Chapter2_BetaBinomial

R Beta Distribution

Beta distribution

$$\text{Beta}(\theta;\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1} \tag{1}$$

In R this function is $dbeta(\alpha,\beta)$.

Beta function

$$B(\alpha,\beta) = \int_0^1 \theta^{\alpha-1}(1-\theta)^{\beta-1}d\theta = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)} \tag{2}$$

Expected Value of Beta distribution

$$E[\theta] = \int_0^1 \theta\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}d\theta = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\int_0^1 \theta\theta^{\alpha-1}(1-\theta)^{\beta-1}d\theta = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\int_0^1 \theta^{\alpha}(1-\theta)^{\beta-1}d\theta \tag{3}$$

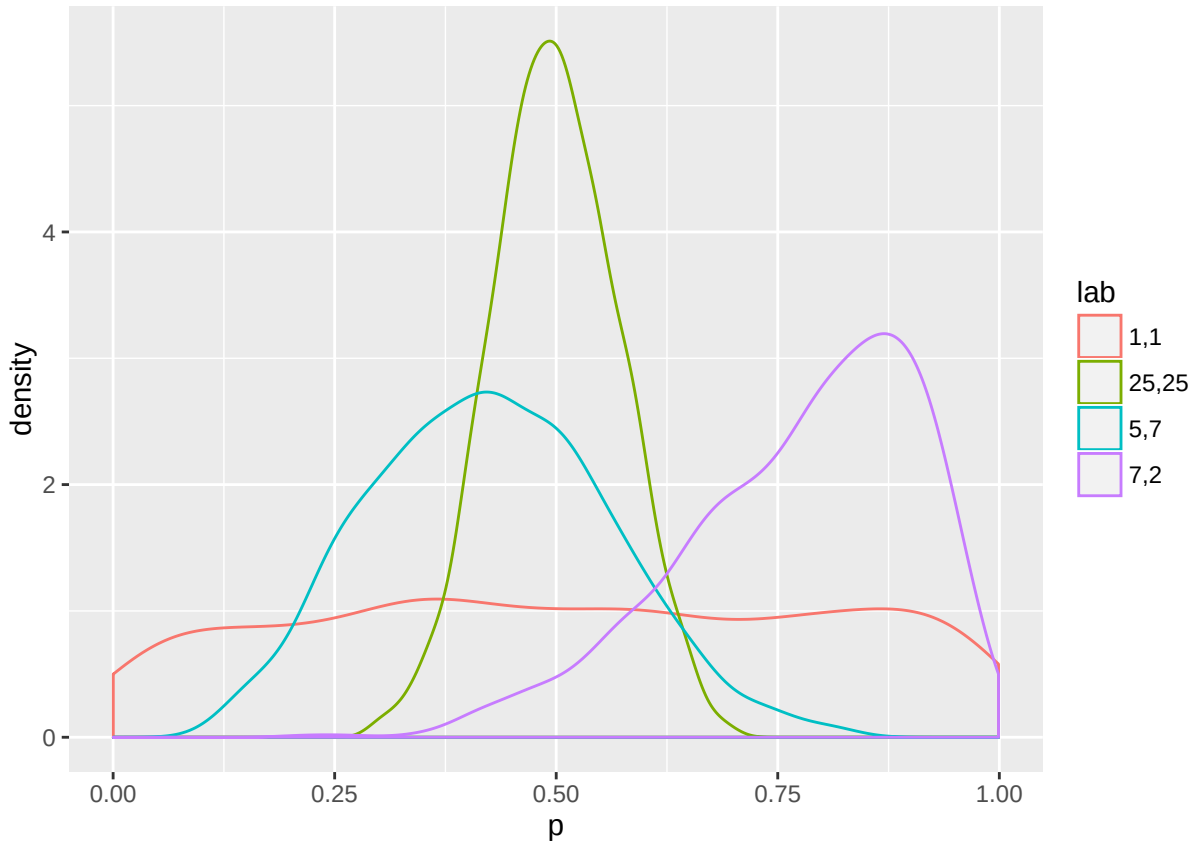
Where $\Gamma(n) = (n-1)!$

$$E[\theta] = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\frac{\Gamma(\alpha+1)\Gamma(\beta)}{\Gamma(\alpha+\beta+1)} = \frac{\alpha}{\alpha+\beta} \tag{4}$$

Variance of Beta

$$Var[\theta] = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} \tag{5}$$

```
n = 1000
p1 = rbeta(n,1,1)
p2 = rbeta(n,25,25)
p3 = rbeta(n,5,7)
p4 = rbeta(n,7,2)
p = c(p1,p2,p3,p4)
lab = factor(c(rep("1,1",n),rep("25,25",n),rep("5,7",n),rep("7,2",n)))
ggplot(data.frame()) + geom_density(aes(x=p,color=lab))
```

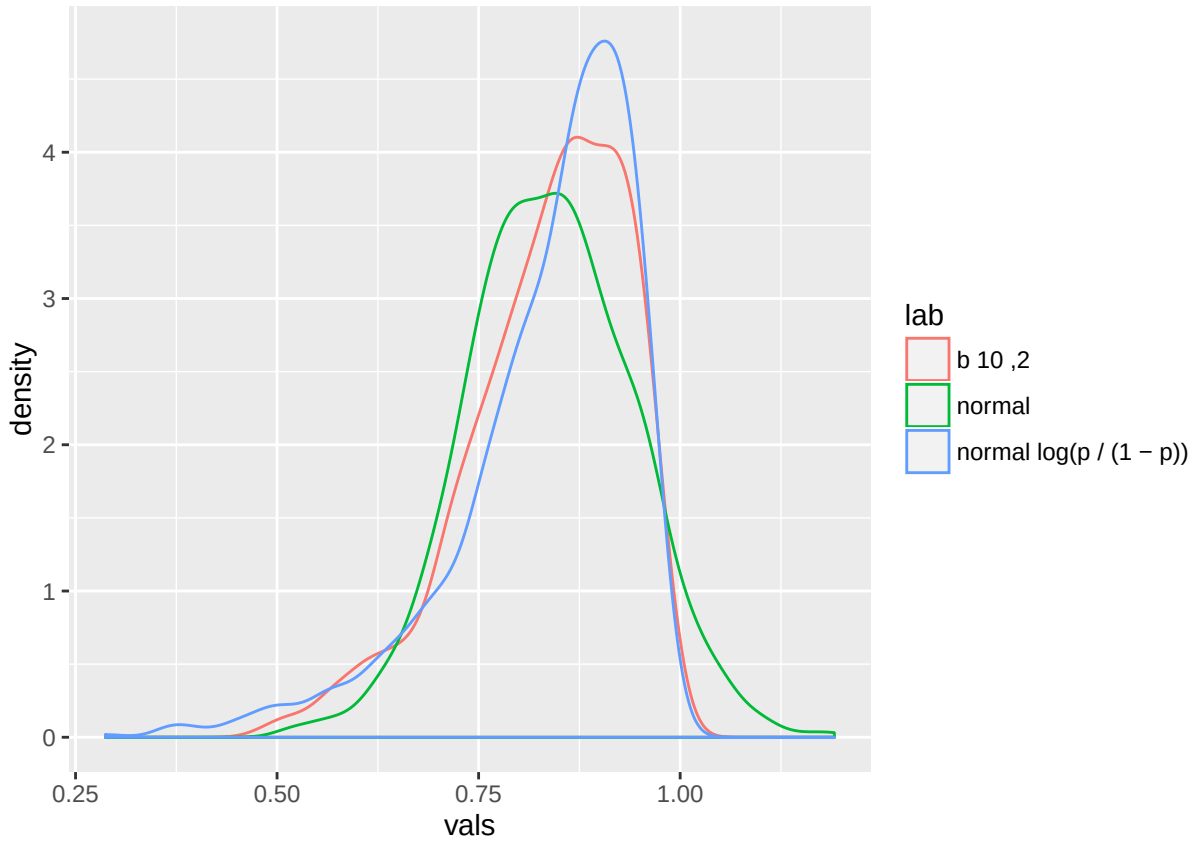


In R this function is $beta(\alpha, \beta)$.

Fitting a beta distribution using a normal approximation.

Using an $Beta(\theta, 7, 2)$, let us try to fit a normal distirbution to it.

```
n = 1000
p = rbeta(n,10,2)
mu = mean(p)
v = var(p)
x = rnorm(n,mean = mu, sd = sqrt(v))

p_trans = log(p / (1-p) )
mu_p_trans = mean(p_trans)
v_p_trans = var(p_trans)

x2 = rnorm(n,mean = mu_p_trans, sd = sqrt(v_p_trans))
x2_untrans = exp(x2) / (1 + exp(x2))

vals = c(p,x,x2_untrans)
lab = c(rep("b 10 ,2",n),rep("normal",n), rep("normal log(p / (1 - p))",n))
ggplot(data.frame()) + geom_density(aes(x=vals,color=lab))
```



The normal approximation using the logit transform $log(p/(1-p))$ works better than just doing a simple normal transformation.

Binomial likelihood with Beta prior over θ has a Beta posterior over θ

A binomial likelihood with y successes from n trials is proportional to

$$p(y|\theta) \propto \theta^y (1-\theta)^{n-y} \tag{6}$$

The prior of θ is proportional to

$$p(\theta) \propto \theta^{\alpha-1} (1-\theta)^{\beta-1} \tag{7}$$

which is a beta distribution with parameters α and β: $\theta \sim \text{Beta}(\alpha, \beta)$. This suggests the prior density corresponds to $\alpha - 1$ successes and $\beta - 1$ failures.

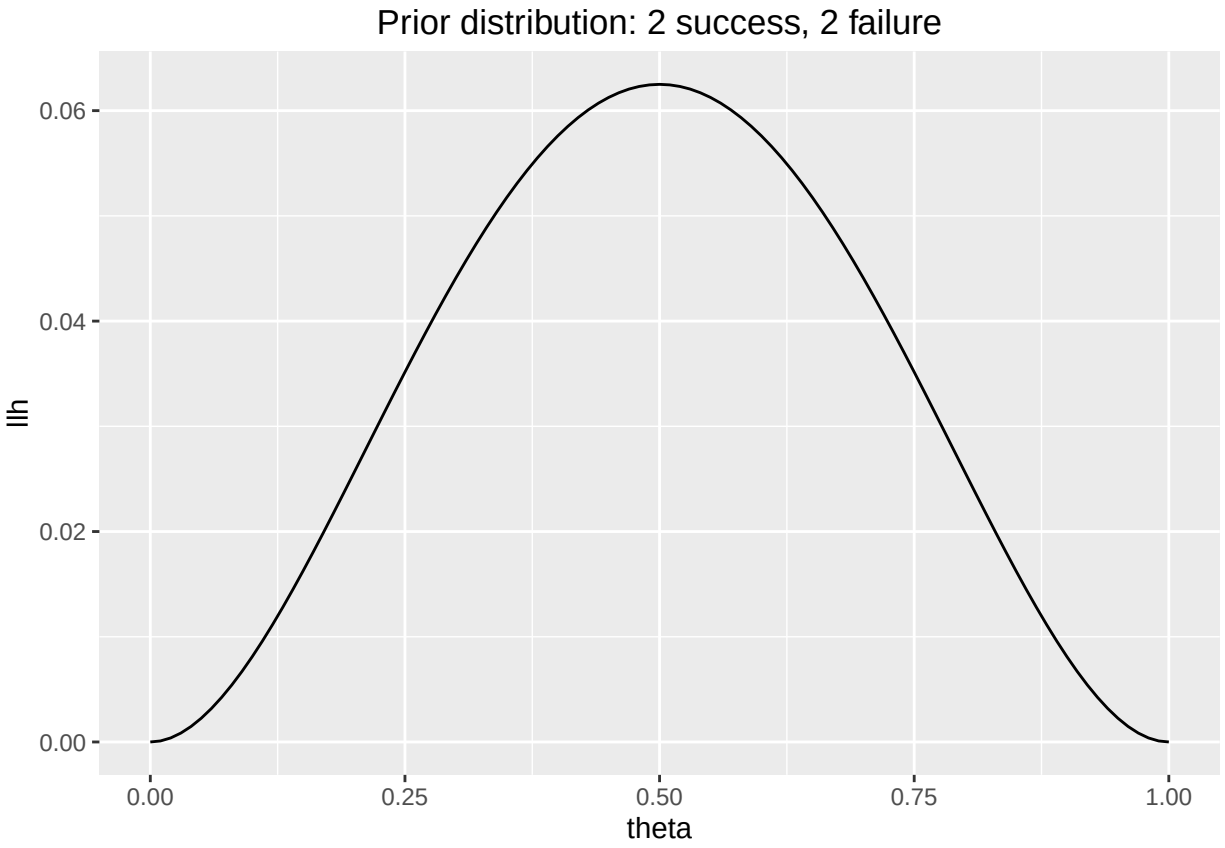
The postierior distribution $p(\theta|y)$ is proportional to

$$p(\theta|y) \propto \theta^y (1-\theta)^{n-y} \theta^{\alpha-1} (1-\theta)^{\beta-1} = \theta^{y+\alpha-1} (1-\theta)^{n-y+\beta-1} \tag{8}$$

