

Lab Report:Lab1-Group8

Udaya Shanker Mohanan Nair(udamo524), Uday Jain(udaja983)

2025-04-21

Assignment 1 - Daniel Bernoulli

Assume, let $y_1, y_2, \dots, y_n \mid \theta \sim \text{Bern}(\theta)$ and we obtained a sample with failures, $f = 35$ in total of 78 trials. Given that α_0 and β_0 are 7.

Part A

The posterior distribution is $\text{Beta}(\alpha_0 + s, \beta_0 + f)$, where successes, $s = n - f = 43$ of 10000 random values.

```
set.seed(123)
f <- 35
n <- 78
alpha_zero <- 7
beta_zero <- 7

s <- n - f

posterior_alpha <- alpha_zero + s
posterior_beta <- beta_zero + f

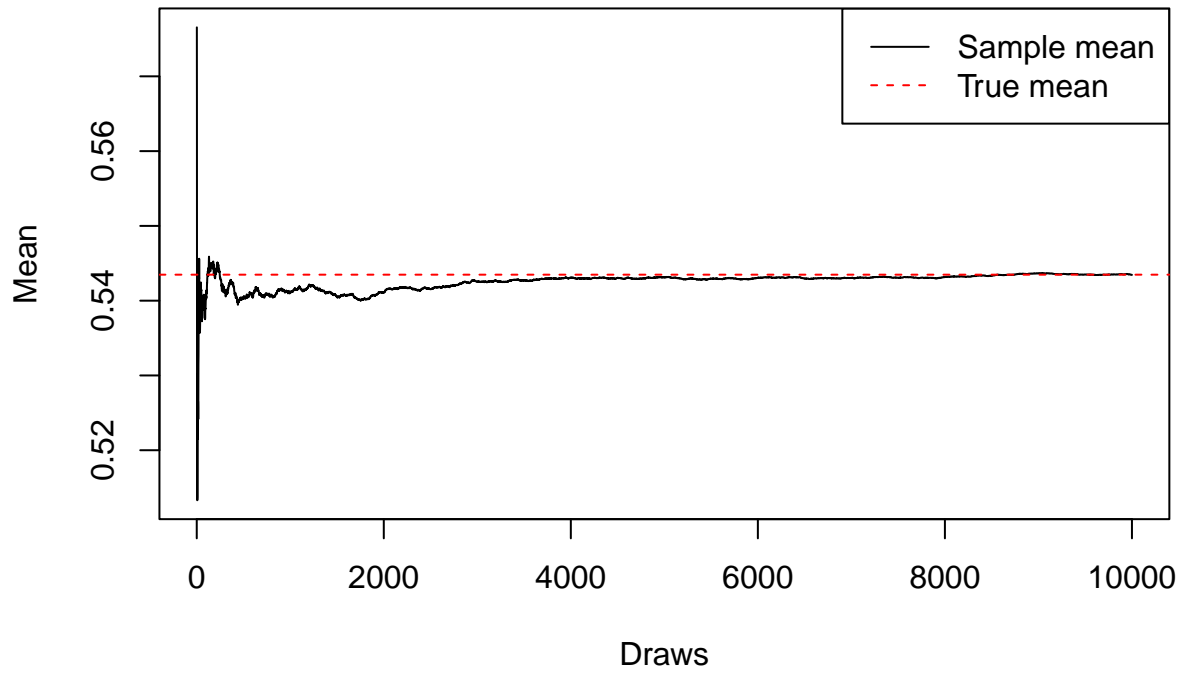
true_mean <- posterior_alpha / (posterior_alpha + posterior_beta)
true_var <- (posterior_alpha * posterior_beta) / ((posterior_alpha + posterior_beta)^2
                                                    * (posterior_alpha + posterior_beta + 1))
true_sd <- sqrt(true_var)

nDraw <- 10000
samples <- rbeta(nDraw, posterior_alpha, posterior_beta)

means <- cumsum(samples) / (1:nDraw)
vars <- cumsum((samples - means)^2) / (1:nDraw)
sds <- sqrt(vars)
```

