

# Lab1

*Naveen Gabriel ,Sridhar Adhikarla*

*2019-04-14*

## Contents

<b>1</b>	<b>1 Bernoulli ... again.</b>	<b>2</b>
1.1	a.) Draw random numbers from the Beta posterior . . . . .	2
1.2	b.) Compute the posterior probability $Pr(\theta < 0.4 y)$ . . . . .	3
1.3	c.)Posterior distribution of the log-odds . . . . .	3
<b>2</b>	<b>2. Log-normal distribution and the Gini coefficient.</b>	<b>4</b>
2.1	a.) Simulate 10000 draws from the posterior of $\sigma^2$ . . . . .	4
2.2	b.) Gini coeffecient . . . . .	5
2.3	c.)Equal tailed interval vs HPD of posterior. . . . .	6
<b>3</b>	<b>3. Von Mises distribution.</b>	<b>7</b>
<b>4</b>	<b>Appendix</b>	<b>10</b>

## 1 1 Bernoulli ... again.

We have Beta posterior distribution as :

$$\theta|y \sim \text{Beta}(\alpha_0 + s, \beta_0 + f)$$

$$\theta|y \sim \text{Beta}(16, 8)$$

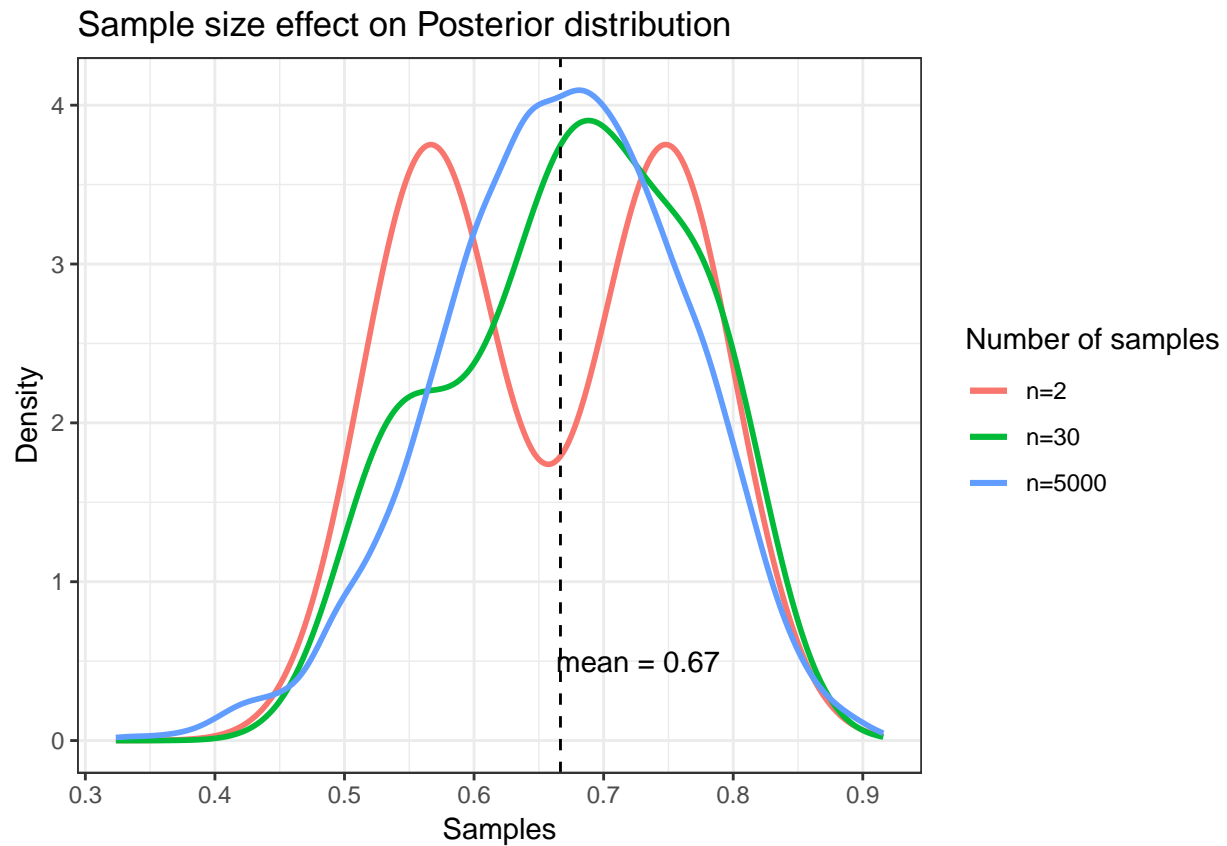
### 1.1 a.) Draw random numbers from the Beta posterior

Knowing the parameters of posterior distribution, we computed the true mean and variance distribution using the below formula:

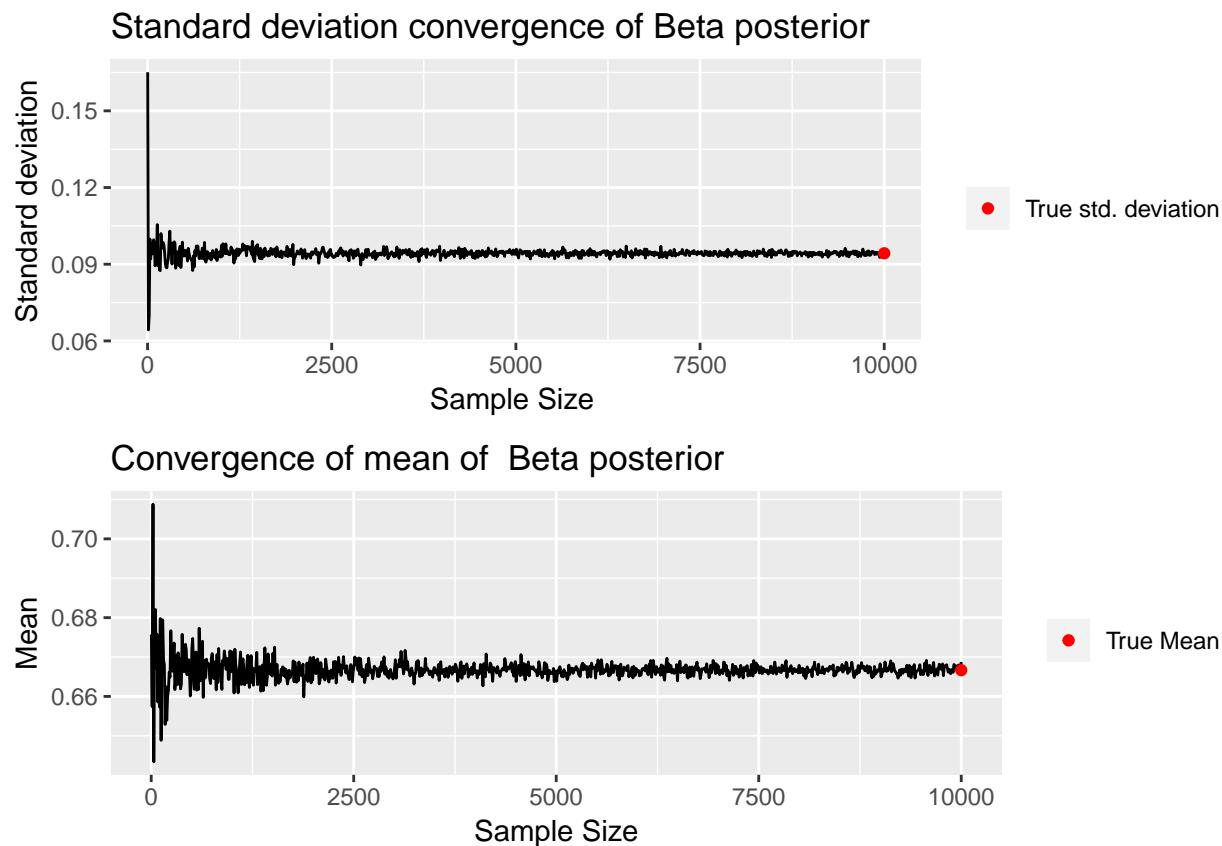
$$\mu = \frac{\alpha}{\alpha + \beta}$$

$$\sigma^2 = \frac{\alpha * \beta}{(\alpha + \beta)^2 * (\alpha + \beta + 1)}$$

Plotting the density graph shows how most of the numbers in sample centered around true mean as the number of samples is increased.



Below graphs shows the convergence of mean and standard deviation to actual mean and standard deviation as the number of samples increases.



