

TDDE07-lab1

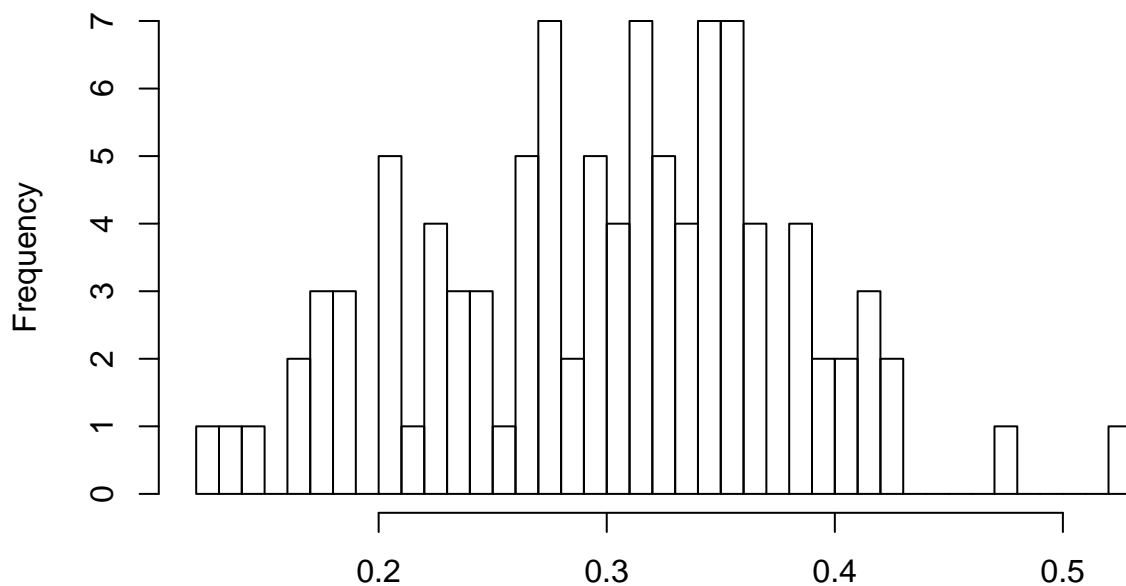
David Nyberg (davny376)

