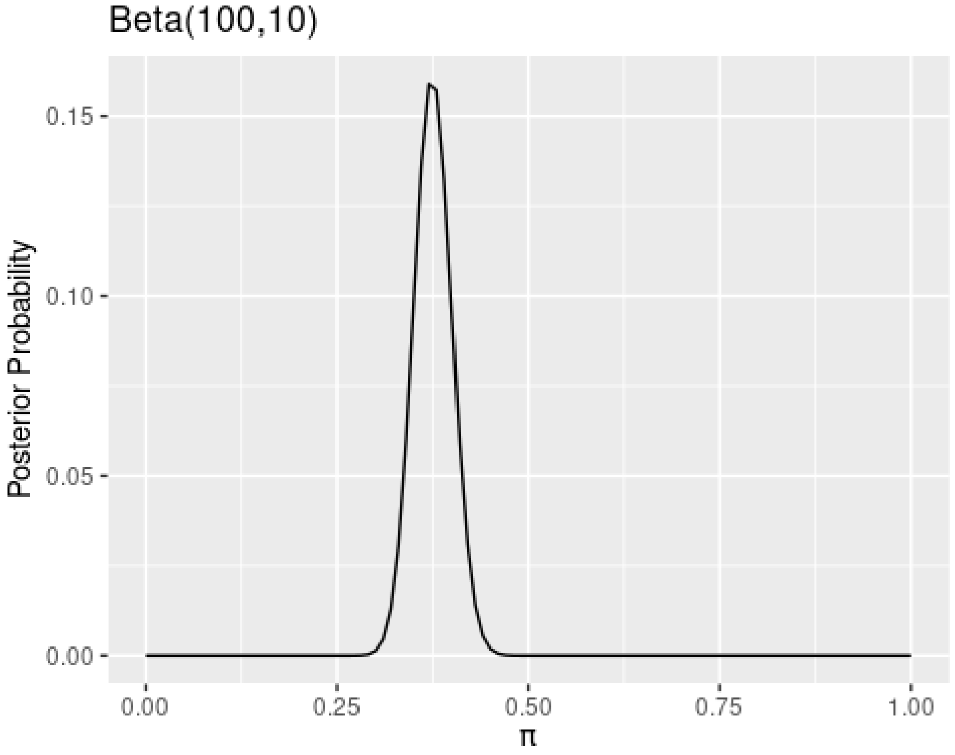
Beta(1, 1)



Beta(5, 5)







Beta(100, 10) and Beta(2, 20) priors result in more peaked posterior distributions, indicating a higher degree of certainty about the parameter PI. On the other hand, Beta(1, 1) and Beta(5, 5) priors lead to flatter posterior distributions, suggesting more uncertainty about PI.