The posterior distribution is thus represented by $\text{Beta}(\theta|\alpha + y, \beta + n - y)$

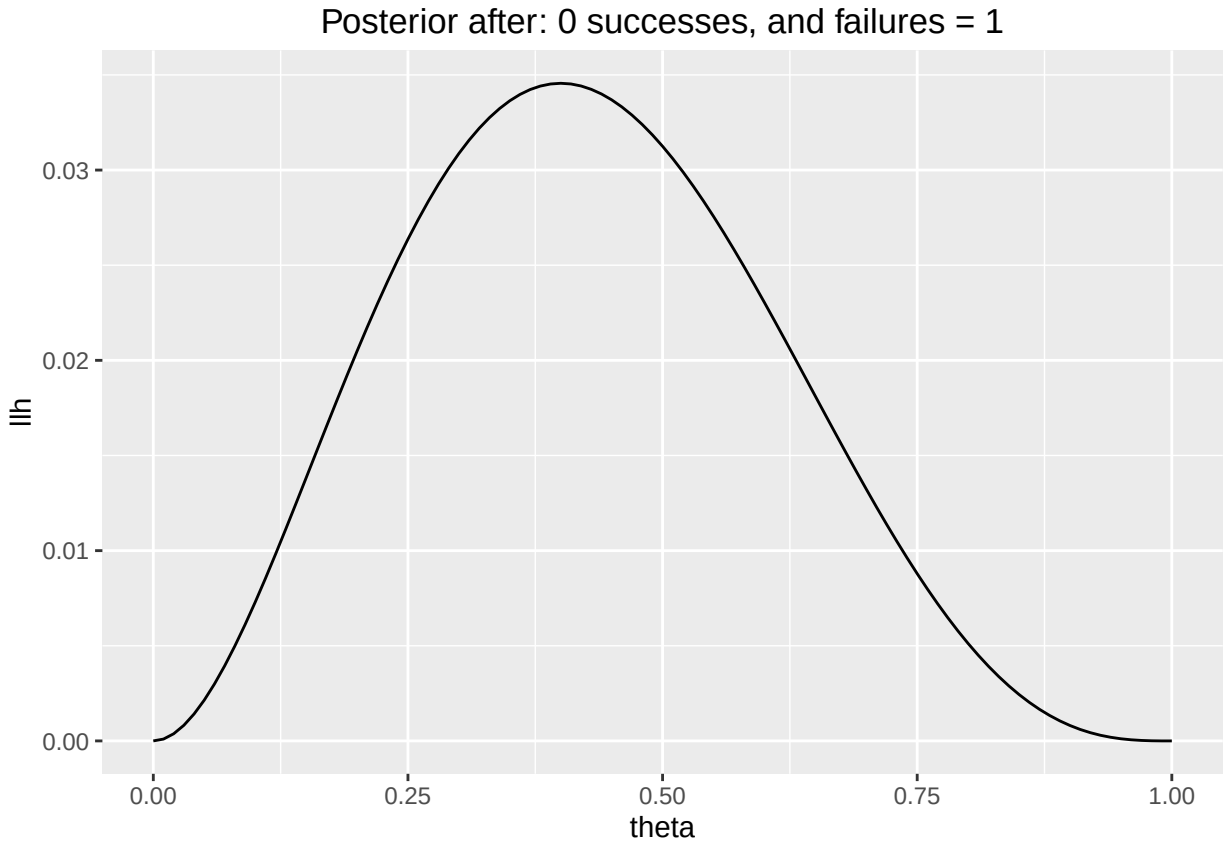
Example of how posterior changes as we get more evidence

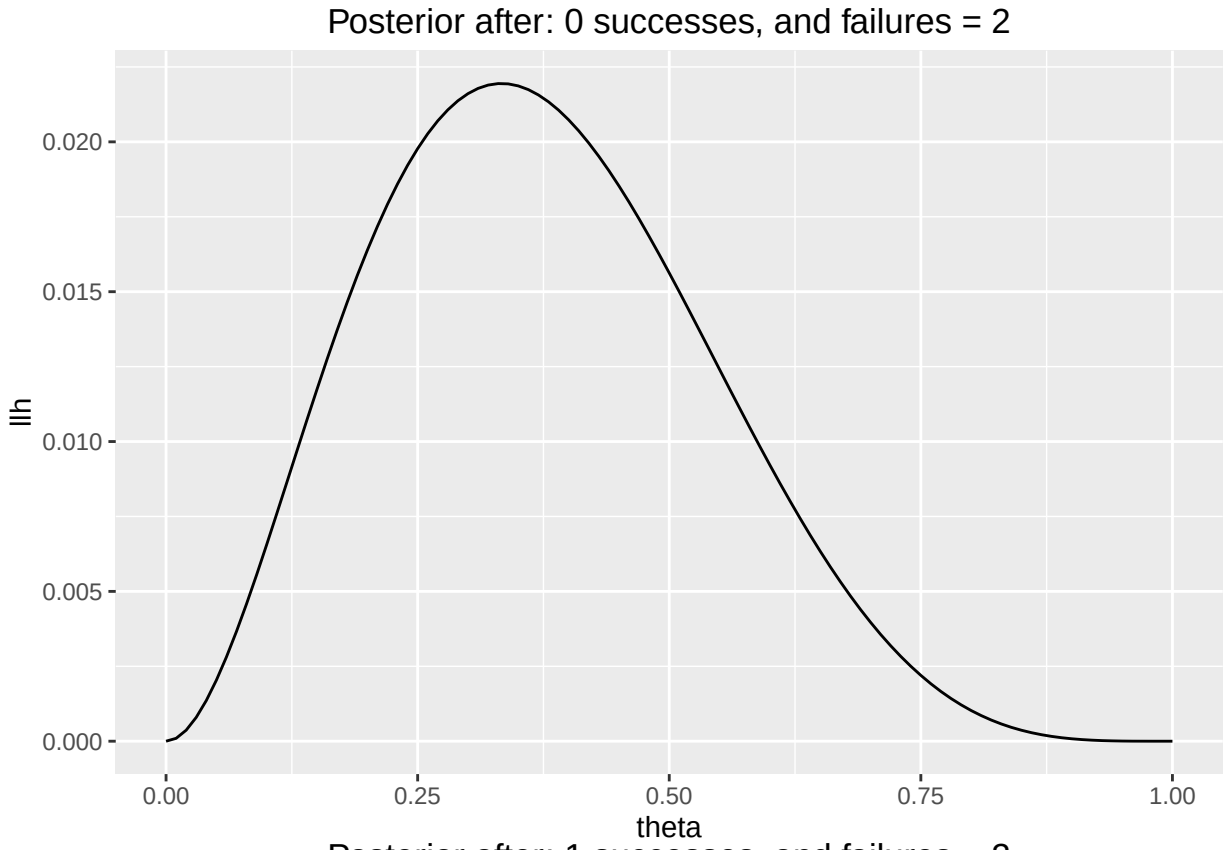
```
# prior
alpha = 3
beta = 3

beta_llh = function(theta,alpha,beta) {
  theta^(alpha - 1) * (1 - theta)^(beta - 1)
}

thetas = seq(0,1,0.01)
df.prior = data.frame(llh = beta_llh(thetas,alpha,beta),theta=thetas)
ggplot(df.prior) + geom_line(aes(x=theta,y=llh)) + ggtitle("Prior distribution: 2 success, 2 failure")
```