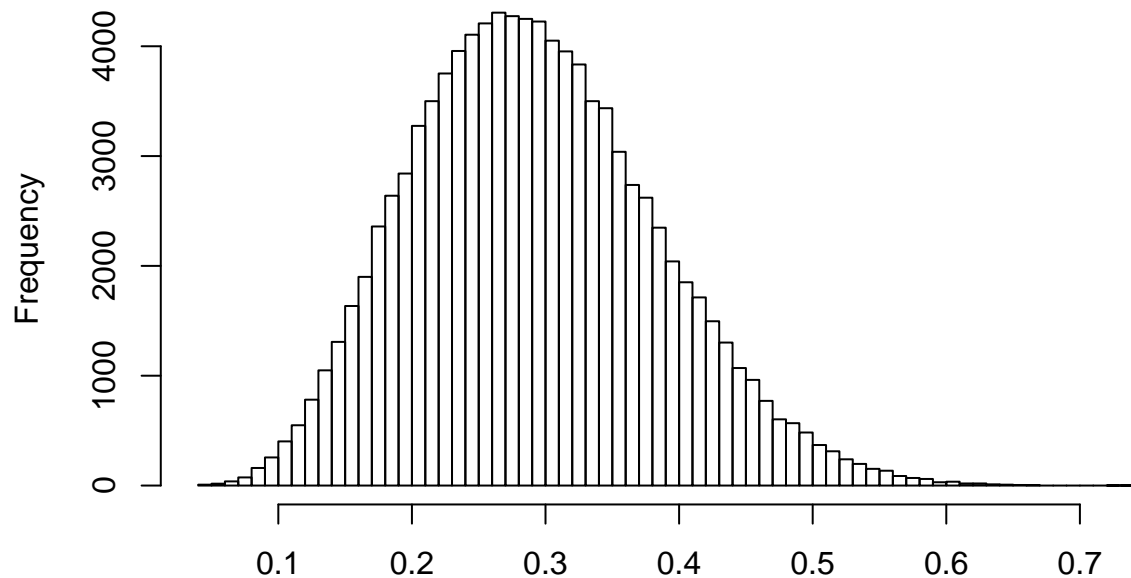
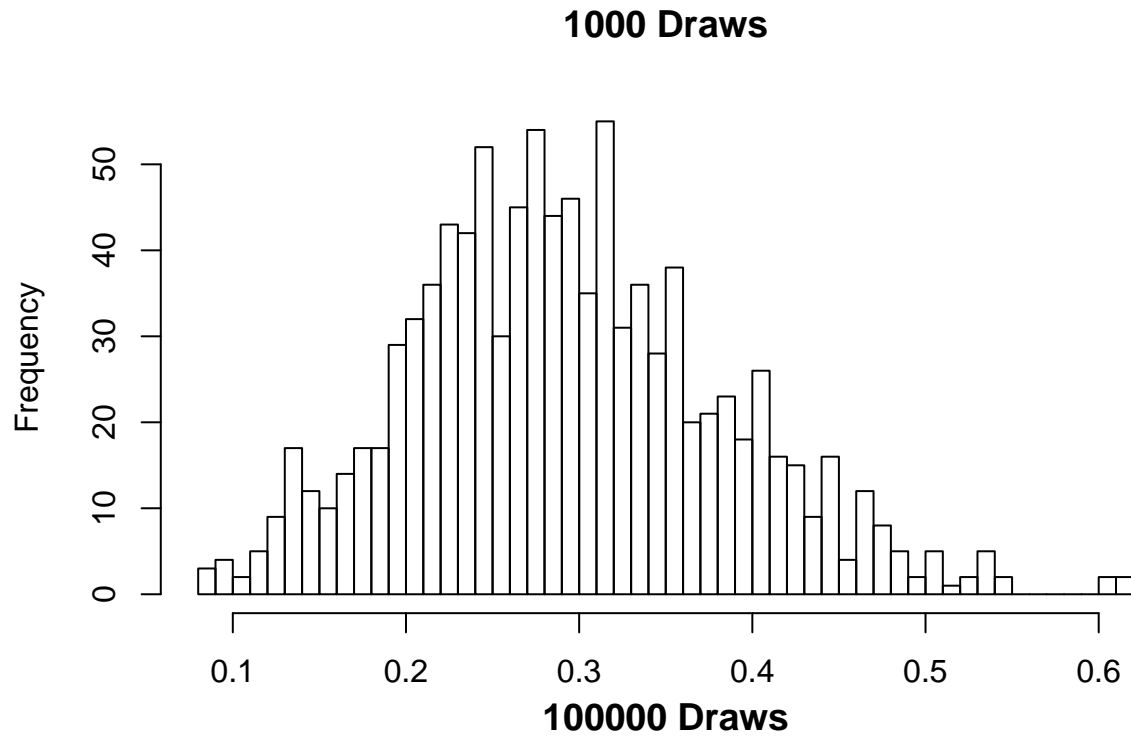
4/2/2020

1) Bernoulli... again

- a) Drawing random numbers from our posterior $\theta|y \sim \text{Beta}(\alpha + s, \beta + f)$ with 5 successes and 15 failures we can see graphically how the posterior converges to the true values when we increase the number over draws from only 100 up to 100,000 gets us a smooth curve.

100 Draws





Calculating the true standard deviation and the true mean we can see the results from 100000 draws looks very accurate in comparison to the true values.

```
## [1] "True standard deviation: 0.00826388888888889"
```