## 1.2 b.) Compute the posterior probability $Pr(\theta < 0.4|y)$

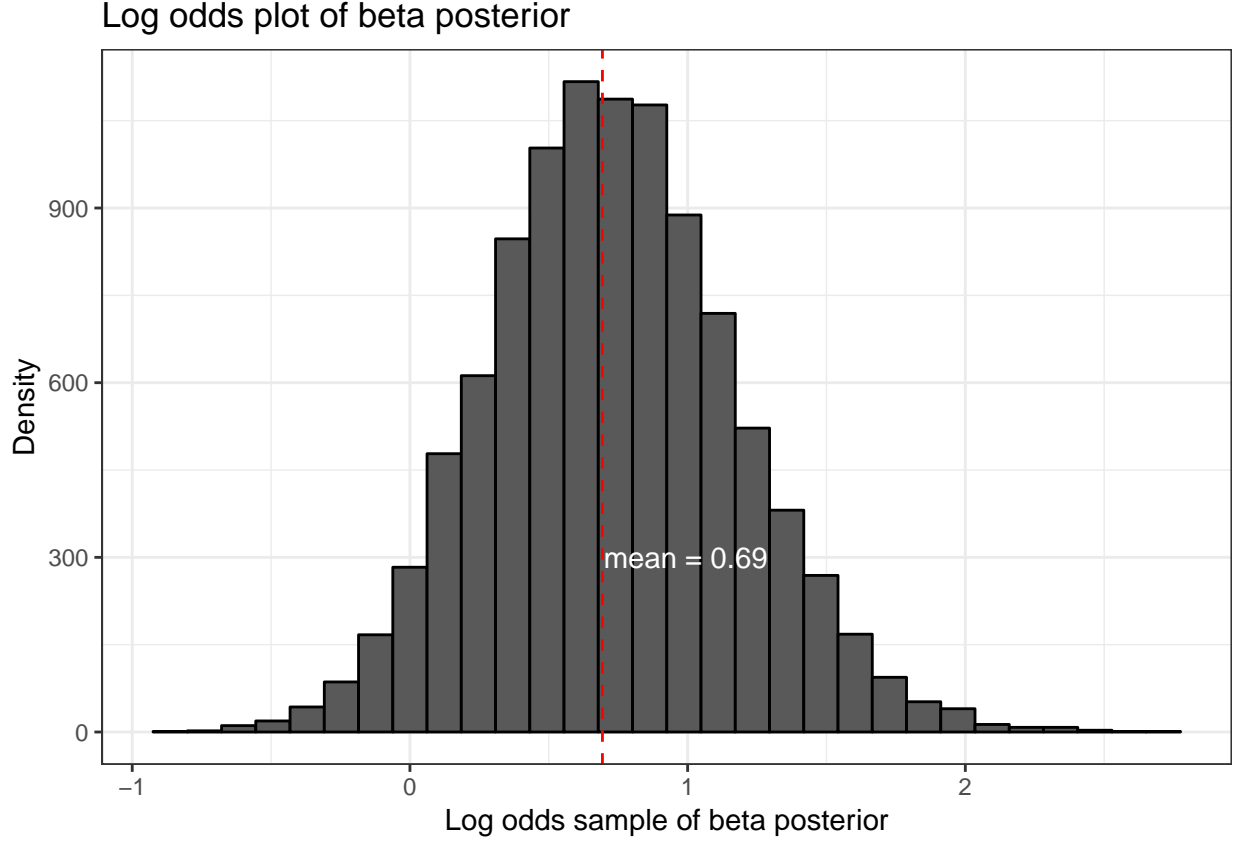
We sampled 10000 points from Beta posterior and checked how many of those points has value lesser than 0.4. Below are computed and true results of  $Pr(\theta < 0.4|y)$ . Both of these values are nearly same.

Computed value of CDF : 0.004

True value of CDF : 0.00397

## 1.3 c.)Posterior distribution of the log-odds

Log odds are an alternate way of expressing probabilities, which simplifies the process of updating them with new evidence. Since our posterior samples are in probabilities, we convert them into log odds and check if the new mean of samples is nearly same as the transformed mean of posterior which is shown as red line. We conclude that, even after transforming posterior parameters to log odds, it mean of the sample agrees with the sam-



ples generated.

## 2 2. Log-normal distribution and the Gini coefficient.

### 2.1 a.) Simulate 10000 draws from the posterior of $\sigma^2$

The task was to simulate 10,000 draws of  $\sigma^2$  from posterior distribution and compare it with the theoretical chi square distribution.

Model:

$$\log x_1, \log x_2, \dots, \log x_n \sim \mathcal{N}(\mu, \sigma^2)$$

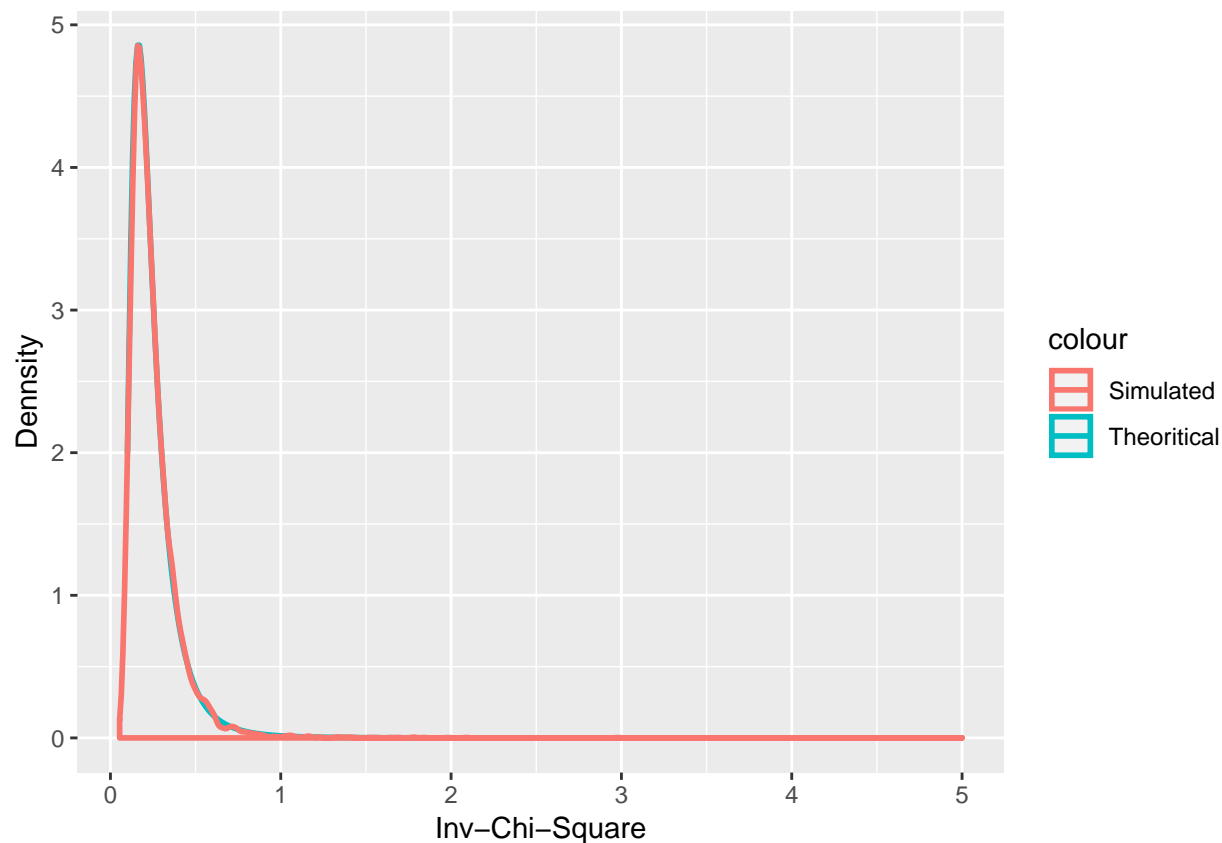
Non informative Prior:

$$p(\sigma^2) \propto (\sigma^2)^{-1}$$

Posterior :

$$\sigma^2 | x \sim \text{Inv} - \chi(n-1, s^2)$$

We used the pdf of scaled inverse chi square to draw samples for our theoretical value. After drawing samples from both we plotted them as below.



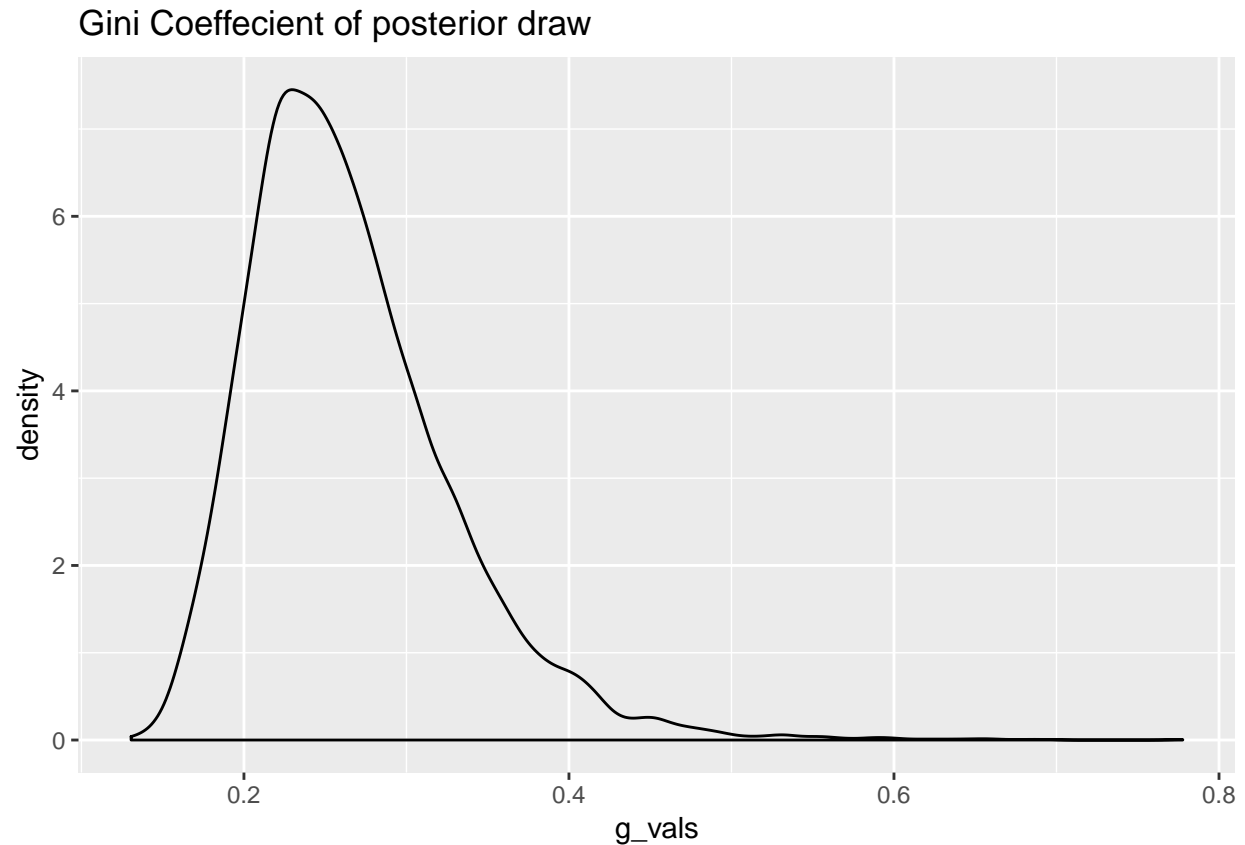
The red curve shows our 10,000 draw from posterior distribution. It gives an idea about probable values of chi square which we can accept. From our posterior it seems most of the chi square value is around 0.1 which we may accept for further computation.