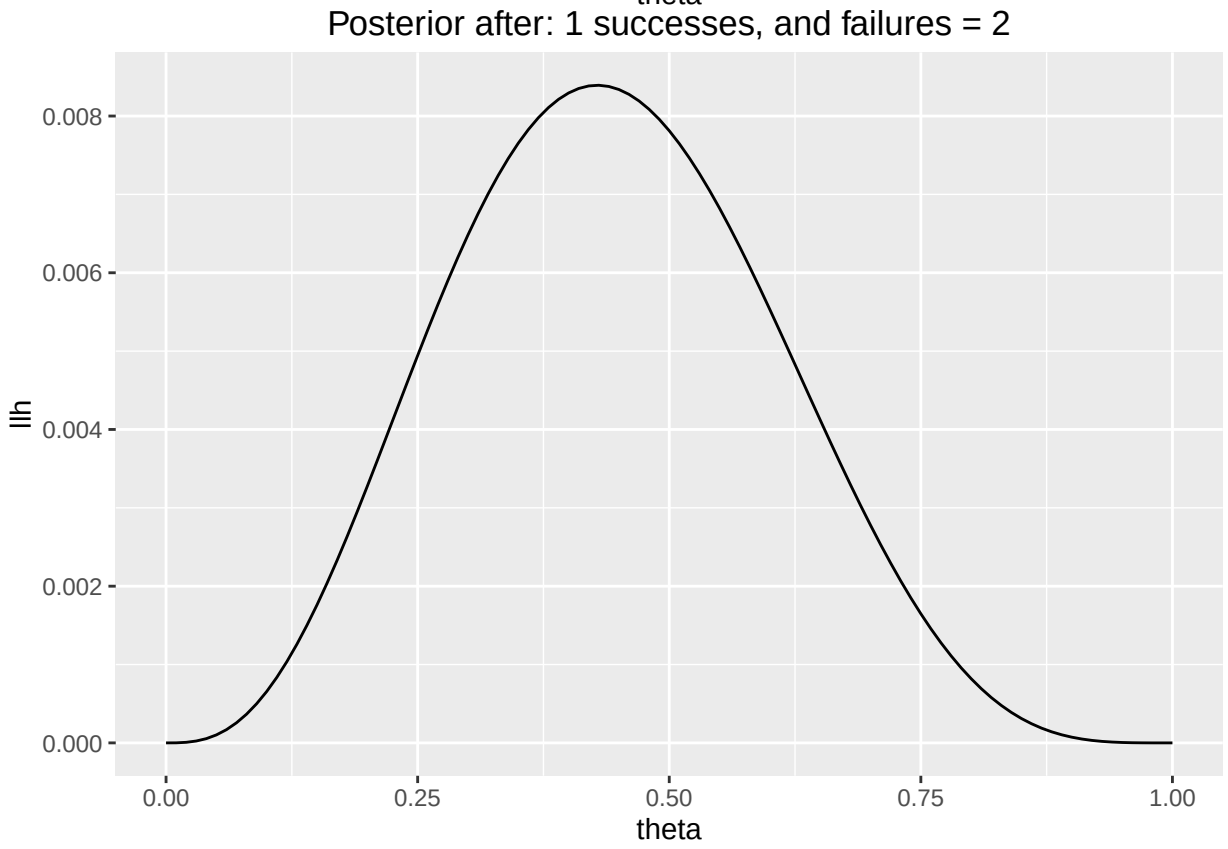


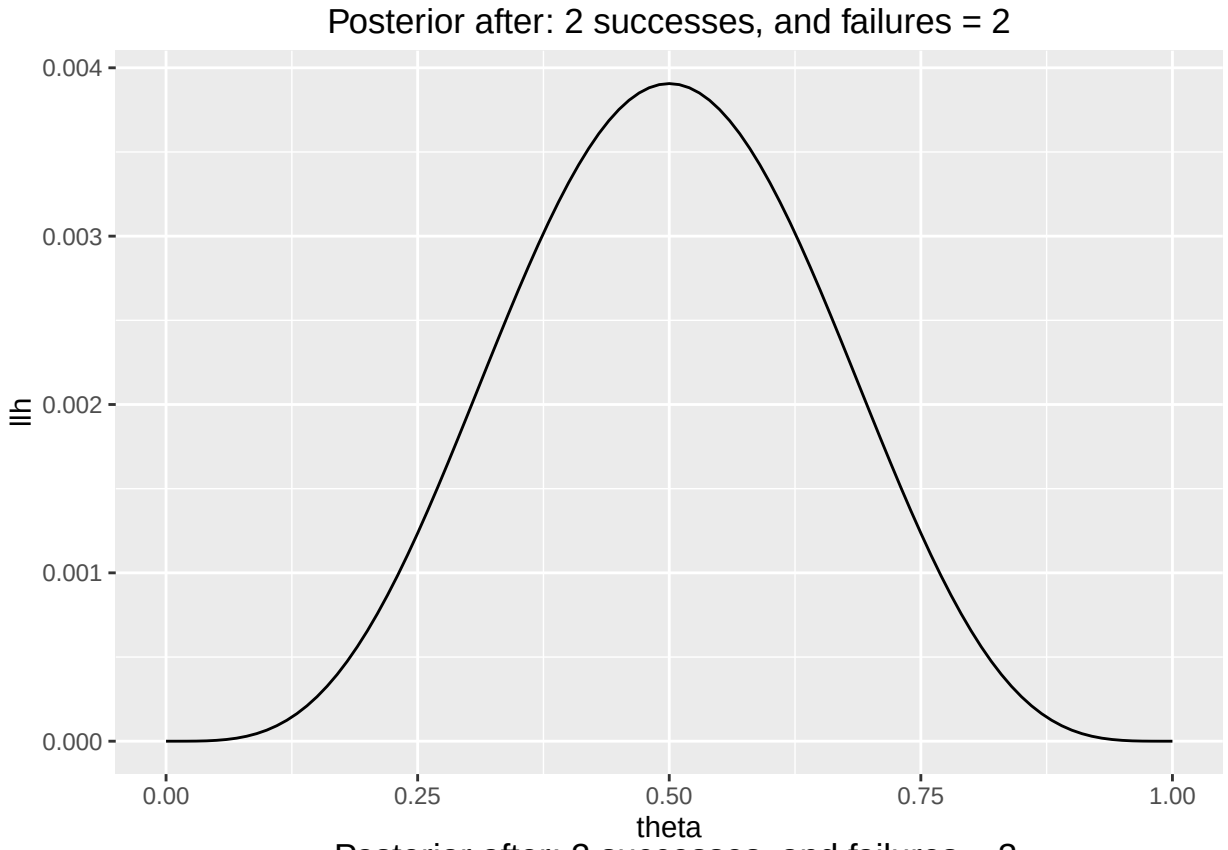

```
data = rbinom(20,prob = 0.7, size = 1)
successes = 0
failures = 0
for(d in data) {
  if(d == 1) {
    successes = successes + 1
    alpha = alpha + 1
  } else {
    failures = failures + 1
    beta = beta + 1
  }
  df.posterior = data.frame(llh = beta_llh(thetas,alpha,beta),theta=thetas)
  gplot = ggplot(df.posterior) + geom_line(aes(x=theta,y=llh)) + ggtitle(paste0("Posterior after: ",suc
```

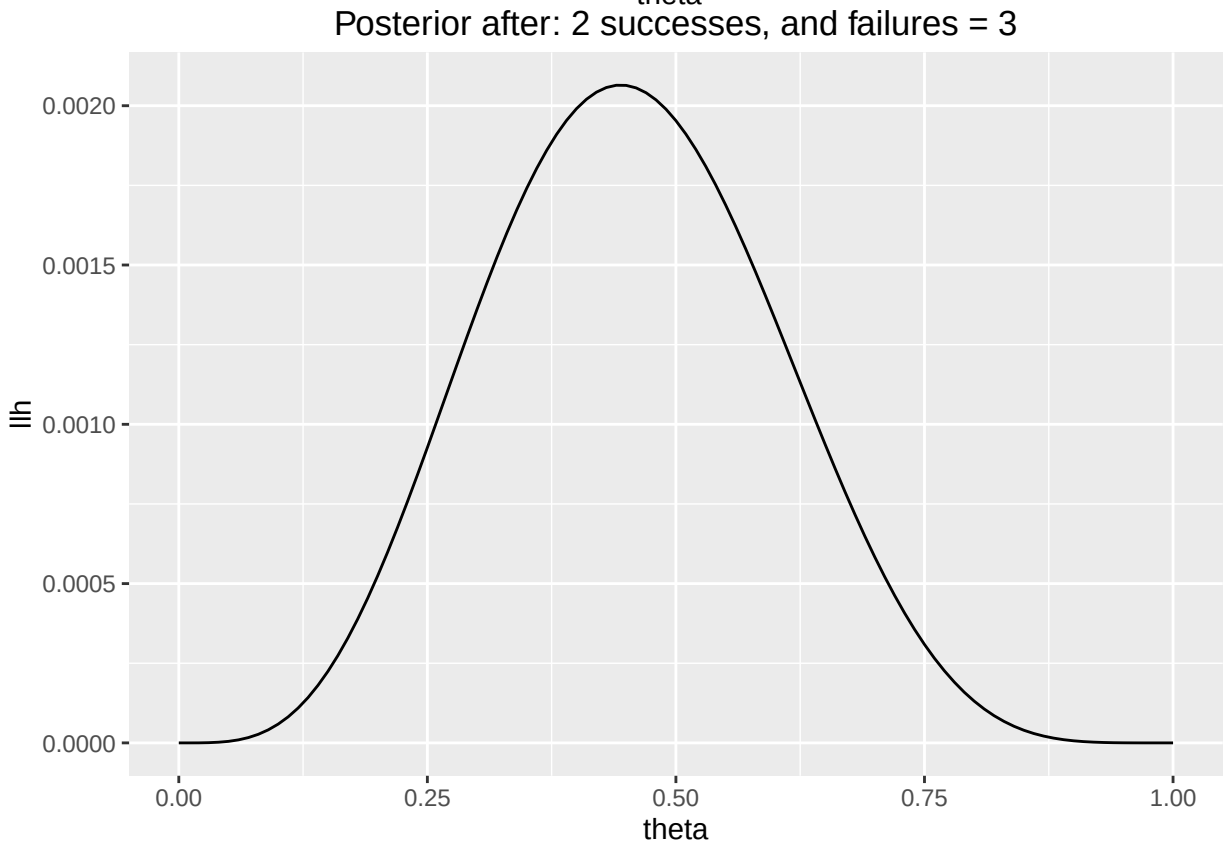
```
  print(gplot)
}
```

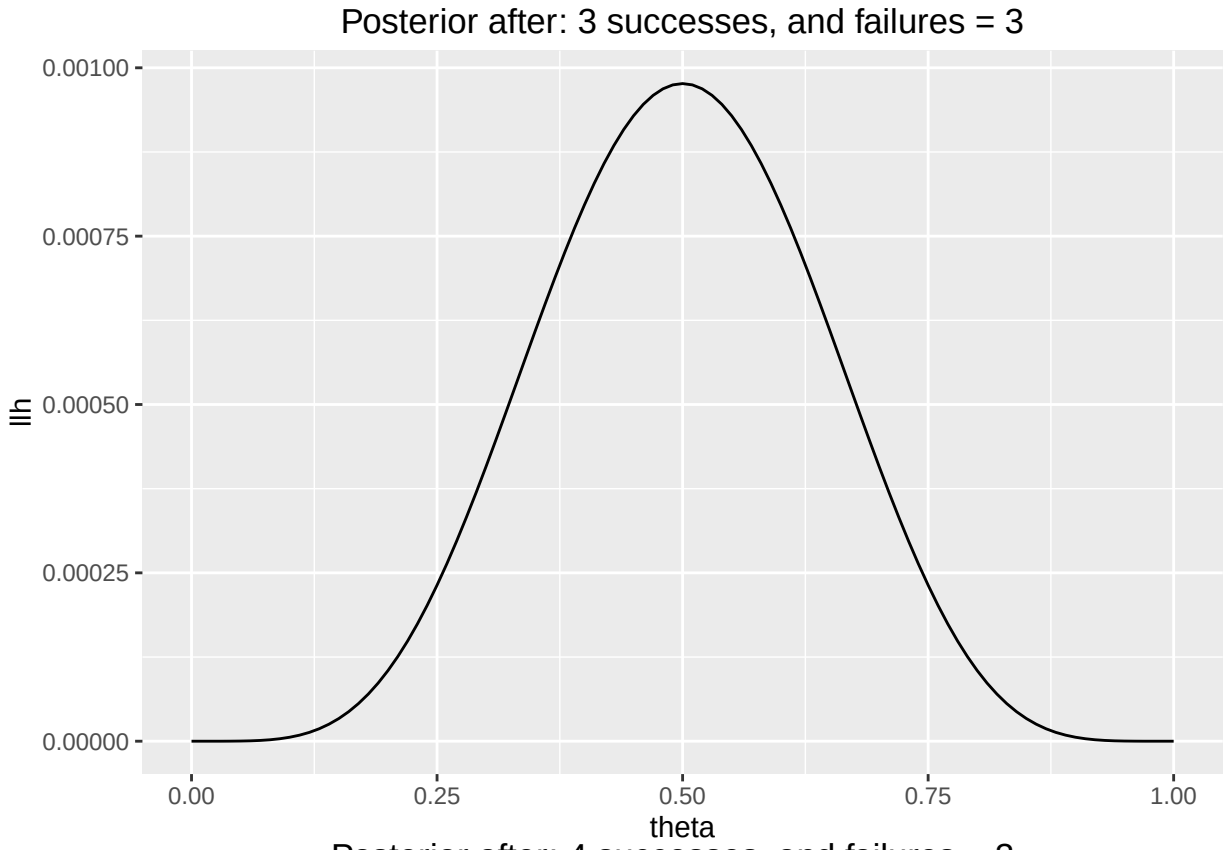

Posterior after: 0 successes, and failures = 1
0.03
0.02
0.01
0.00
llh
0.00
0.25
0.50
0.75
1.00
theta