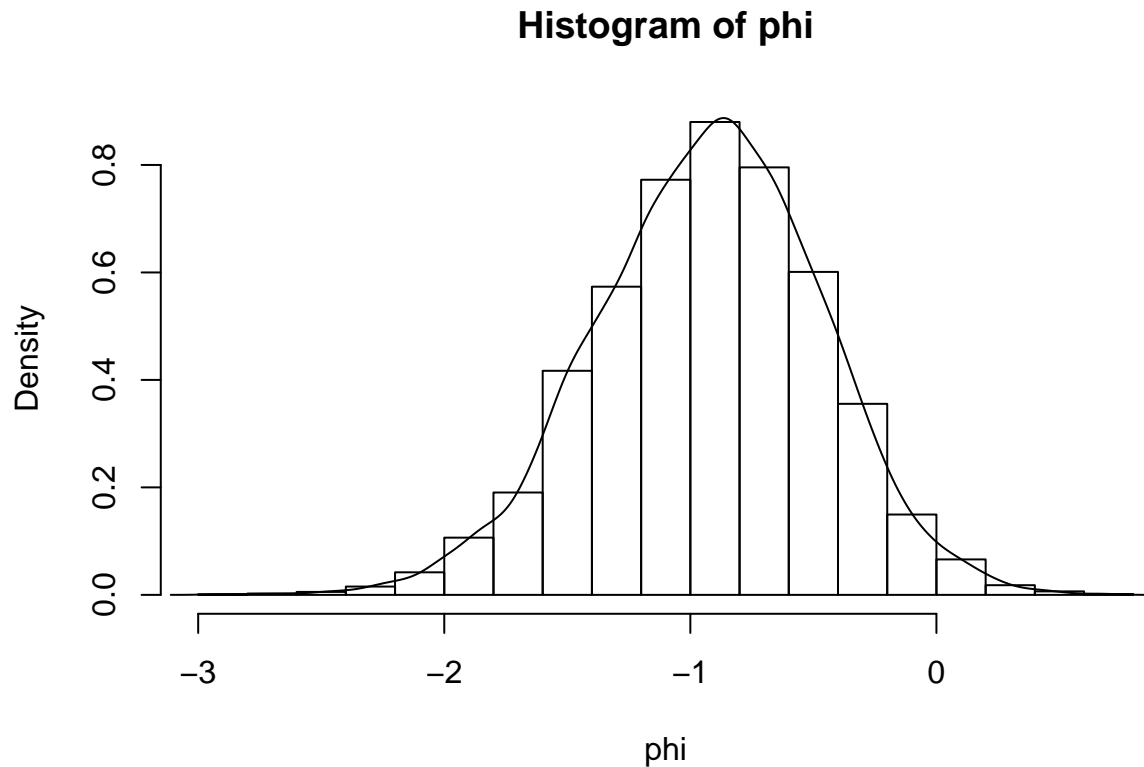
```
## [1] "True mean: 0.291666666666667"
```

- b) Use simulation ($n\text{Draws} = 10000$) to compute the posterior probability $\Pr(\theta > 0.3|y)$ compared with the simulated we can see the value is almost the same in both results. This shows that our posterior probability is very close to the true probability.

```
## [1] "The true probability theta > 0.3: 0.439947215300724"
```

```
## [1] "The sampled probability theta > 0.3: 0.4433"
```

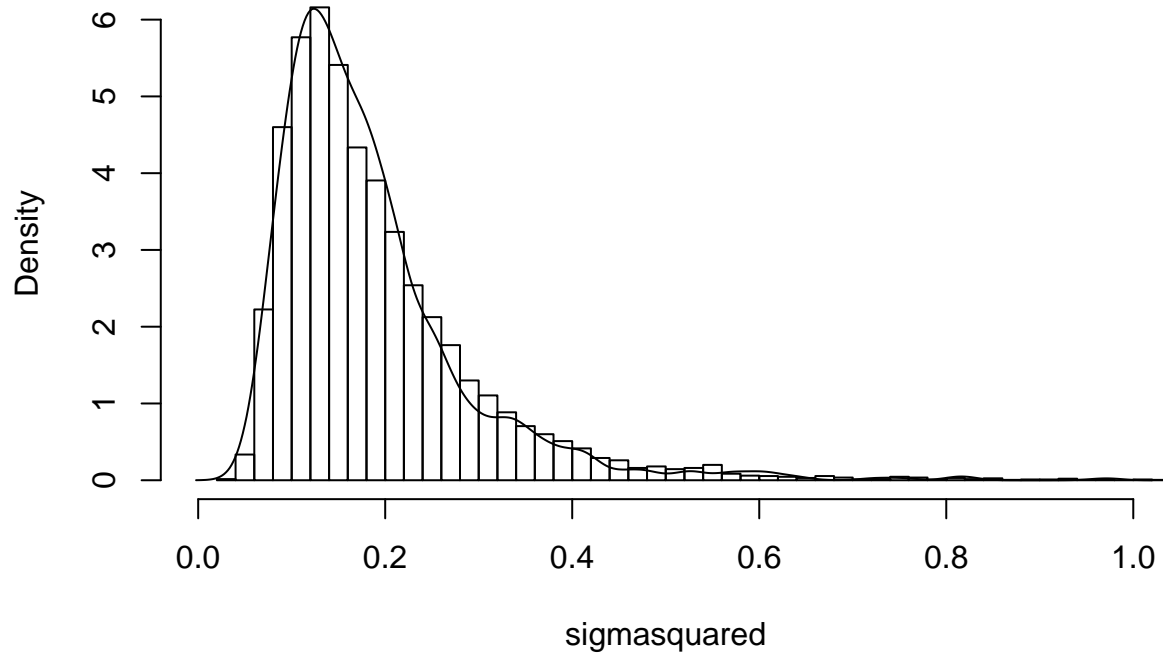
c) Here is a plot of the posterior distribution of the log odds by simulation showing density of the log odds.