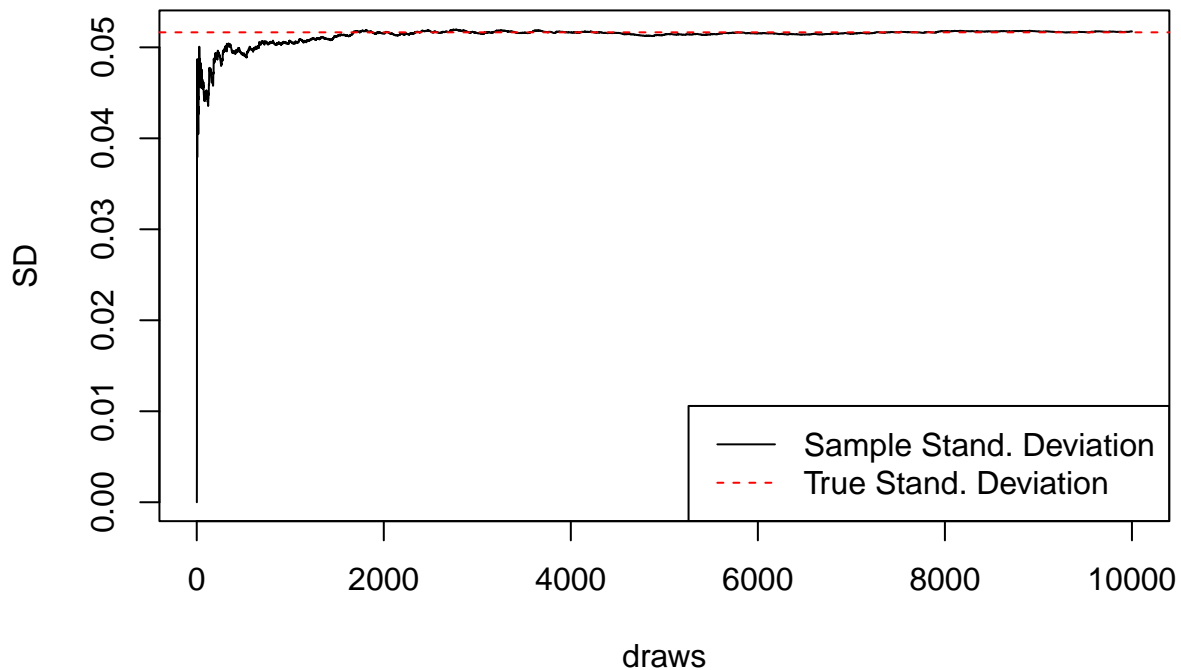
Now, we are going to plot the values of means and standard deviation.

Convergence of Posterior Mean



From the above graph, it is very clear that for the first 500 samples, mean value fluctuates (between 0.52 and 0.56), afterwards it converges to 0.5434783 as the number of Draws increases and which is the actual $\text{mean}(\text{true mean})$.

Convergence of Posterior Standard Deviation



From the above graph, it is very clear that for the first 1000 samples, standard deviations fluctuates (between 0 and 0.05), afterwards it converges to 0.05165119 as the number of Draws increases and which is the actual standard deviation(true sd).

Part B

Here, we calculated the exact and actual probabilities of the selected 10000 samples generated randomly having value greater than 0.5.

```
actual_probability <- mean(samples > 0.5)

exact_probability <- 1 - pbeta(0.5, posterior_alpha, posterior_beta)

cat("Actual probability ", actual_probability, "\n")
```

```
## Actual probability 0.7954
```

```
cat("Exact probability ", exact_probability, "\n")
```

```
## Exact probability 0.7990936
```

