Computer Lab1 - 732A73

Helena Llorens Lluís (hllor282), Yi Yang (yiyan338)

1. Daniel Bernoulli

Let $y_1, ..., y_n|\theta \sim Bern(\theta)$, and assume a prior distribution for the parameter θ as $\theta \sim Beta(\alpha 0, \beta 0)$, with $\alpha_0 = \beta_0 = 7$. We have observed a sample of $n = 78$ trials with $f = 35$ failures. Using Bayes' theorem, the posterior distribution of θ is

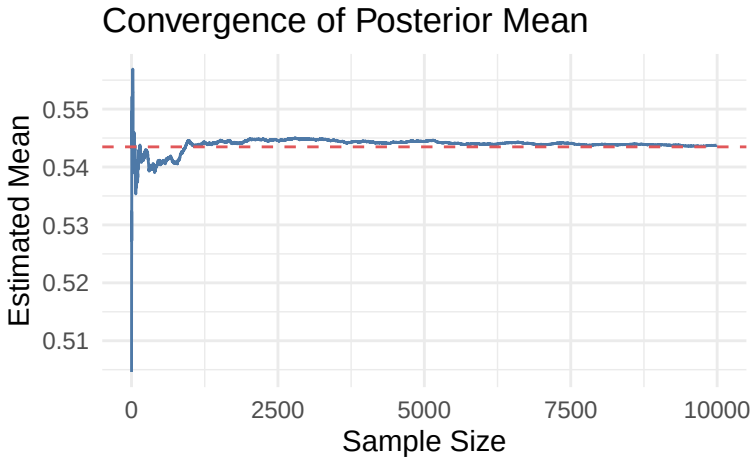
$$\theta|y \sim Beta(\alpha = \alpha_0 + s, \beta = \beta_0 + f)$$

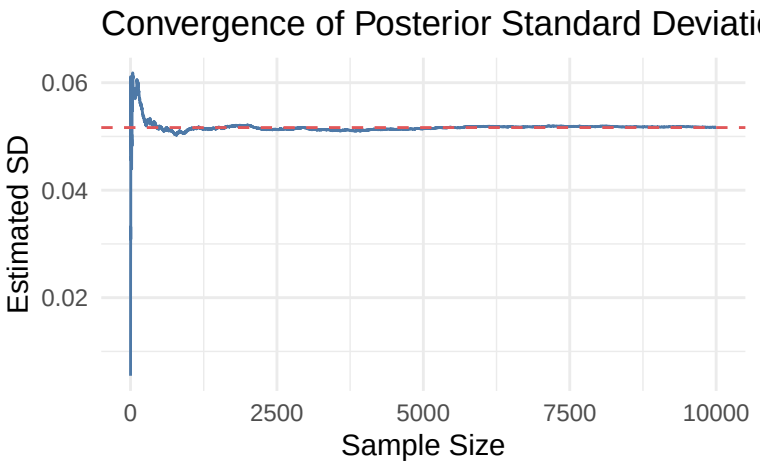
The theoretical (true) posterior mean and standard deviation are given by:

$$E[\theta|y] = \frac{\alpha}{\alpha + \beta}$$

$$SD[\theta|y] = \sqrt{\frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}}$$

To examine how estimates of the posterior mean and standard deviation behave as the number of random draws increases, we simulate 10000 samples from the posterior distribution and compute running estimates for both quantities.

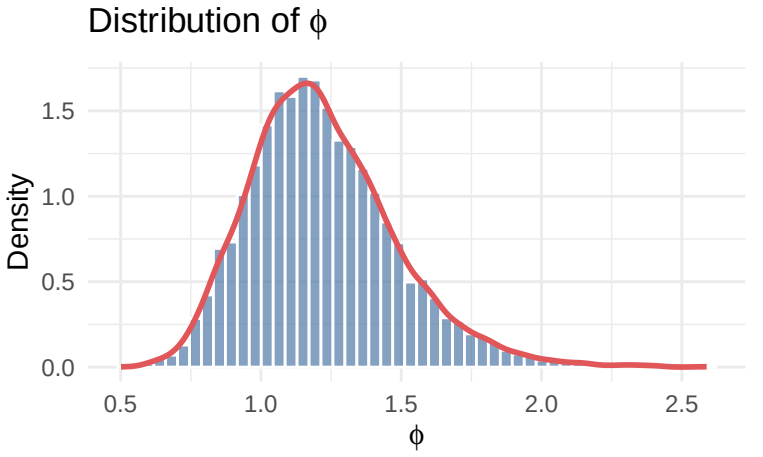




The plots above clearly illustrate that the estimates of the posterior mean and standard deviation converge toward their theoretical values as the number of random draws increases.