Blue colour shows the theoretical chi square and it seems to be similar to the distribution of posterior. Though both is not accurate to dot but we can say our posterior has done good job in estimating parameter which resembles nearly to the theoretical inverse chi square distribution.

## 2.2 b.) Gini coefficient

We have salary (in thousand SEK ) from the following ten observations: 14, 25, 45, 25, 30, 33, 19, 50, 34 and 67. Gini coefficient of 0 expresses perfect equality, where all values are the same (where everyone has the same income). A Gini coefficient of 1 (or 100%) expresses maximal inequality among value .

Based on our previous draw of  $\sigma^2$  from posterior we computed the gini coefficient. Below graph shows the probable value of gini value. From the graph we can immediately say that most of the value is nearly 0.3 hence there is high chances that the mentioned salary is nearly same among the people.



### 2.3 c.)Equal tailed interval vs HPD of posterior.

Using the posterior draws from b)we computed a 95% equal tail credible interval and HPD. Below value shows the difference of values for both method.

```
[1] 1
```

```
95% Equal tailed credible interval for posterior:
```

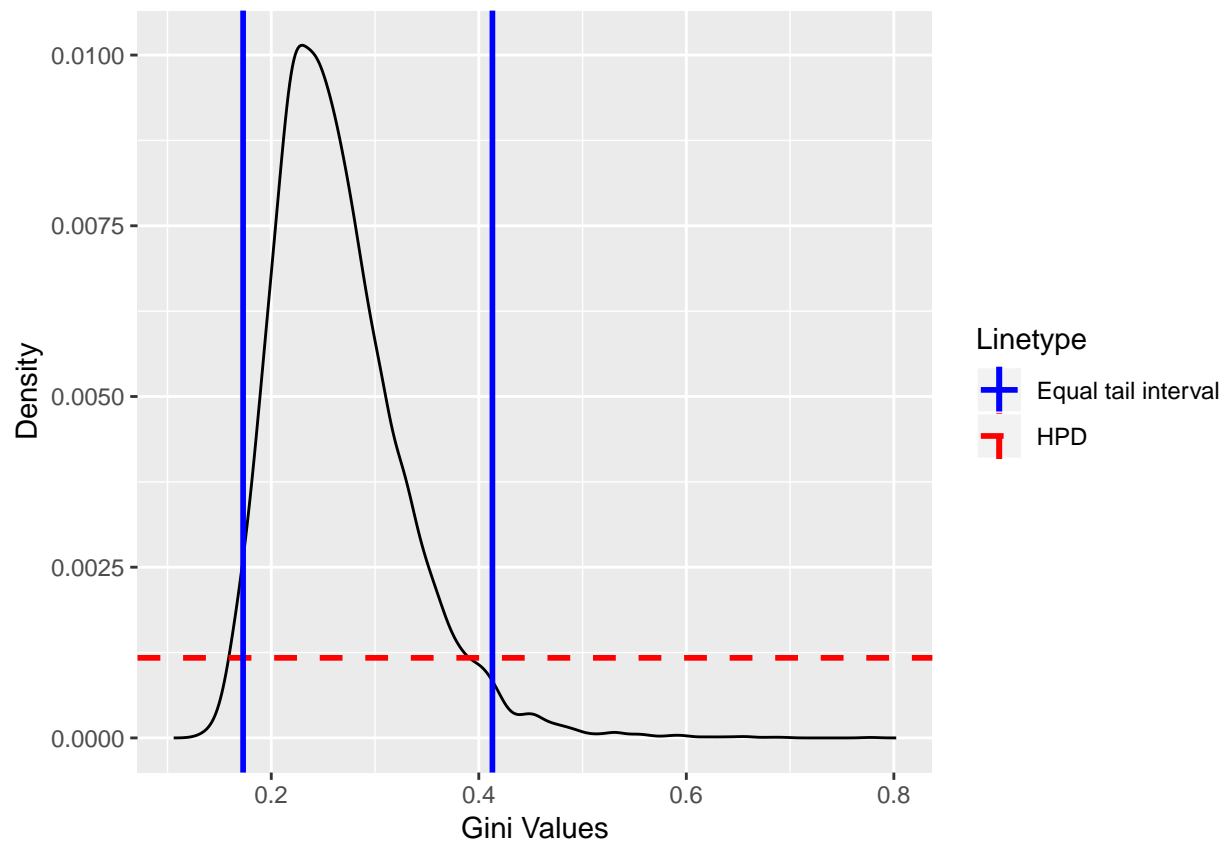
```
Lower Interval: 0.3907731
```

```
UpperInterval: 0.1591066
```

```
95% Equal tailed credible interval for posterior:
```

```
Lower Interval: 0.1728546
```

```
UpperInterval: 0.413131
```



Equal tail interval cuts the interval from side of the distribution while HPD considers the highest density from top to the bottom and cuts the distribution horizontally