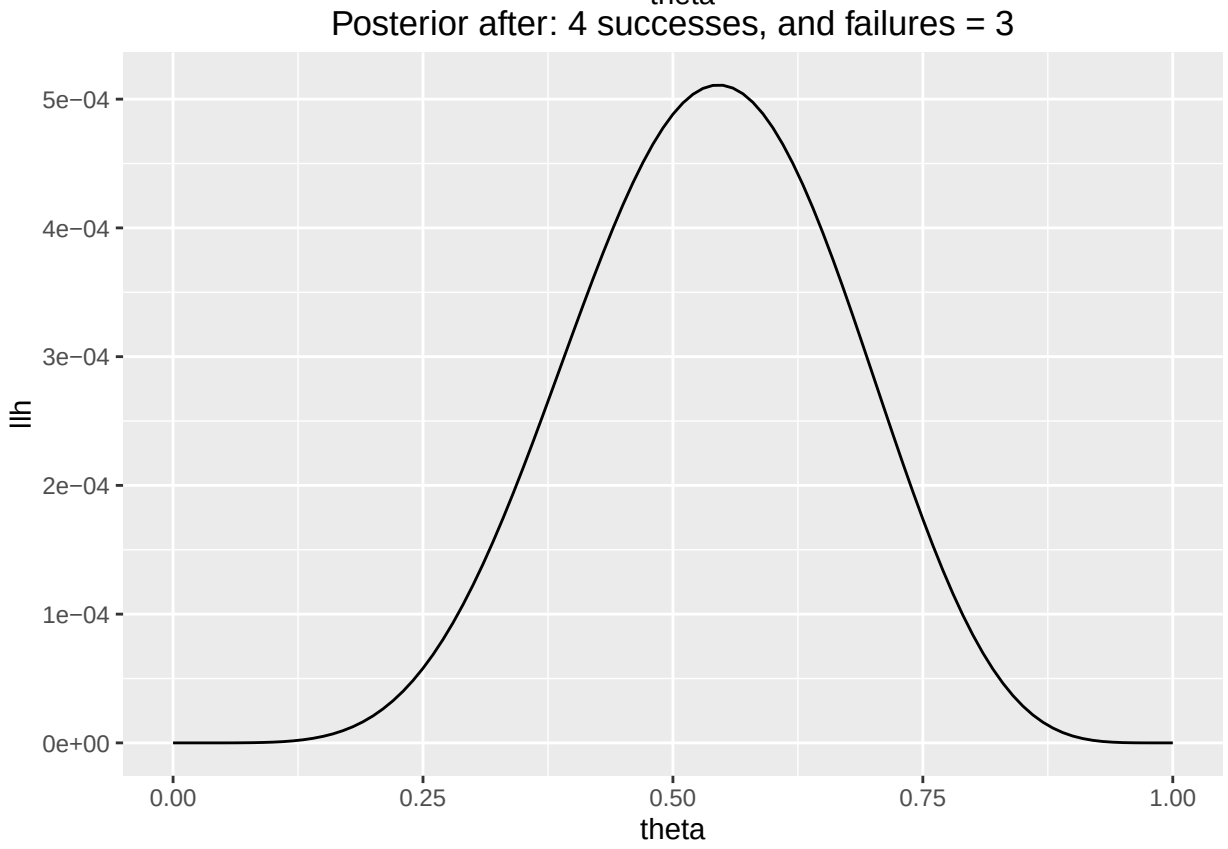

Posterior after: 0 successes, and failures = 2
0.020
0.015
0.010
0.005
0.000
llh
0.00
0.25
0.50
0.75
1.00
theta

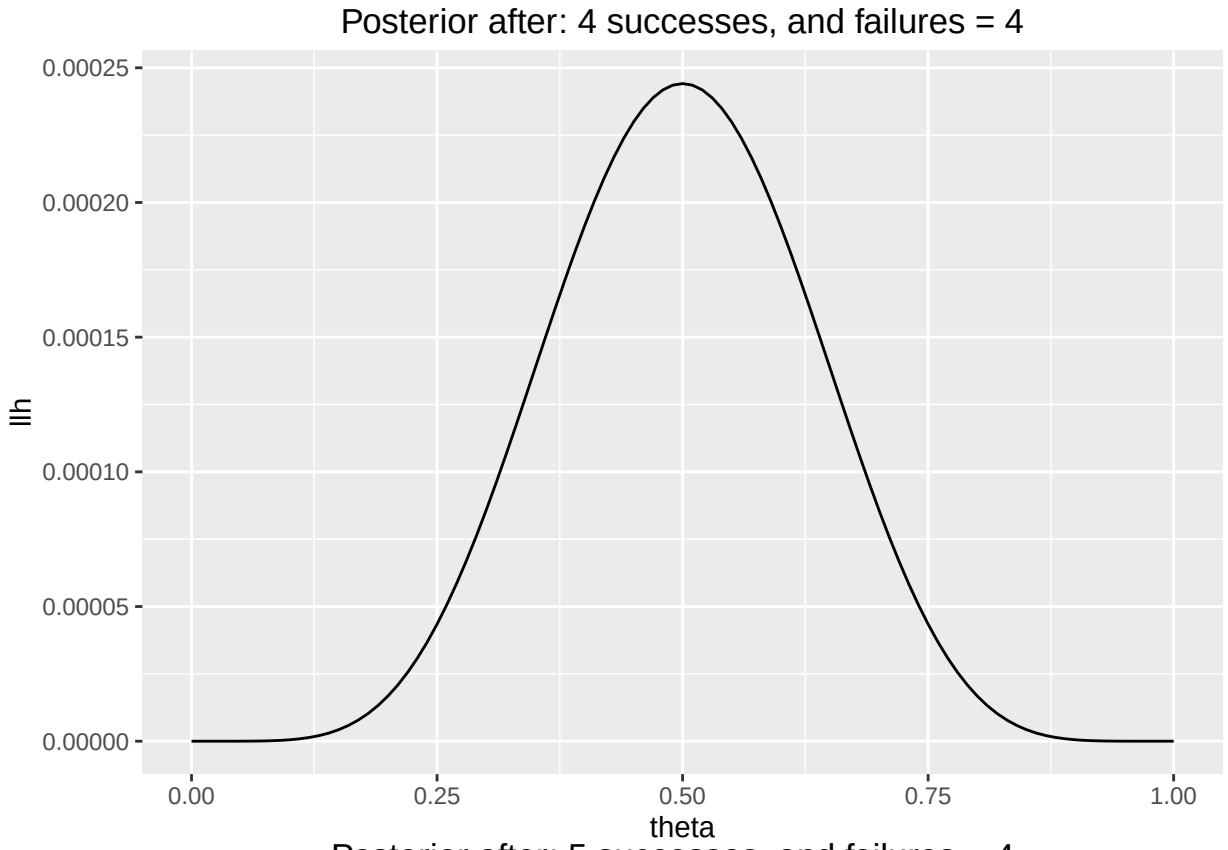

Posterior after: 1 successes, and failures = 2
0.008
0.006
0.004
0.002
0.000
llh
0.00
0.25
0.50
0.75
1.00
theta

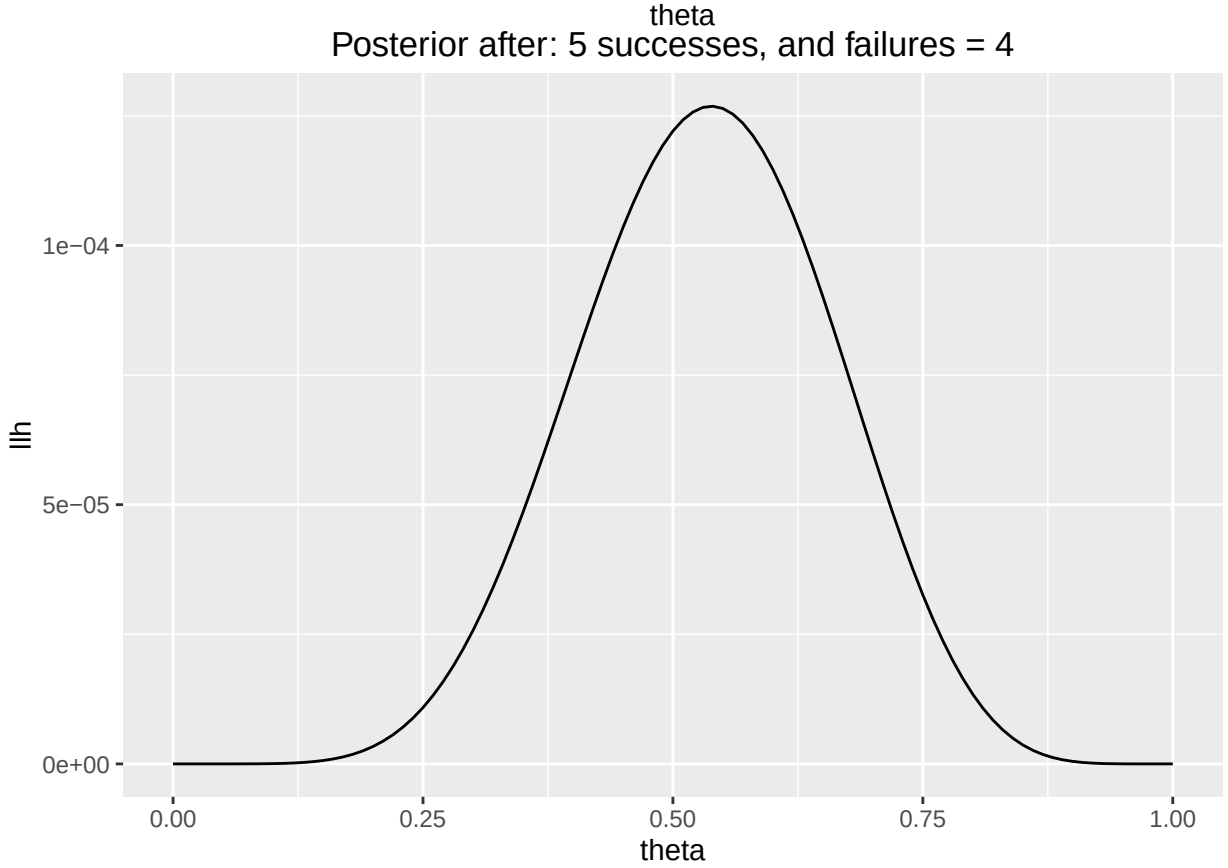

Posterior after: 2 successes, and failures = 2
0.004
0.003
0.002
0.001
0.000
llh
0.00
0.25
0.50
0.75
1.00
theta


Posterior after: 2 successes, and failures = 3
0.0020
0.0015
0.0010
0.0005
0.0000
llh
0.00
0.25
0.50
0.75
1.00
theta