Then, we compute the posterior probability $P(\theta > 0.5|y)$. The posterior probability of $\theta > 0.5$ is 0.8055, and the exact value of $\theta > 0.5$ from the Beta posterior is 0.7990936.

Finally, we draw 10000 random values from the posterior of the odds $\phi = \frac{\theta}{1-\theta}$ and plot the posterior distribution of ϕ.



2.Log-normal distribution and the Gini coefficient

We consider a sample of 8 observations representing monthly incomes (in thousands of Swedish Krona). Assume the data follow a log-normal distribution:

$$y_1, ..., y_n|\mu, \sigma^2 \sim \log N(\mu, \sigma^2)$$

with known $\mu = 3.65$ and unknown variance σ^2. For the variance, we use a non-informative prior:

$$p(\sigma^2) \propto 1/\sigma^2$$

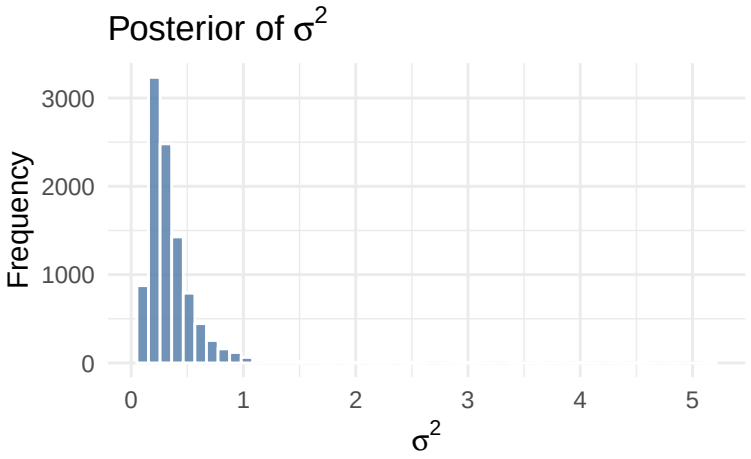
Under this model, the posterior distribution for σ^2 follows a scaled inverse-chi-squared distribution:

$$\text{Scale-inv-}\chi^2(n, \tau^2), \quad \text{where} \quad \tau = \frac{\sum_{i=1}^{n} (\log y_i - \mu)^2}{n}$$

To draw samples from this posterior, we use the fact that if:

$$\sigma \sim \text{Scale-inv-}\chi^2(n, \tau^2), \quad \text{then} \quad \sigma^2 = \frac{n\tau^2}{\chi_n^2}$$

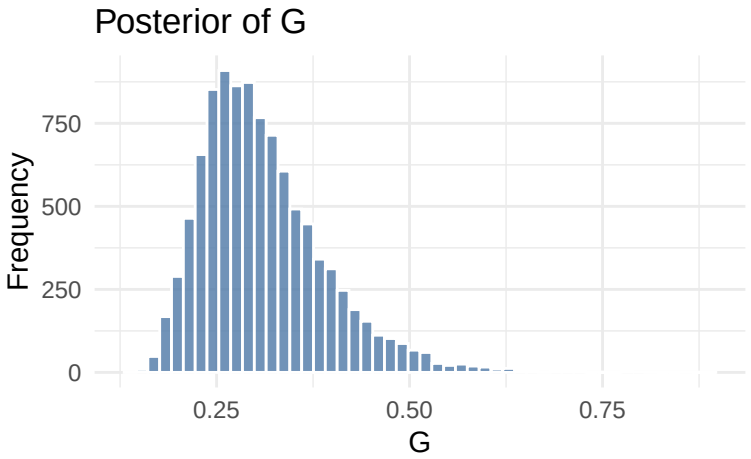
We generate 10000 posterior draws for σ^2 and visualize the resulting distribution below.