### 3 3. Von Mises distribution.

a.) Posterior distribution plot -

$$\text{Likelihood} : P(y|\mu, k) \sim \frac{\exp(k * \cos(y - \mu))}{2 * \pi * I_0(k)}, \text{ where } (-\pi \leq y \leq \pi), k > 0,$$

$$\text{Known} : \mu = 2.39, \lambda = 1,$$

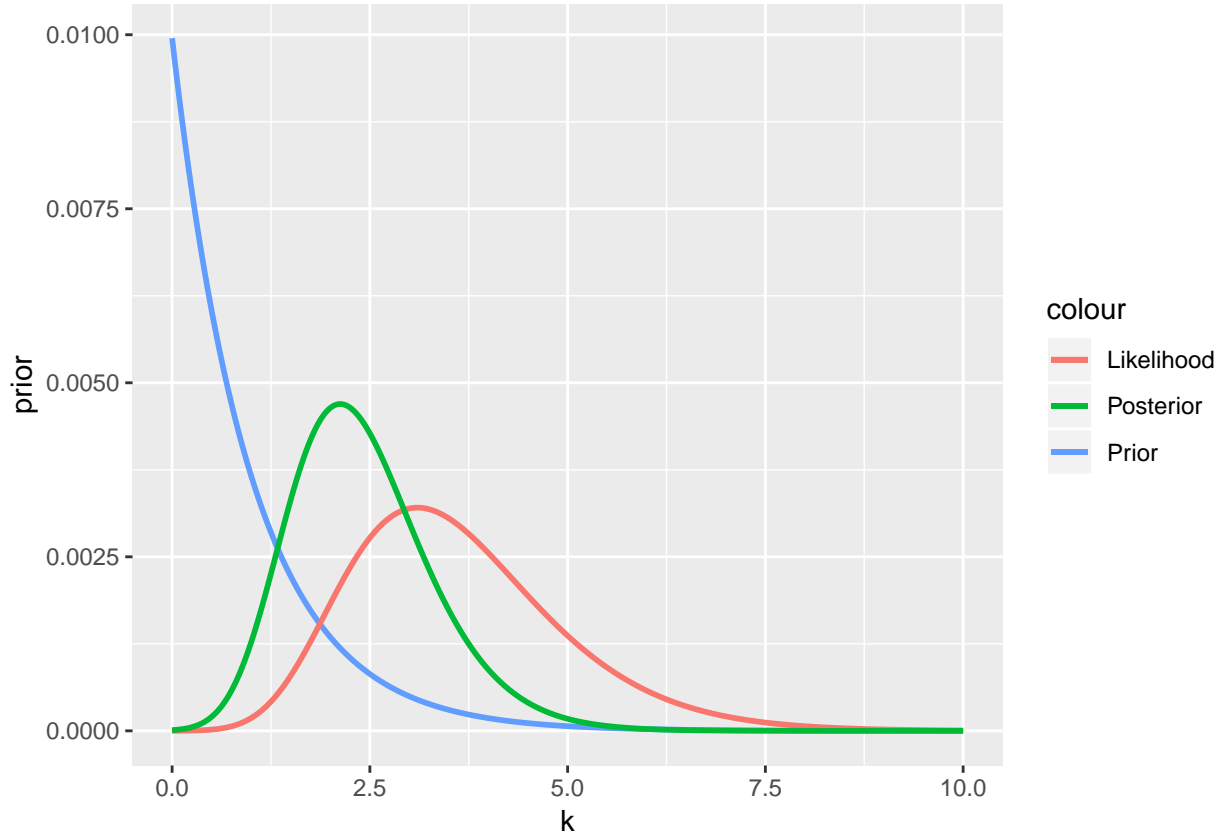
$$\text{Data} : y = (-2.44, 2.14, 2.54, 1.83, 2.02, 2.33, -2.79, 2.23, 2.07, 2.02),$$