2) Log-normal distribution and the Gini coefficient.

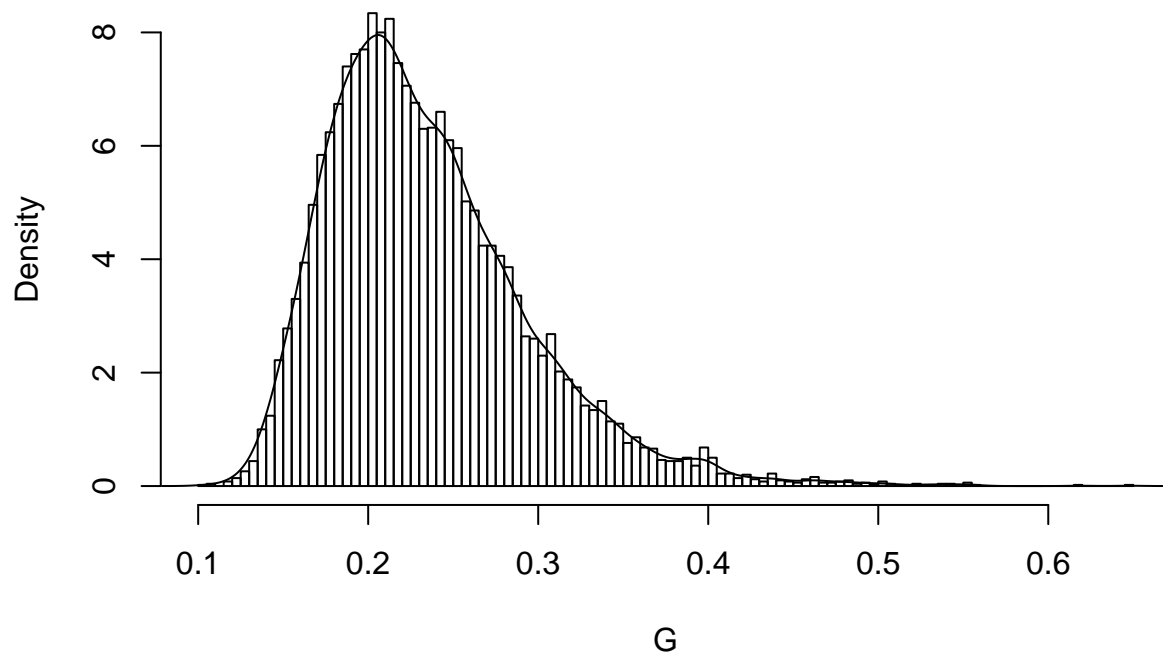
- a) Start by simulating 10,000 draws from the posterior of σ^2 with μ of 3.7 is assumed. Comparing this simulated result with the theoretical inv chisquared we can see that they are similar. In the plot the histogram is the simulated inv chi squared with the calculated tau squared. The line is the true theoretical distribution.

Histogram of sigmasquared



- b) Using the gini coefficient calculation with the previous results as input the following plot shows the density of the gini coefficient.