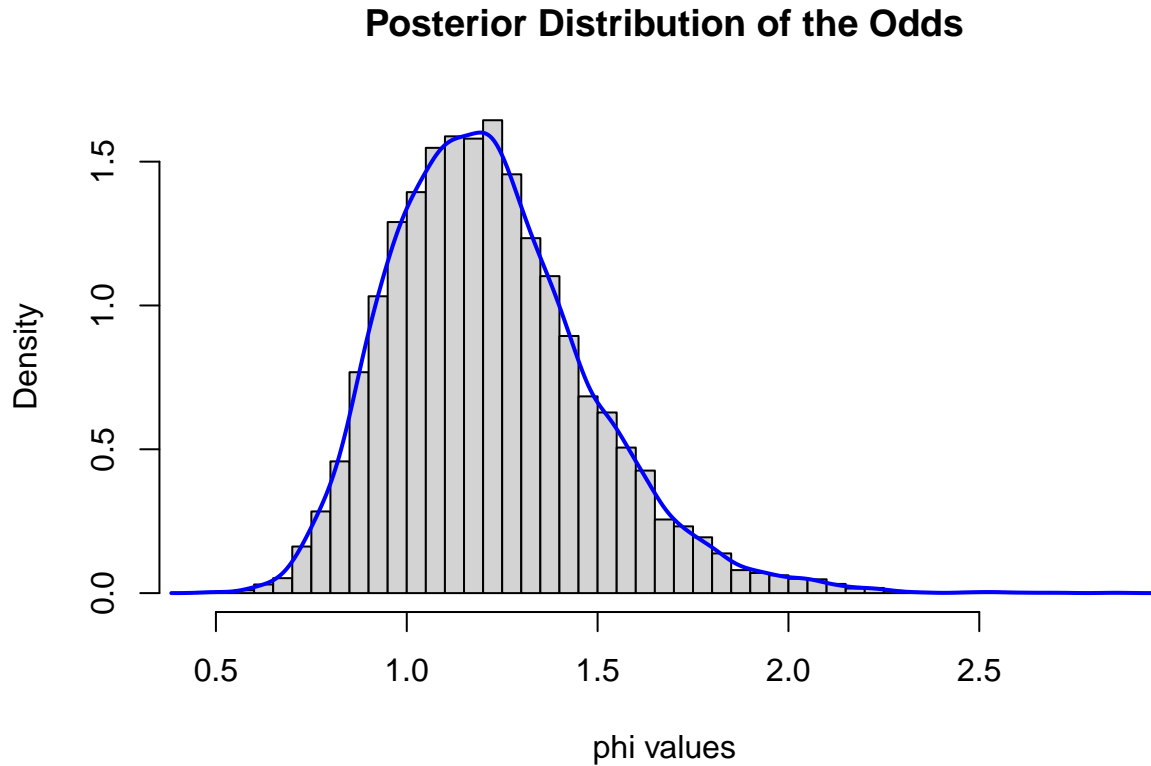
So both actual and exact probabilities are exact close to each other with some minute difference between 0.004.

Part C

Now drawing 10000 random values for generating posterior distribution of the odds $\phi = \frac{\theta}{1-\theta}$.

```
phi_values <- samples / (1 - samples)
```

Plotting the histogram of the phi values.



Assignment 2 : Log-normal distribution and the Gini coefficient

Given monthly incomes of 8 randomly selected persons (in thousands Swedish Krona) : 22, 33, 31, 49, 65, 78, 17, 24. Now we modelled the incomes using a log-normal distribution.

The log-normal distribution $\log \mathcal{N}(\mu, \sigma^2)$ has density function:

$$p(y|\mu, \sigma^2) = \frac{1}{y \cdot \sqrt{2\pi\sigma^2}} \exp \left[-\frac{1}{2\sigma^2} (\log y - \mu)^2 \right]$$

where $y > 0$, $-\infty < \mu < \infty$ and $\sigma^2 > 0$.

The log-normal distribution is related to the normal distribution as follows: if $y \sim \log \mathcal{N}(\mu, \sigma^2)$ then $\log y \sim \mathcal{N}(\mu, \sigma^2)$.