$$\text{Prior} : P(k) = \lambda * \exp(-\lambda * k) = \exp(-k),$$

$$\text{Posterior} : \text{posterior} \propto \text{likelihood} * \text{prior}$$

$$P(k|y, \mu) \propto P(y|\mu, k) * P(k)$$

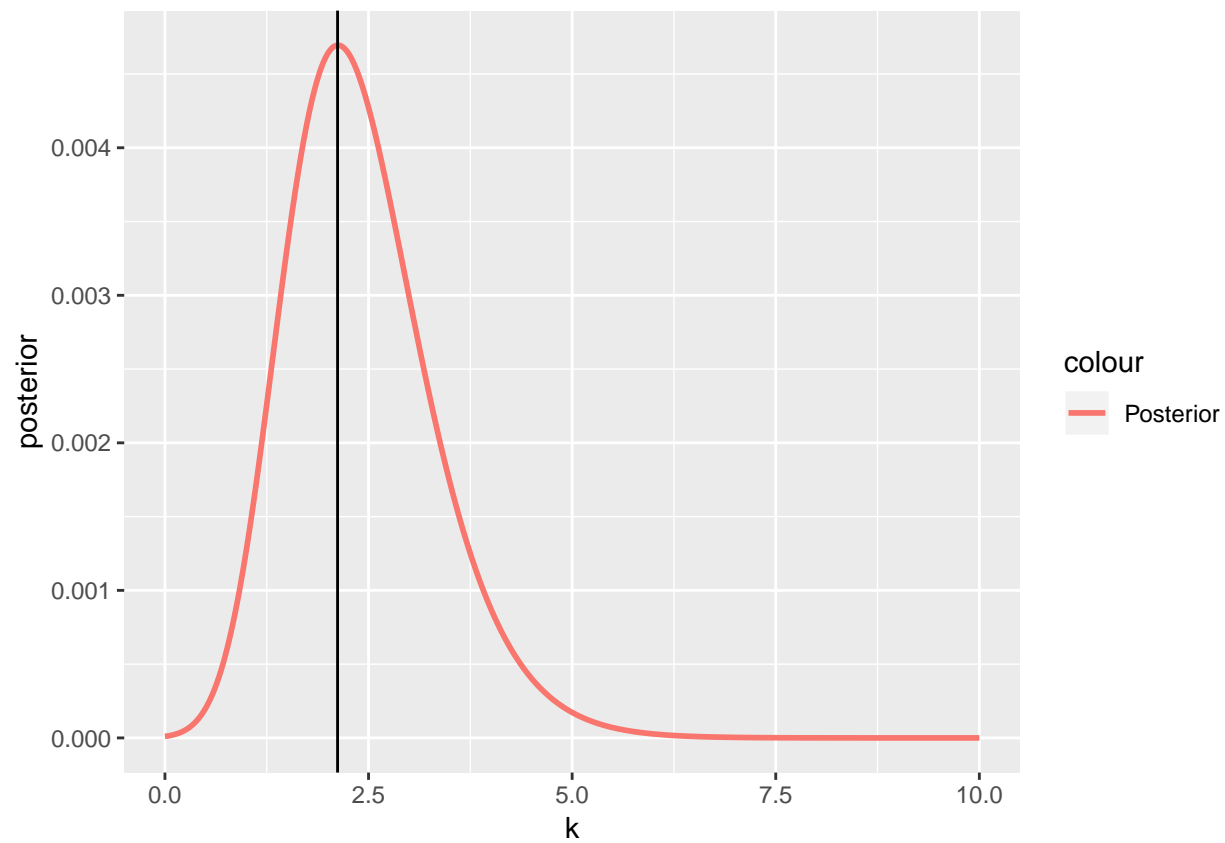
$$P(k|y, \mu) \propto \frac{\exp(k * \cos(y - \mu))}{2 * \pi * I_0(k)} * \exp(-k)$$



We normalized the likelihood, prior and the posterior so that we get a better understanding of the calculation by plotting them on the same plot. I chose the range 0-10 for possible k values. From the posterior, we can see that the most probable value for k is somewhere between (2 and 2.5).