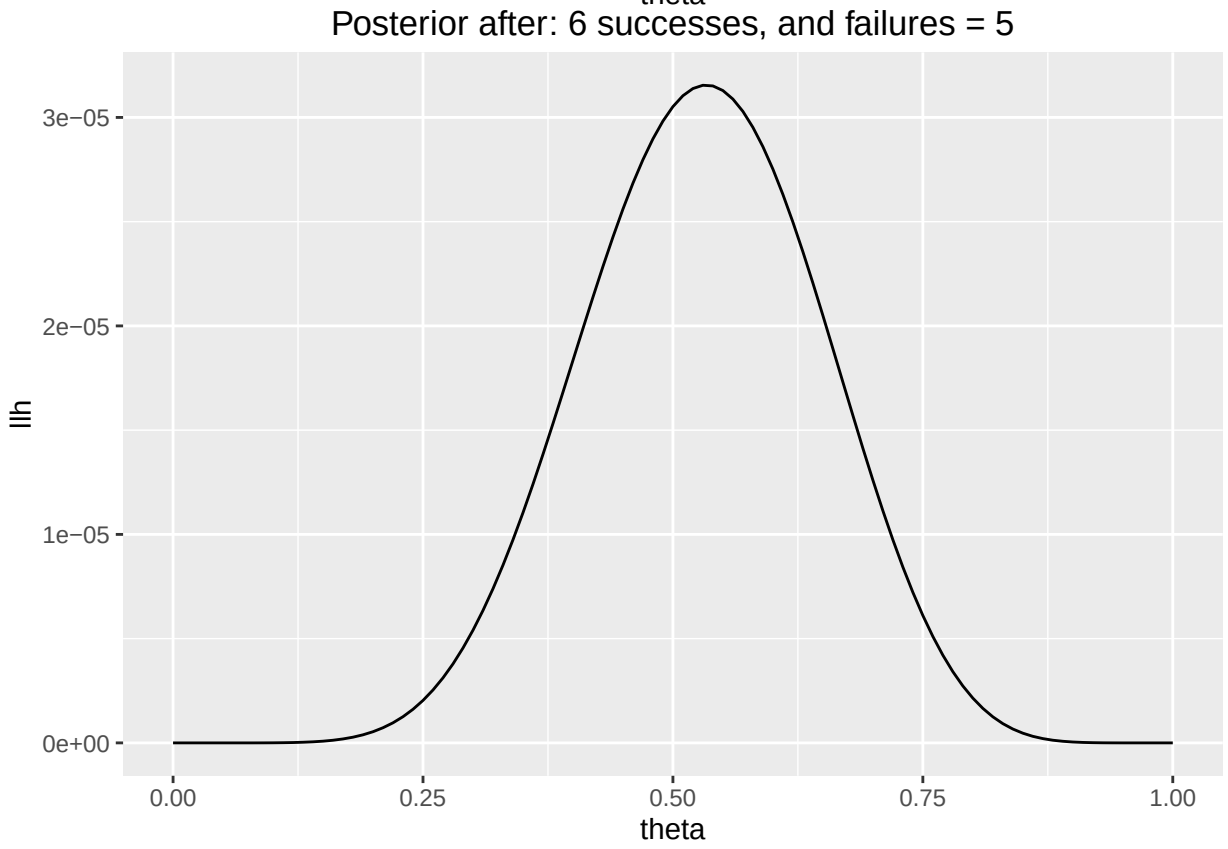

Posterior after: 3 successes, and failures = 3
0.00100
0.00075
0.00050
0.00025
0.00000
llh
0.00
0.25
0.50
0.75
1.00
theta

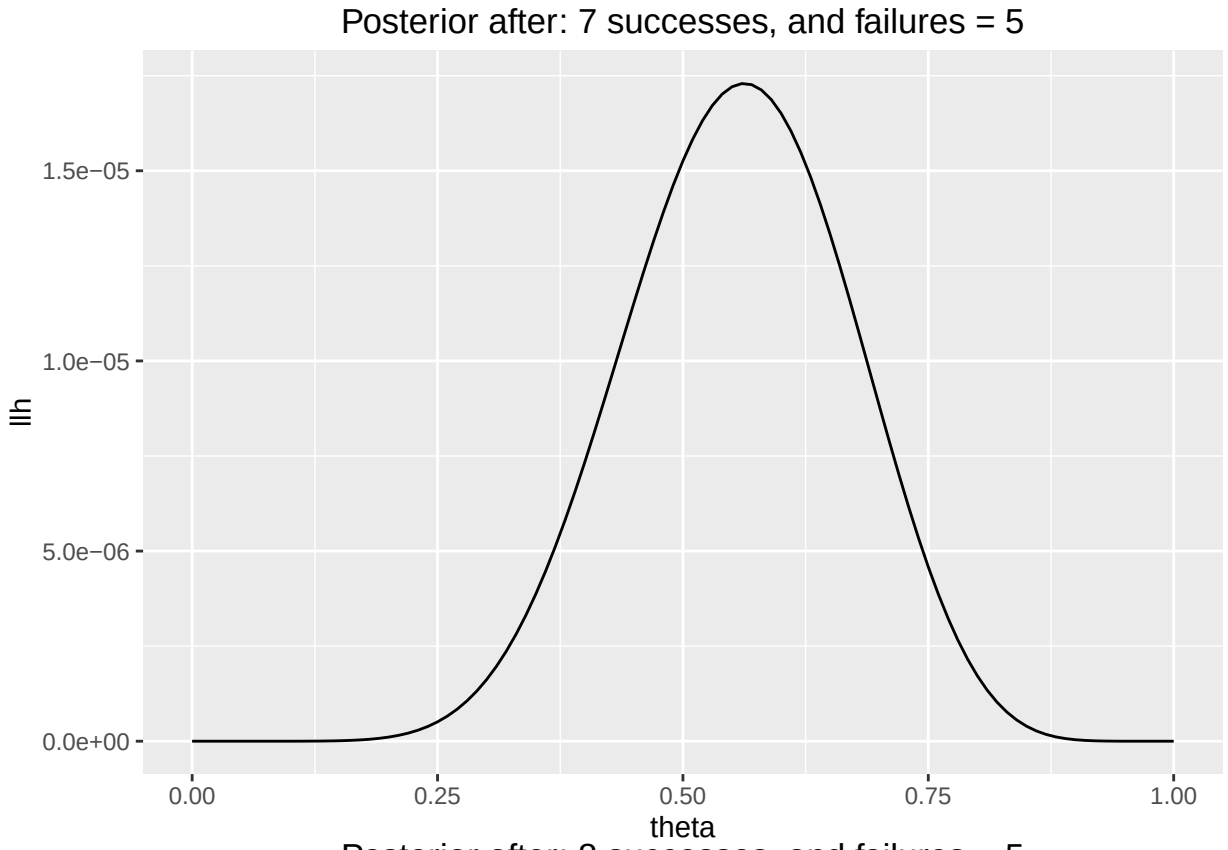

Posterior after: 4 successes, and failures = 3
5e−04
4e−04
3e−04
2e−04
1e−04
0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta

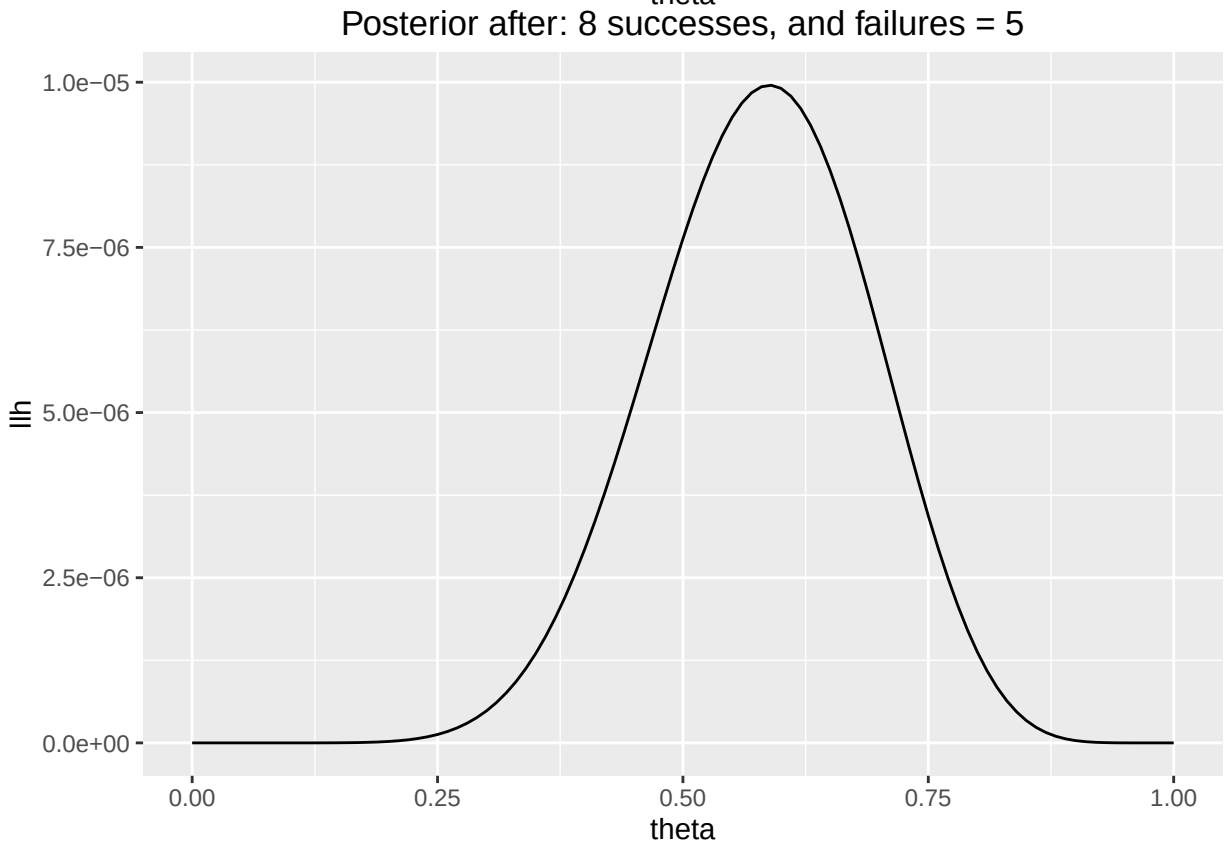

Posterior after: 4 successes, and failures = 4
0.00025
0.00020
0.00015
0.00010
0.00005
0.00000
llh
0.00
0.25
0.50
0.75
1.00
theta