Let $y_1, \dots, y_n | \mu, \sigma^2 \stackrel{iid}{\sim} \log \mathcal{N}(\mu, \sigma^2)$, where:

$\mu = 3.65$ (known)

σ^2 unknown with non-informative prior $p(\sigma^2) \propto 1/\sigma^2$

The posterior for σ^2 is the scaled inverse chi-squared distribution: Scale-inv- $\chi^2(n, \tau^2)$

where:

$$\tau^2 = \frac{\sum_{i=1}^n (\log y_i - \mu)^2}{n}$$

Part A

Generated 10000 random values using posterior distribution of σ^2 .

```
set.seed(123)
incomes <- c(22, 33, 31, 49, 65, 78, 17, 24)
mu <- 3.65
size <- length(incomes)

income_log_values <- log(incomes)
chi_square <- sum((income_log_values - mu)^2) / size

posterior_samples <- (size * chi_square)/rchisq(10000,size)
```

Plotting the posterior distribution of the values.

```
posterior_samples <- (size * chi_square)/rchisq(10000,size)
```

Part B

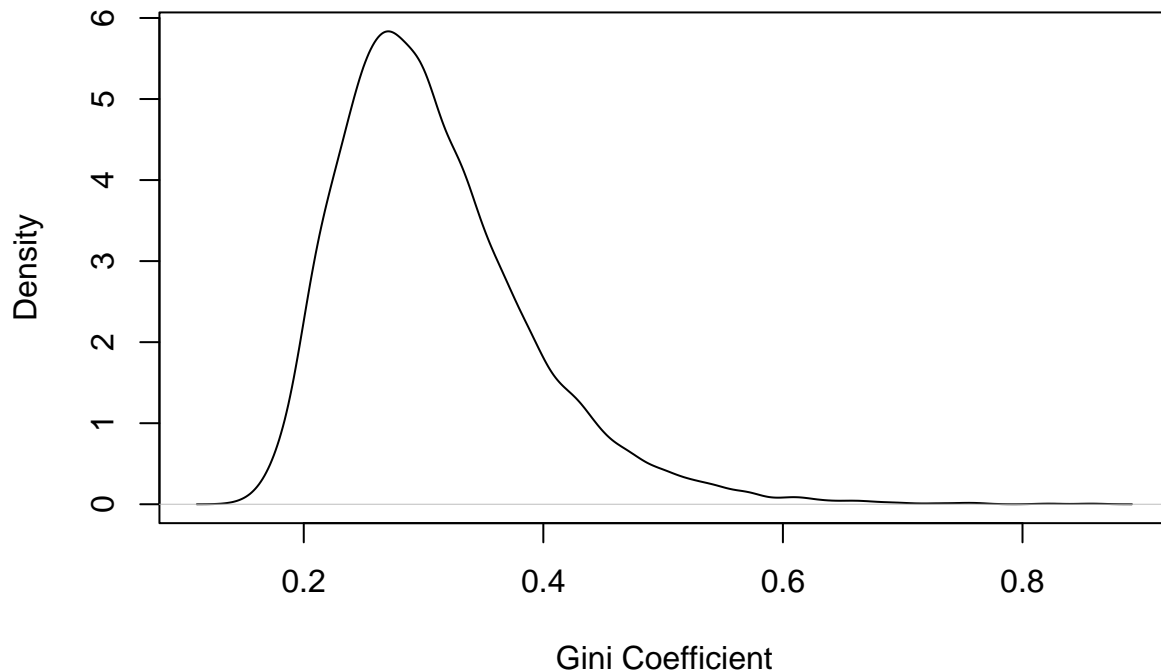
In this part, we transformed the samples generated in part A to estimate the Gini Coefficients to estimate the income inequality.

```
sigma_samples <- sqrt(posterior_samples)

gini_samples <- 2 * pnorm(sigma_samples / sqrt(2)) - 1
```

Now plotting the samples transformed.

Posterior Distribution of Gini Coefficient



Part C

Now we will be using samples generated in part b to construct a 95% equal tail credible interval.

```
ci <- quantile(gini_samples, probs = c(0.025, 0.975))
```

```
## 95% Equal Tail Credible Interval: 0.1952102 0.5075332
```

Part D

Now we will be using samples generated in part b to construct a 95% higher posterior density(hdi).

```
library(bayestestR)
hdi_ci <- hdi(gini_samples, ci = 0.95)
cat("95% HPDI:", hdi_ci$CI_low, hdi_ci$CI_high, "\n")
```

```
## 95% HPDI: 0.1759007 0.4685842
```

Assignment 3 : Bayesian inference for the rate parameter in the Poisson distribution.