b. ) Approximate Mode of the posterior distribution -





The approximate mode for the distribution is : 2.12

## 4 Appendix

```
knitr::opts_chunk$set(  
  echo = FALSE,  
  eval=TRUE,  
  message = FALSE,  
  warning = FALSE,  
  comment = NA  
)  
  
library(ggplot2)  
library(gridExtra)  
library(LaplacesDemon)  
alpha <- 16  
beta <- 8  
  
beta_mean <- alpha/(alpha+beta)  
beta_var <- alpha*beta/(((alpha+beta)^2)*(alpha+beta+1))  
  
set.seed(1234)  
pos_beta1 <- rbeta(2,alpha,beta,ncp=0)  
pos_beta2 <- rbeta(30,alpha,beta,ncp=0)  
pos_beta3 <- rbeta(5000,alpha,beta,ncp=0)  
  
ggplot() + stat_density(aes(x=pos_beta1, color="pos_beta1"),geom="line",size=1) +  
  stat_density(aes(x=pos_beta2, color="pos_beta2"),geom="line",size=1) +  
  stat_density(aes(x=pos_beta3, color="pos_beta3"),geom="line",size=1) +  
  geom_vline(xintercept = beta_mean,linetype="dashed") +  
  annotate("text", label = paste0("mean = ", round(beta_mean,2)), size = 4, x = beta_mean+0.06,  
  scale_color_discrete(name="Number of samples",labels=c("n=2","n=30","n=5000")) + xlab("Samples")+ y.  
  theme_bw()  
  
mn <- c()  
s_dev <- c()  
j<-1  
sq <- seq(2,10000,length=1000)  
  
for (i in sq ) {  
  posbeta <- rbeta(i,alpha,beta,ncp=0)  
  mn[j] <- mean(posbeta)  
  s_dev[j] <- sd(posbeta)  
  j<- j+1  
}  
  
cat ("\n")  
plot1 <- ggplot() + geom_line(aes(x = sq, y = s_dev)) + ggtitle("Standard deviation convergence of Beta  
  xlab("Sample Size ") + ylab("Standard deviation") +  
  scale_color_manual(name = NULL, values = c("red"),labels = "True std. deviation")  
  
cat ("\n")  
plot2 <- ggplot() + geom_line(aes(x = sq, y = mn)) + ggtitle("Convergence of mean of Beta posterior")
```

```

    geom_point(aes(x = 10000, y = beta_mean, color = "red")) +
    xlab("Sample Size") + ylab("Mean") +
    scale_color_manual(name = NULL, values = c("red"), labels = "True Mean")

grid.arrange(plot1, plot2, nrow=2)
set.seed(12)
pos_beta <- rbeta(10000, alpha, beta)
pos_beta_l <- sum(pos_beta < 0.4)

computed_val <- pos_prob <- pos_beta_l / 10000

true_val <- pbeta(0.4, alpha, beta)

cat("Computed value of CDF :", computed_val)
cat("\nTrue value of CDF      :", round(true_val, 5))

beta_mean_log <- log(beta_mean / (1 - (beta_mean)))
pos_beta_log <- log(pos_beta / (1 - pos_beta))

ggplot() + geom_histogram(aes(x = pos_beta_log), color = "black") +
  geom_vline(xintercept = beta_mean_log, linetype = "dashed", color = "red") +
  annotate("text", label = paste0("mean = ", round(beta_mean_log, 2)), size = 4, x = beta_mean_log, y = 0.05) +
  ggtitle("Log odds plot of beta posterior") + xlab("Log odds sample of beta posterior") + ylab("Density")
set.seed(12345)
mu = 3.5
y = c(14, 25, 45, 25, 30, 33, 19, 50, 34, 67)
n = 10
tow = sum((log(y) - mu)^2) / n

x = seq(0.1, 5, 0.01)
theo_inv_chi = dinvchisq(x, n, tow)
simu_inv_chi = rinvcchisq(10000, n, tow)
ggplot() +
  geom_line(aes(x, theo_inv_chi, col = "Theoretical"), size = 1) +
  geom_density(aes(simu_inv_chi, col = "Simulated"), size = 1) +
  ylab("Density") +
  xlab("Inv-Chi-Square")

g_vals = (2 * pnorm(sqrt(simu_inv_chi / 2))) - 1
ggplot() + geom_density(aes(g_vals)) + ggtitle("Gini Coefficient of posterior draw")
g_dens = density(g_vals)
data_for_hpd = data.frame(giniVals = g_dens$x, giniDens = g_dens$y)
data_for_hpd$giniDens = data_for_hpd$giniDens / sum(data_for_hpd$giniDens)

data_for_hpd = data_for_hpd[order(data_for_hpd$giniDens, decreasing = TRUE), ]
data_for_hpd$cumPost = cumsum(data_for_hpd$giniDens)
max(data_for_hpd$cumPost)
hpd_vals = which(data_for_hpd$cumPost < 0.95)
hpd_cut = length(hpd_vals)

cat("95% Equal tailed credible interval for posterior:\nLower Interval: ",
    data_for_hpd$giniVals[hpd_cut], "\nUpper Interval: ",
    data_for_hpd$giniVals[hpd_cut - 1], "\n\n")

```

```

quant = quantile(g_vals, c(0.025, 0.975))
cat("95% Equal tailed credible interval for posterior:\nLower Interval: ",
    quant[1], "\nUpperInterval: ",
    quant[2], "\n")

ggplot() + geom_line(aes(data_for_hpd$giniVals, data_for_hpd$giniDens)) +
  geom_hline(aes(yintercept = data_for_hpd$giniDens[hpd_cut], linetype = "HPD"), color = "red", size = 1) +
  geom_vline(aes(xintercept = quant[1], linetype = "Equal tail interval"), color = "blue", size = 1) +
  geom_vline(aes(xintercept = quant[2], linetype = "Equal tail interval"), color = "blue", size = 1) +
  xlab("Gini Values") +
  ylab("Density") +
  scale_linetype_manual(name="Linetype", values = c(1,2), guide = guide_legend(override.aes = list(color = "red")))
#A3
mu = 2.39
y = c(-2.44, 2.14, 2.54, 1.83, 2.02, 2.33, -2.79, 2.23, 2.07, 2.02)
prior_a3 = function(k){
  return(exp(-k))
}
likelihood_a3 = function(y, k, mu){
  numer = exp(k*(sum(cos(y - mu))))
  dinom = (2 * pi * besseli(k, 0))^10
  return(numer/dinom)
}

k = seq(0, 10, 0.01)
vals = data.frame(k, prior=prior_a3(k), likelihood=likelihood_a3(y, k, mu))
vals$posterior = vals$prior*vals$likelihood
prior_sum = sum(vals$prior)
likel_sum = sum(vals$likelihood)
postr_sum = sum(vals$posterior)
vals$prior = vals$prior/prior_sum
vals$likelihood = vals$likelihood/likel_sum
vals$posterior = vals$posterior/postr_sum

ggplot(vals) + geom_line(aes(k, prior, col="Prior"), size=1) +
  geom_line(aes(k, likelihood, col="Likelihood"), size=1) +
  geom_line(aes(k, posterior, col="Posterior"), size=1)
max_ind = order(vals$posterior, decreasing = TRUE)[1]
mode_aprox = vals$k[max_ind]
ggplot(vals) + geom_line(aes(k, posterior, col="Posterior"), size=1) +
  geom_vline(xintercept = mode_aprox)
cat("The approximate mode for the distribution is : ", mode_aprox)

```