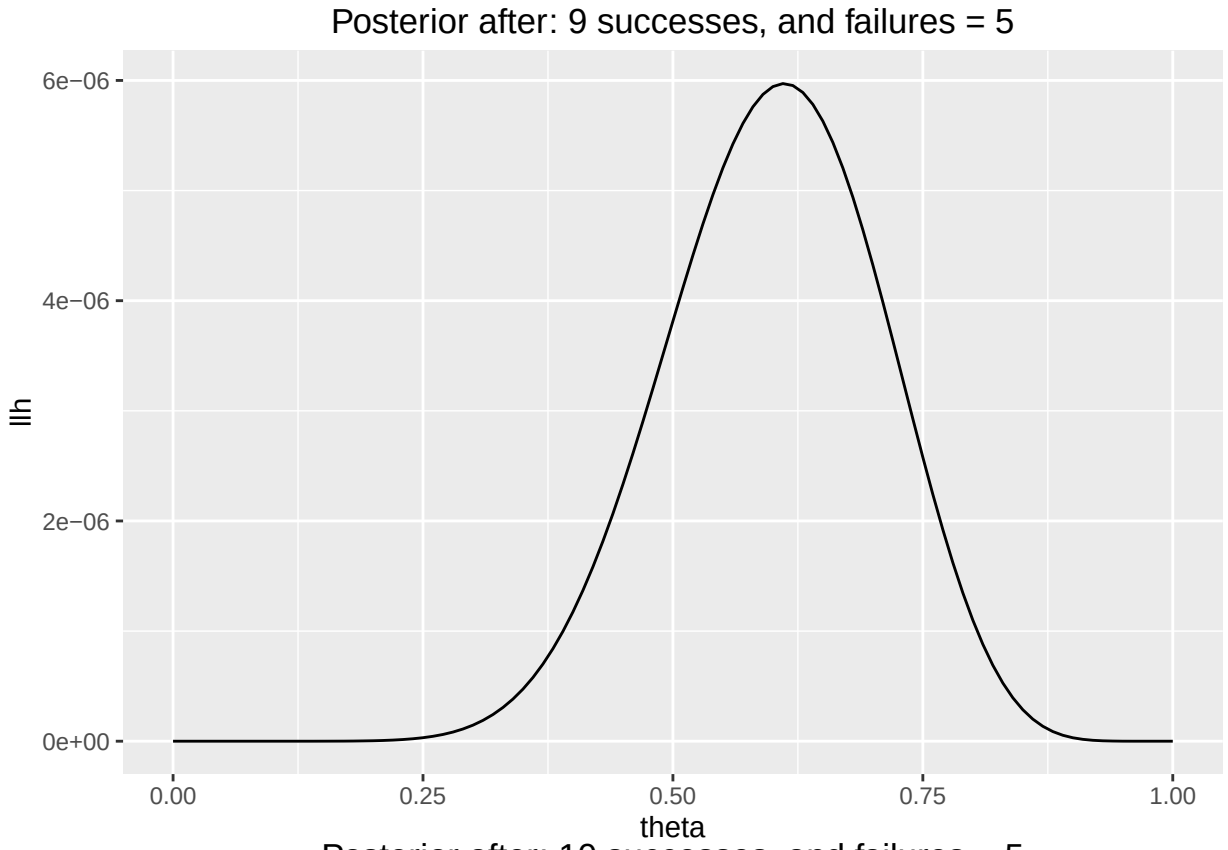

Posterior after: 5 successes, and failures = 4
1e−04
5e−05
0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta

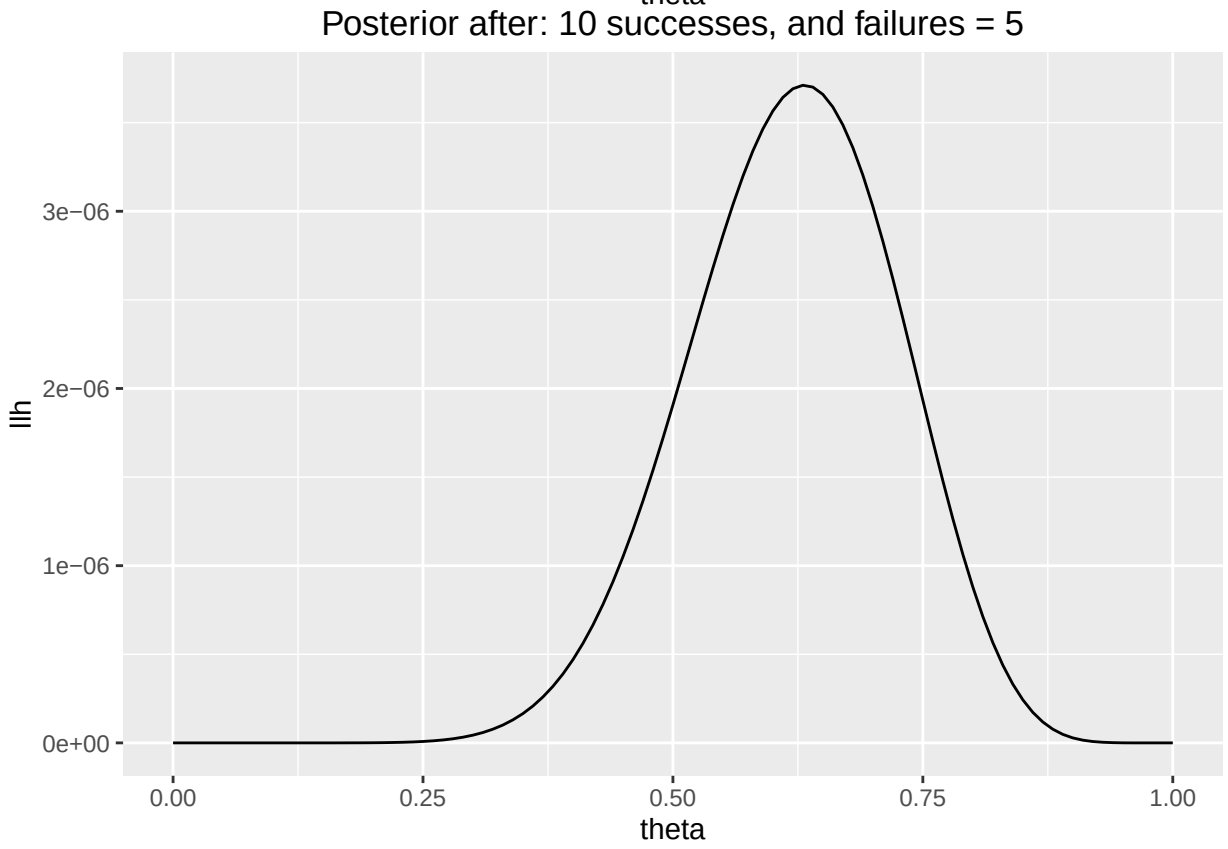

Posterior after: 5 successes, and failures = 5
6e−05
4e−05
2e−05
0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta

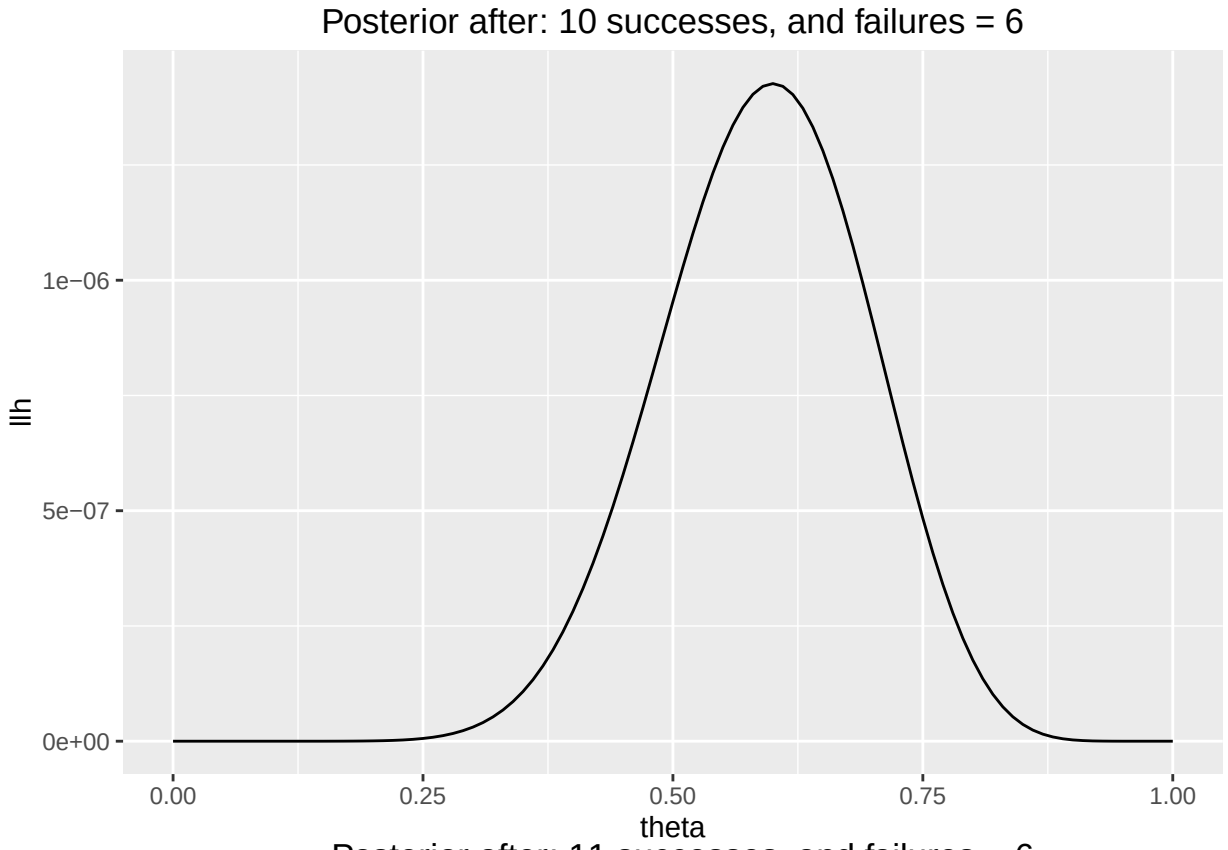

Posterior after: 6 successes, and failures = 5
3e−05
2e−05
1e−05
0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta

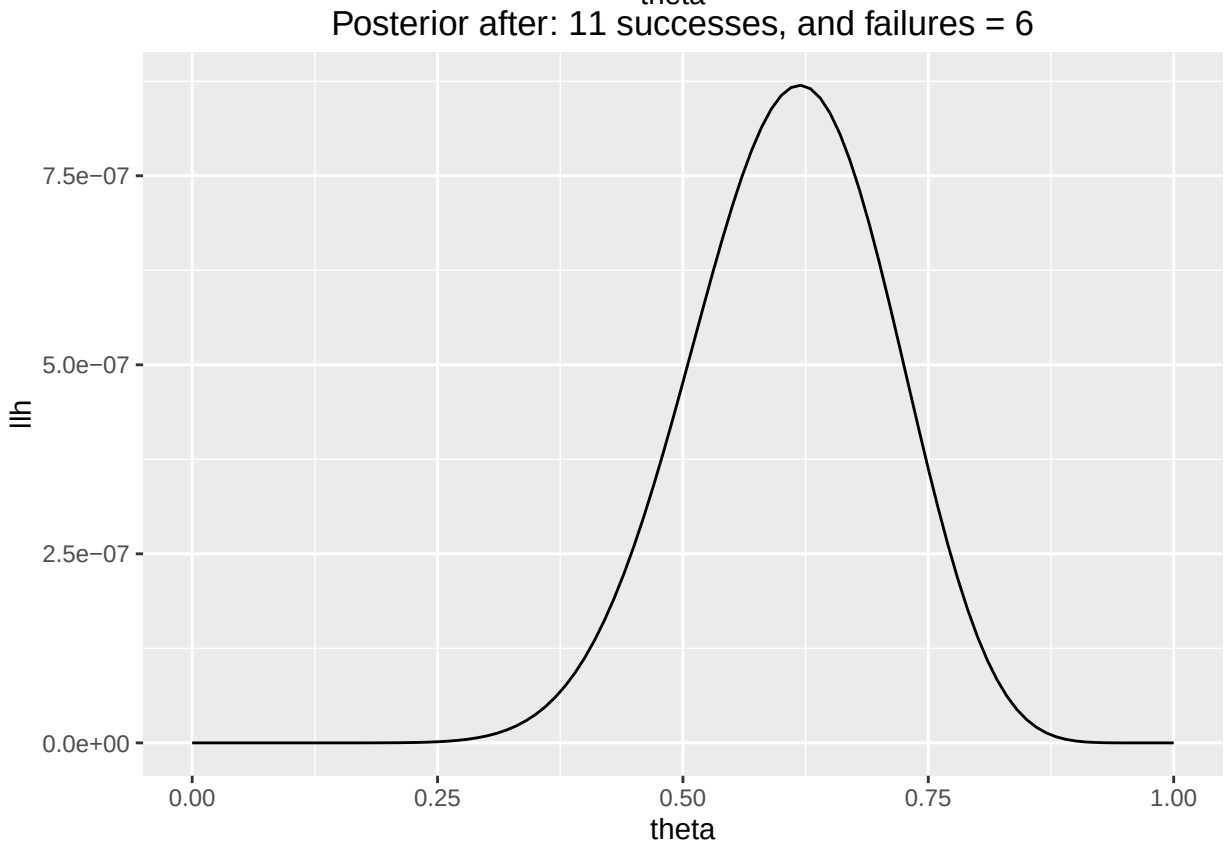

Posterior after: 7 successes, and failures = 5
1.5e−05
1.0e−05
5.0e−06
0.0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta

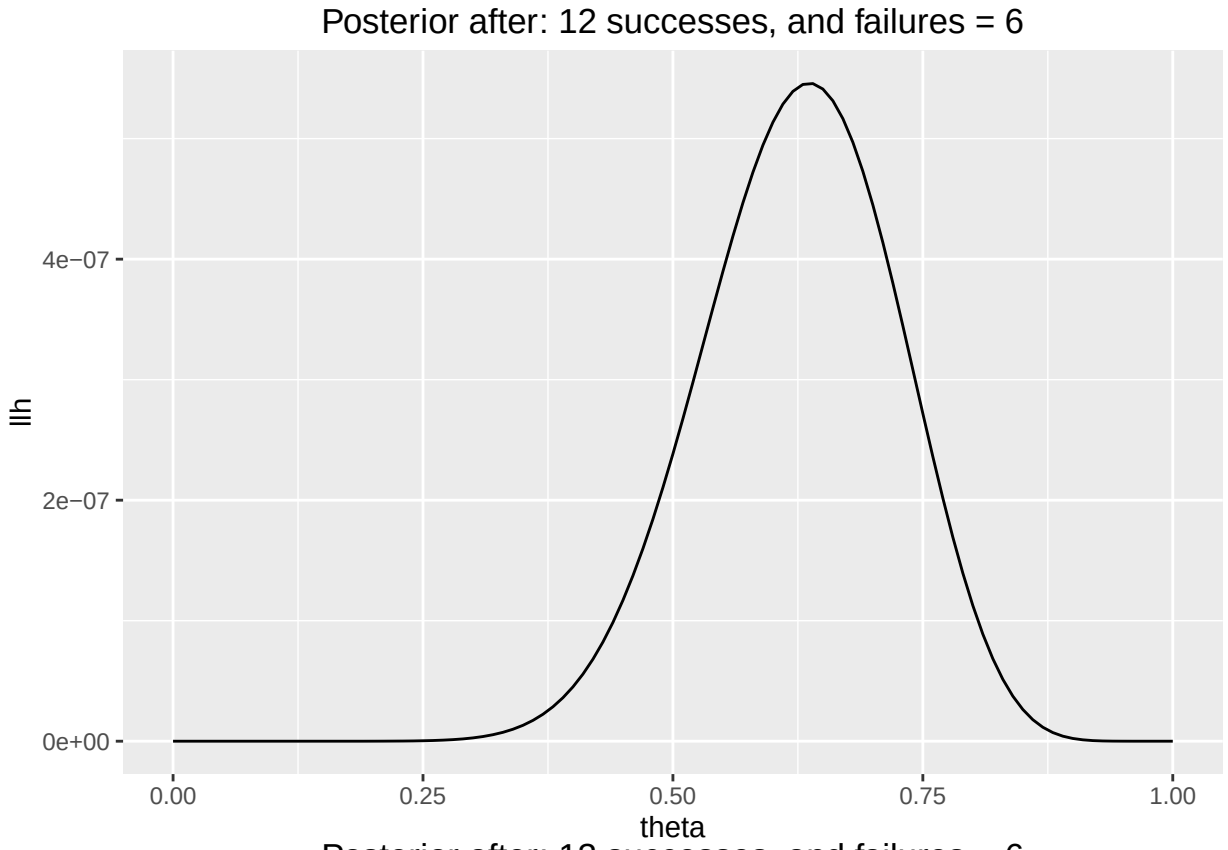

Posterior after: 8 successes, and failures = 5
1.0e−05
7.5e−06
5.0e−06
2.5e−06
0.0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta

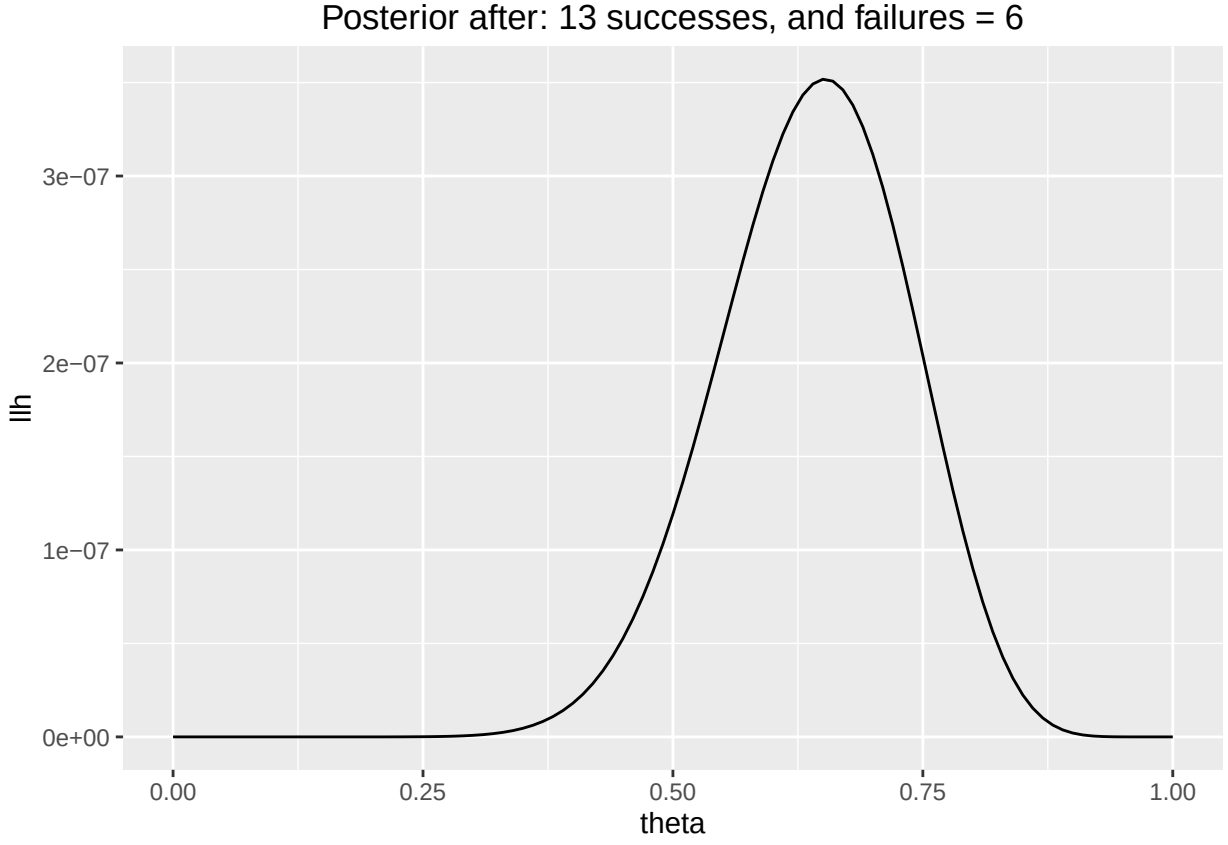

Posterior after: 9 successes, and failures = 5
6e−06
4e−06
2e−06
0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta

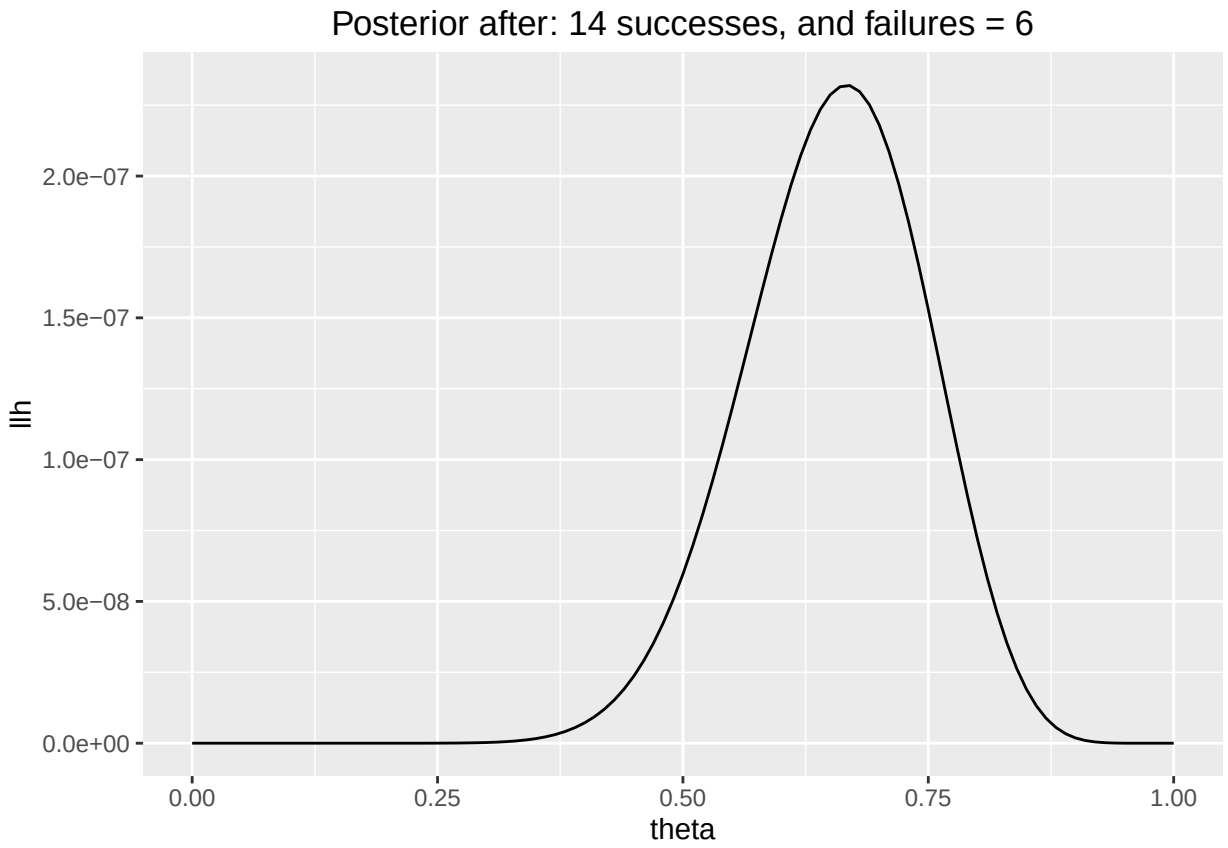

Posterior after: 10 successes, and failures = 5
3e−06
2e−06
1e−06
0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta


Posterior after: 10 successes, and failures = 6
1e−06
5e−07
0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta

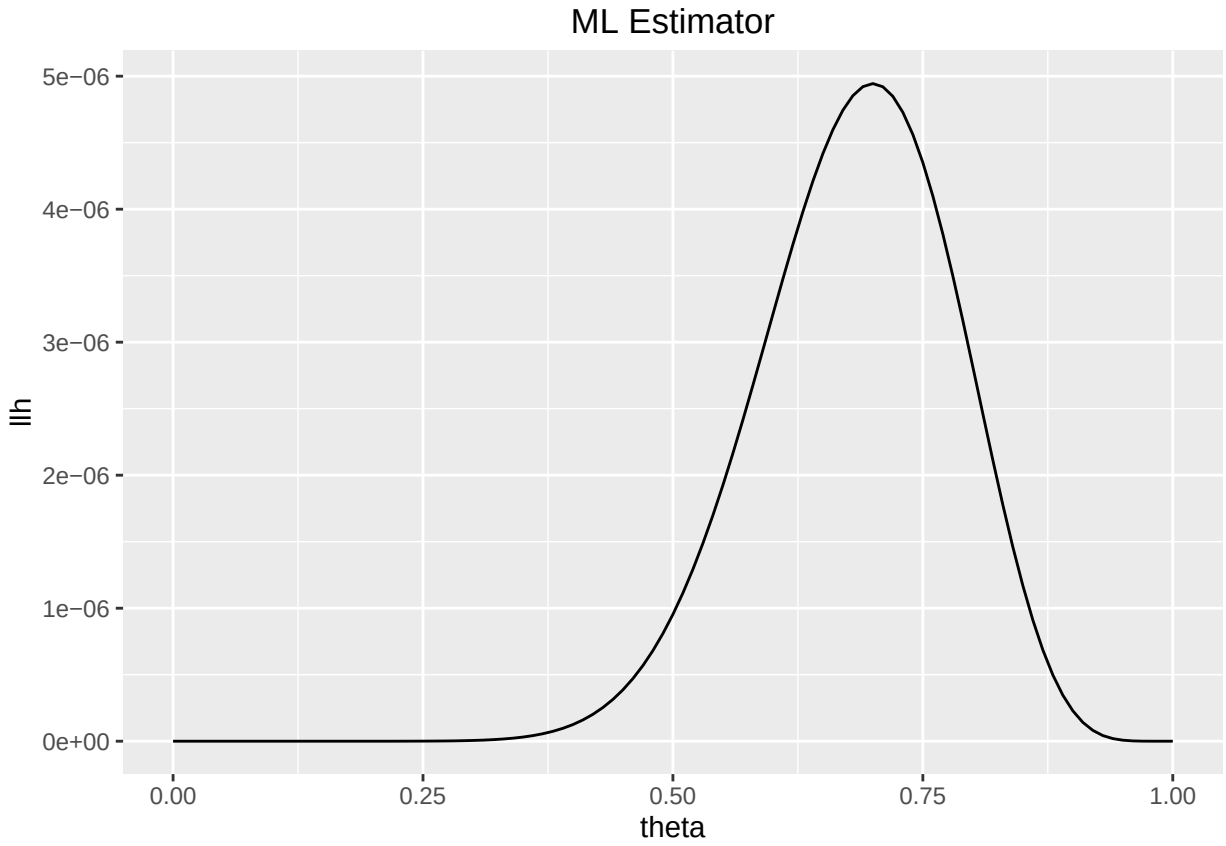

Posterior after: 11 successes, and failures = 6
7.5e−07
5.0e−07
2.5e−07
0.0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta


Posterior after: 12 successes, and failures = 6
4e−07
2e−07
0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta


Posterior after: 13 successes, and failures = 6
3e−07
2e−07
1e−07
0e+00
llh
0.00
0.25
0.50
0.75
1.00
theta



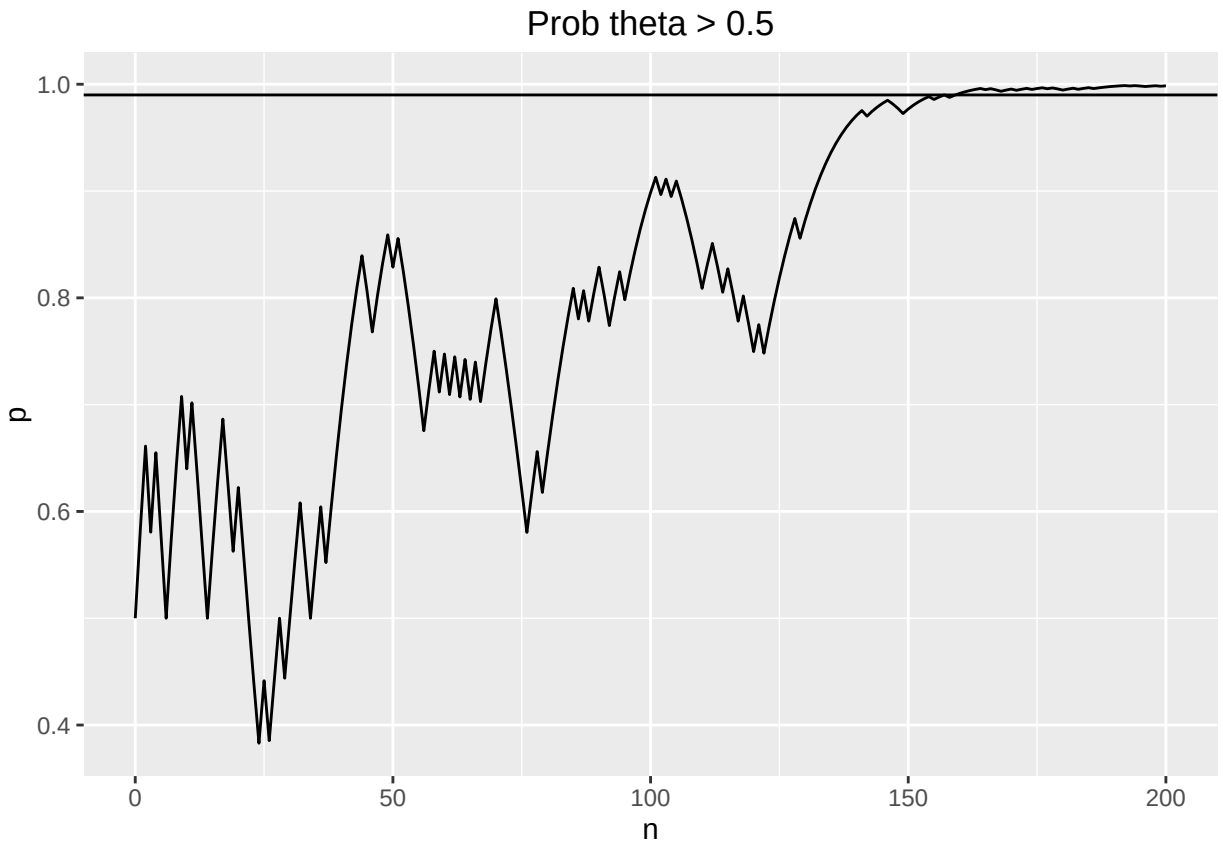

```
df.ml = data.frame(llh = beta_llh(thetas,successes + 1,failures + 1),theta=thetas)
 ggplot(df.ml) + geom_line(aes(x=theta,y=llh)) + ggtitle("ML Estimator")
```



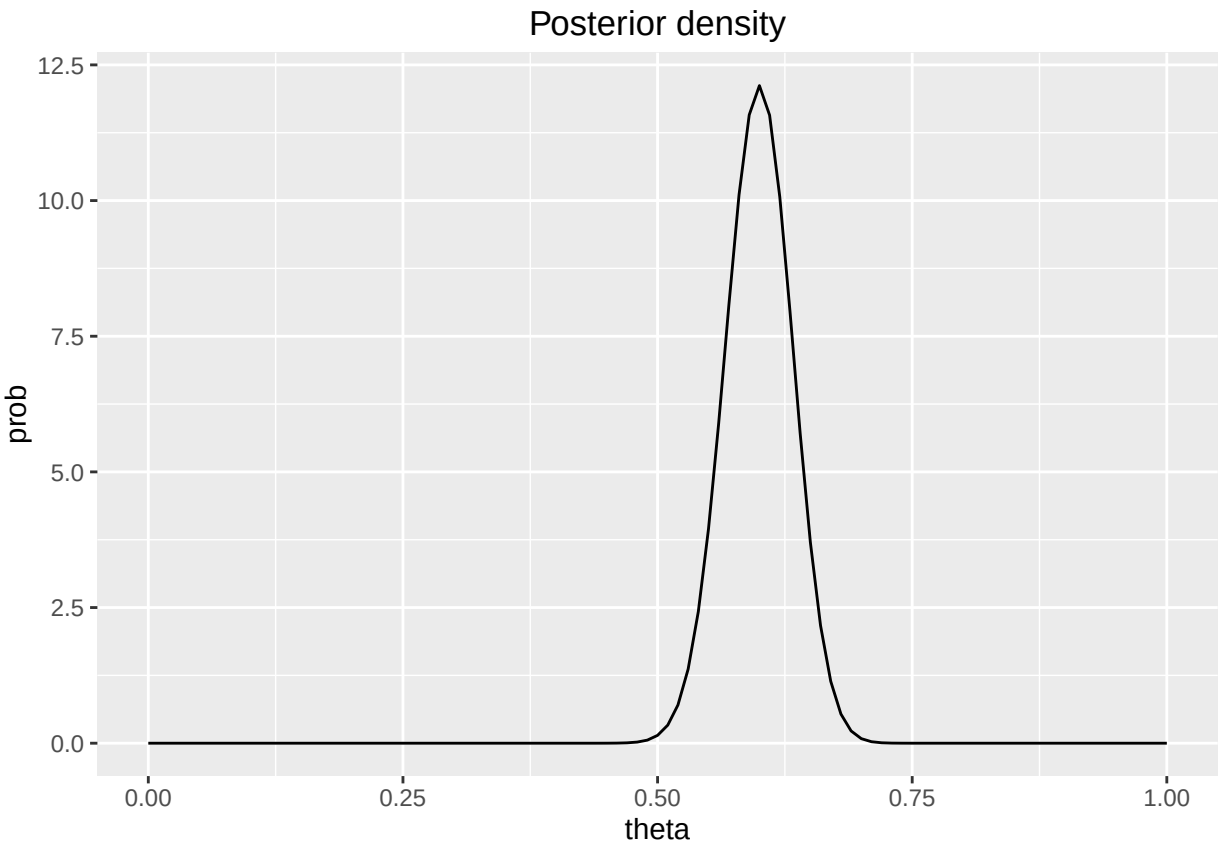
Biased Coin Example

In this example, we will start with the assumption that the coin is fair and keep flipping the coin until we can declare it to be biased. We can do this by integrating the posterior distribution and computing $p(\theta > 0.5|y) > 0.99$. We can use either numerical integration or simulation.

```
p = 0.6
alpha = 11
beta = 11
n = 200

data = rbinom(n,prob = p,size = 1)
s = 0
f = 0
ps_g_0_5 = c( pbeta(0.5,alpha,beta,lower.tail=F))

for( d in data) {
  if(d == 1){
    alpha = alpha + 1
  } else {
    beta = beta + 1
  }
  ps_g_0_5 = c(ps_g_0_5, pbeta(0.5,alpha,beta,lower.tail=F))
}

ggplot(data.frame(p=ps_g_0_5,n=0:n)) + geom_line(aes(x=n,y=p)) + geom_abline(slope = 0, intercept = 0.9
```




```
theta = seq(0,1,0.01)
y = dbeta(theta,alpha,beta)
ggplot(data.frame(theta=theta,prob=y))+ geom_line(aes(x=theta,y=prob)) + ggtitle("Posterior density")
```


Posterior density
12.5
10.0
7.5
5.0
2.5
0.0
prob
0.00
0.25
0.50
0.75
1.00
theta