Then, the Gini coefficient is calculated using the formula:

$$G = 2\Phi(\sigma/\sqrt{2}) - 1$$

where Φ denotes the cumulative distribution function of the standard normal distribution, and $\sigma = \sqrt{\sigma^2}$. Using the posterior draws of σ^2, we obtain the posterior distribution of the Gini coefficient:



Using the posterior distribution of the Gini coefficient G, we compute both a 95% equal-tail credible interval and a 95% Highest Posterior Density Interval (HPDI) for G.

Table 1: Confidence intervals for G

	Lower.bound	Upper.bound
Equal tail	0.1930103	0.5080196
HPDI	0.1757738	0.4726963

The equal-tail interval removes 2.5% of the posterior mass from each tail, while the HPDI represents the narrowest interval containing 95% of the posterior mass. In this case, the HPDI is slightly narrower and starts at a lower bound compared to the equal-tail interval. This suggests that the posterior distribution of G is slightly right-skewed, and the HPDI better captures the region of highest probability. Therefore, if interpretability and precision are important, the HPDI may be preferred.

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

We are going to derive the expression that the posterior pdf $p(\lambda \mid y, \sigma)$ is proportional to. The observations are from the Poisson distribution with rate parameter $\lambda > 0$.The prior distribution of $\lambda > 0$ is the **half-normal distribution** with prior pdf:

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

From bayesian theorem, we know that the posterior is propotional the product of likelihood and the prior:

$$p(\theta \mid \text{Data}) \propto p(\text{Data} \mid \theta)\, p(\theta)$$

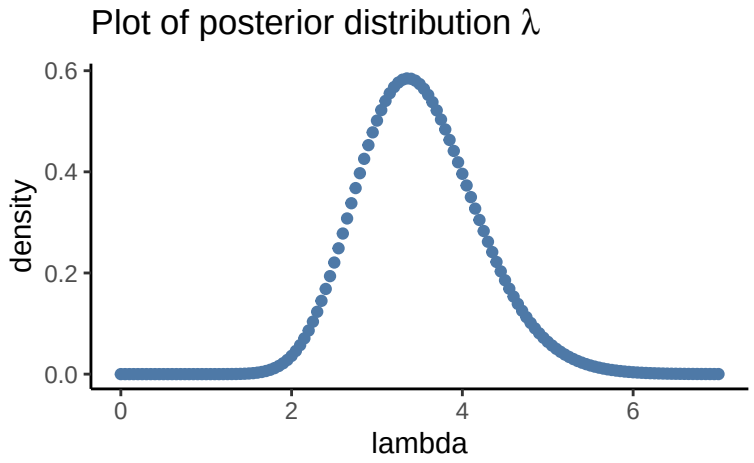
Then we could derive the expression of the posterior pdf :

$$p(\lambda \mid y, \sigma) \propto \lambda^{\sum y_i} \exp\left(-n\lambda - \frac{\lambda^2}{2\sigma^2}\right), \quad \lambda \geq 0$$

We select a grid of $\lambda > 0$ values from 0 to 7 and set the increment of the value equals to 0.5. We compute the posterior values and then use the following formulate to normalize the values:

$$\text{Normalized posterior} = \frac{p(\lambda)}{\sum p(\lambda) \cdot \Delta\lambda}$$

The plot of the posterior distribution is shown as follows:



The approximate posterior mode of lambda is 3.35

Appendix

Assignment 1

```
# data and parameters
n <- 78
f <- 35
s <- n - f
alpha0 <- 7
beta0 <- 7
alpha <- alpha0 + s
beta <- beta0 + f

# calculate theoretical mean and sd for the beta posterior
true_mean <- alpha / (alpha + beta)
true_sd <- sqrt(alpha * beta / ((alpha + beta)^2 * (alpha + beta + 1)))

# (a)
# sample of size 10000 from the beta posterior distribution
set.seed(12345)
draw <- rbeta(10000, shape1 = alpha, shape2 = beta)

# calculate mean and sd for each draw
mean_vals <- numeric(9999)
sd_vals <- numeric(9999)
for(i in 1:10000){
  mean_vals[i-1] <- mean(draw[1:i])
  sd_vals[i-1] <- sd(draw[1:i])
}

# mean values plot with theoretical value
ggplot(mapping = aes(x = seq_along(mean_vals), y = mean_vals)) +
  geom_line(color = "#4E79A7") +
```

```
  geom_hline(yintercept = true_mean, color = "#E15759", linetype = "dashed") +
  labs(title = "Convergence of Posterior Mean",
       x = "Sample Size",
       y = "Estimated Mean") +
  theme_minimal()

# sd values plot with theoretical value
ggplot(mapping = aes(x = seq_along(sd_vals), y = sd_vals)) +
  geom_line(color = "#4E79A7") +
  geom_hline(yintercept = true_sd, color = "#E15759", linetype = "dashed") +
  labs(title = "Convergence of Posterior Standard Deviation",
       x = "Sample Size",
       y = "Estimated SD") +
  theme_minimal()

# (b)
# sample probability
prob <- sum((draw > 0.5))/10000

# theoretical probability with posterior distribution
true_prob <- pbeta(0.5, alpha, beta, lower.tail = F)

# (c)
# calculate odds
odds <- draw/(1 - draw)

# plot posterior distribution of odds
ggplot(mapping = aes(x = odds)) +
  geom_histogram(aes(y = ..density..), bins = 50, fill = "#4E79A7", color = "white", alpha = 0.7) +
  geom_density(color = "#E15759", size = 1) +
  labs(title = expression(paste("Distribution of ", phi)),
       x = expression(phi),
       y = "Density") +
  theme_minimal()
```

Assignment 2

```
# data and parameters
y <- c(22, 33, 31, 29, 65, 78, 17, 24)
n <- 8
mu <- 3.65
tau <- sum((log(y) - mu)^2)/n

# (a)
# posterior distribution of sigma^2 (scale-inv-chi)
set.seed(12345)
samples <- (n * tau) / rchisq(10000, df = n)

# plot posterior distribution of sigma^2
ggplot(mapping = aes(x = samples)) +
  geom_histogram(bins = 50, fill = "#4E79A7", color = "white", alpha = 0.8) +
```

```
  labs(
    title = expression(paste("Posterior of ", sigma^2)),
    x = expression(sigma^2),
    y = "Frequency"
  ) +
  theme_minimal()

# (b)
# calculate gini index with the provided formula
gini <- 2 * pnorm(sqrt(samples)/sqrt(2), 0, 1) - 1

# plot posterior distribution of gini index
ggplot(mapping = aes(x = gini)) +
  geom_histogram(bins = 50, fill = "#4E79A7", color = "white", alpha = 0.8) +
  labs(
    title = "Posterior of G",
    x = expression(G),
    y = "Frequency"
  ) +
  theme_minimal()

# (c)
# compute equal-tail credible interval
ci <- quantile(gini, c(0.025, 0.975))

# (d)
# compute HPDI
hpdi <- hdi(gini, ci = 0.95)

table <- data.frame("Lower bound" = c(ci[1], hpdi[1,2]), "Upper bound" = c(ci[2], hpdi[1,3]))
rownames(table) <- c("Equal tail", "HPDI")
kable(table, caption = "Confidence intervals for G")
```

Assignment 3

```
# (a)
# data and parameters
y <- c(0, 2, 5, 5, 7, 1, 4)
sigma <- 5

# the expression of posterior distribution unnormalized
posterior <- function(lambda, sigma, y){
  s <- sum(y)
  n <- length(y)
  res <- lambda^(s)*exp(-n*lambda - (lambda^2)/(2*sigma^2))
  return(res)
}

# unnormalized posterior distribution
lambda_grid <- seq(0, 7, by=0.05)
unnormalized <- posterior(lambda_grid, sigma, y)
```

```r
# normalized posterior
normalized <- unnormalized / sum(unnormalized * 0.05)

# plot normalized posterior
df <- data.frame(lambda = lambda_grid, density = normalized)
ggplot(data = df, mapping = aes(x = lambda, y = density))+
  geom_point(color = "#4E79A7")+
  labs(title = "Plot of posterior distribution ")+
  theme_classic()

#(b)
mode_index <- which.max(normalized)
mode <- lambda_grid[mode_index]
```