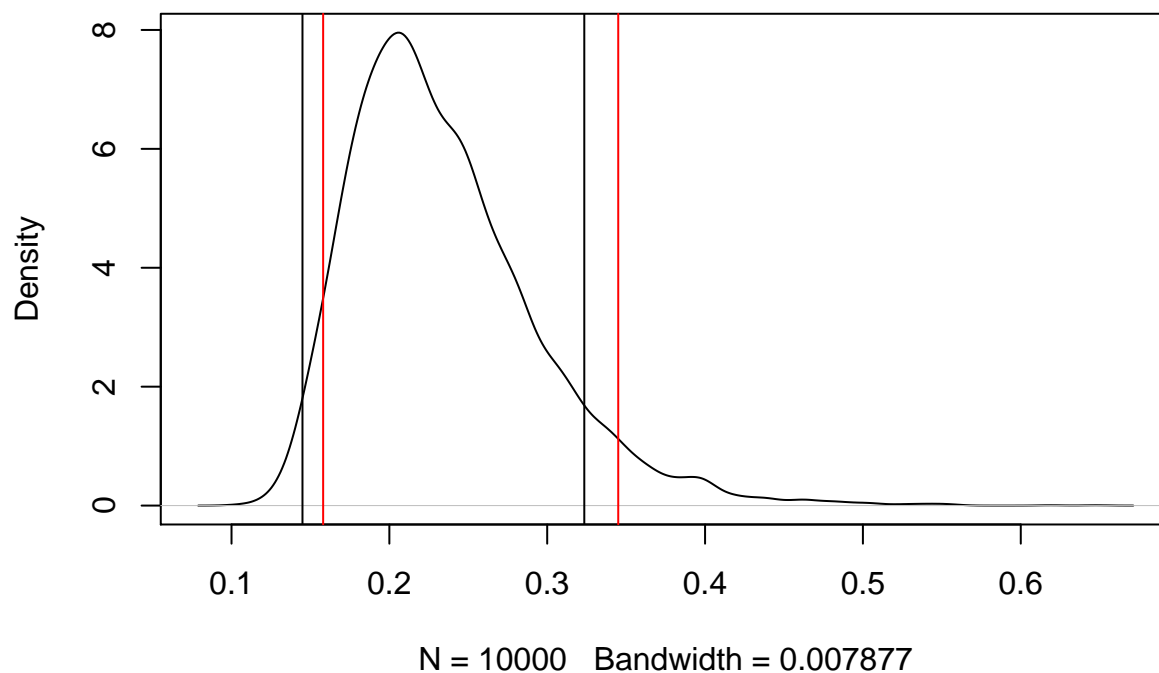
Histogram of G



- c) The 90% equal tail credible interval for G: (0.158, 0.346) at 5% and 95% respectively. Kernel density estimate of 90% highest posterior density: (0.144, 0.323) at 5% and 95% respectively. Plotted below is the comparison of the two interval. Red is equal tail which we can see is slightly shifted to the right

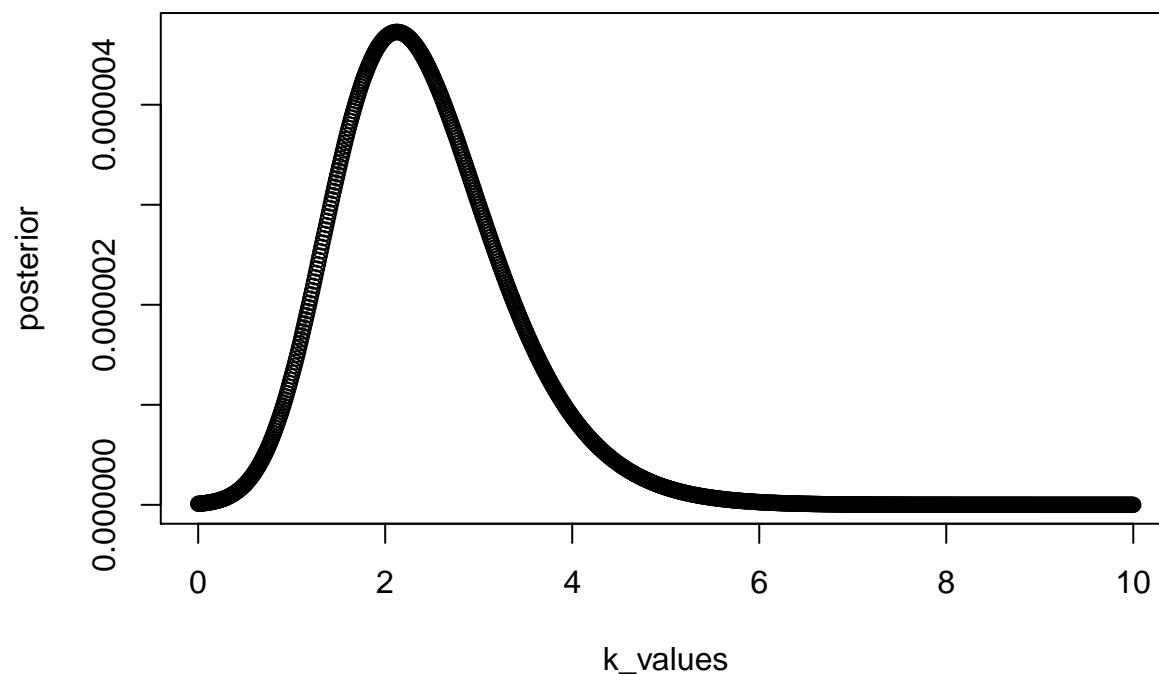
when compared to the density interval which captures more of the 'peak' area.