Given data points that represents the number of goals scored in each match of the opening week of the 2024 Swedish women's football league : 0, 2, 5, 5, 7, 1, 4. These values are assumed to be as independent

observations from the Poisson Distribution with $\lambda > 0$. The prior distribution of λ be the half-normal distribution with prior pdf is given as following:

$$p(\lambda|\sigma) = \frac{\sqrt{2}}{\sigma\sqrt{\pi}} \exp\left(-\frac{\lambda^2}{2\sigma^2}\right), \quad \lambda \geq 0,$$

where the scale parameter σ is set to 5.

Part A

Here we'll derive unnormalized posterior distribution and then plot the normalized posterior density over a randomly generated lambda values of length 1000.

```
y <- c(0, 2, 5, 5, 7, 1, 4)
sum <- sum(y)
lambda_values <- seq(0, 10, length.out = 1000)

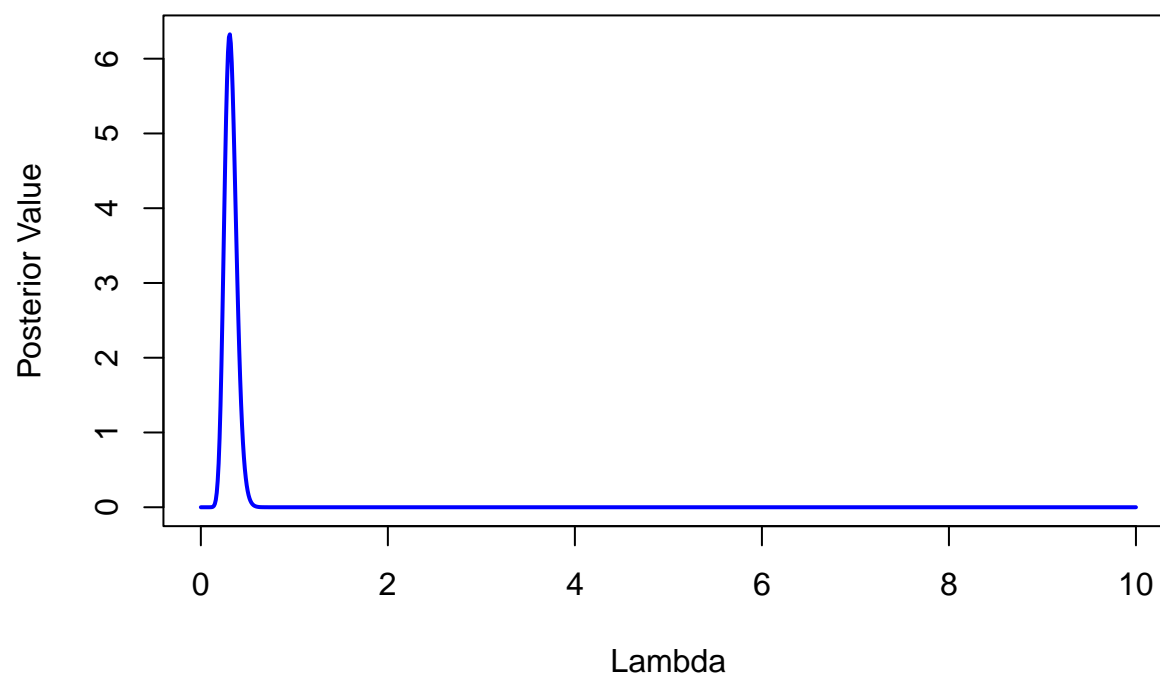
prior_values <- log(sqrt(2) / (5 * sqrt(pi))) - (lambda_values^2) / (2 * 5^2)
likelihood_values <- -n * lambda_values + sum * log(lambda_values)
posterior_values <- prior_values + likelihood_values

posterior_unnormalized <- exp((posterior_values - max(posterior_values)))

posterior_normalized <- posterior_unnormalized /
  sum(posterior_unnormalized * diff(lambda_values)[1])
```

Now Plotting the normalized posterior density

Posterior of lambda



Part B

Here we will identify the posterior mode for the above generated distribution.

```
posterior_mode <- lambda_values[which.max(posterior_normalized)]  
cat("The posterior mode is:", posterior_mode)
```

```
## The posterior mode is: 0.3103103
```