density.default(x = G)



3) von Mises distribution

- The plot below shows the posterior distribution of K over a sequence of k values.
- From this plot we can see the mode from this data is right above two. Checking this by seeing which k values correspond to the max of the posterior we get a result of 2.12 being the mode.



```

poster100 <- rbeta(100, 2+5, 2+15)
poster1000 <- rbeta(1000, 2+5, 2+15)
poster100000 <- rbeta(100000, 2+5, 2+15)
hist(poster100, 50, main = "100 Draws", xlab="")
hist(poster1000, 50, main = "1000 Draws", xlab="")
hist(poster100000, 50, main = "100000 Draws", xlab="")

a <- 2 + 5
b <- 2 + 15
standard_deviation <- (a*b) / ((a+b)^2 * (a+b+1) )
true_mean <- a / (a + b)
print(paste0("True standard deviation: ", standard_deviation))
print(paste0("True mean: ", true_mean))
true_prob <- pbeta(0.3, 2+5, 2+15, lower.tail = FALSE)
print(paste0("The true probability theta > 0.3: ", true_prob))
poster <- rbeta(10000, 2+5, 2+15)
sumnewsamples <- sum(poster > 0.3) / 10000
print(paste0("The sampled probability theta > 0.3: ", sumnewsamples))
phi <- log( poster / (1 - poster))
#density(phi)
hist(phi, freq = FALSE)
lines(density(phi))
#2
data <- c(44,25, 45, 52, 30, 63, 19, 50, 34, 67)
u <- 3.7
n <- length(data)
t <- sum((log(data) - u) ^ 2) / n
#simulating variance from an inverse chi squared distribtution
sigmasquared <- ((n-1)*t)/rchisq(10000, n-1)

library(LaplacesDemon)
theoretical <- rinvcchisq(1000, n, scale = t)

hist(sigmasquared, 100, xlim=c(0,1), freq = FALSE)
lines(density(theoretical))
#b
G <- 2 * pnorm(sqrt(sigmasquared)/sqrt(2)) - 1
hist(G, 100, freq = FALSE)
lines(density(G))
#c
equal_tail <- quantile(G, c(.05,.95))

#kernel density estimate
gdens <- density(G)

library(HDInterval)
#could not make code below equal to package results...
hdi <- hdi(gdens, 0.90)

sort_g <- sort(gdens$y, decreasing = TRUE)
#plot(sort_g)
#choose 90% of data from the density
#a <- gdens$y[gdens$y < .95 & gdens$y > .05 ]

```

```

#b <- gdens$y[gdens$y < quantile(gdens$y, .95) & gdens$y > quantile(gdens$y,.05) ]
options(scipen=999)
#gdens$x[max(b)]
#gdens$x[which.max(b)]
plot(density(G))
abline(v=equal_tail[1], col = 'red')
abline(v=equal_tail[2], col = 'red')
abline(v=hdi[1])
abline(v=hdi[2])
#3
#data <- c(40, 303, 326, 285, 296, 314, 20, 308, 299, 296)
radians <- c(-2.44, 2.14, 2.54, 1.83, 2.02, 2.33, -2.79, 2.23, 2.07, 2.02)

posteriorfunc <- function(kvals, data = radians){
  u <- 2.39
  Io <- bessell(kvals, 0)
  #likelihood x prior
  return( prod( (exp(kvals * cos(data - u))) / (2 * pi * Io) ) * dexp(kvals) )
}

options(scipen=999)
k_values <- seq(0,10,0.01)
posterior <- c()
for(x in 1:length(k_values)){
  posterior[x] <- posteriorfunc(k_values[x]) #function defaults to radians data if not supplied
}
plot(x= k_values, y = posterior)
#k_values[which.max(posterior)]

```