Maloney_HW1

Emily Maloney

January 22, 2019

Homework - Chapters 2 and 3

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------------------------- tidyverse 1.2.

## v ggplot2 3.1.0     v purrr   0.2.5
## v tibble  2.0.1     v dplyr   0.7.8
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.3.1     v forcats 0.3.0

## -- Conflicts ------------------------------------------------------------------ tidyverse_conflicts(
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(rethinking)
```

```
## Loading required package: rstan

## Loading required package: StanHeaders

## rstan (Version 2.18.2, GitRev: 2e1f913d3ca3)

## For execution on a local, multicore CPU with excess RAM we recommend calling
## options(mc.cores = parallel::detectCores()).
## To avoid recompilation of unchanged Stan programs, we recommend calling
## rstan_options(auto_write = TRUE)

## For improved execution time, we recommend calling
## Sys.setenv(LOCAL_CPPFLAGS = '-march=native')
## although this causes Stan to throw an error on a few processors.

##
## Attaching package: 'rstan'

## The following object is masked from 'package:tidyr':
##
##     extract

## Loading required package: parallel

## rethinking (Version 1.59)

##
## Attaching package: 'rethinking'

## The following object is masked from 'package:purrr':
##
##     map
```

```
library(tidybayes)
```

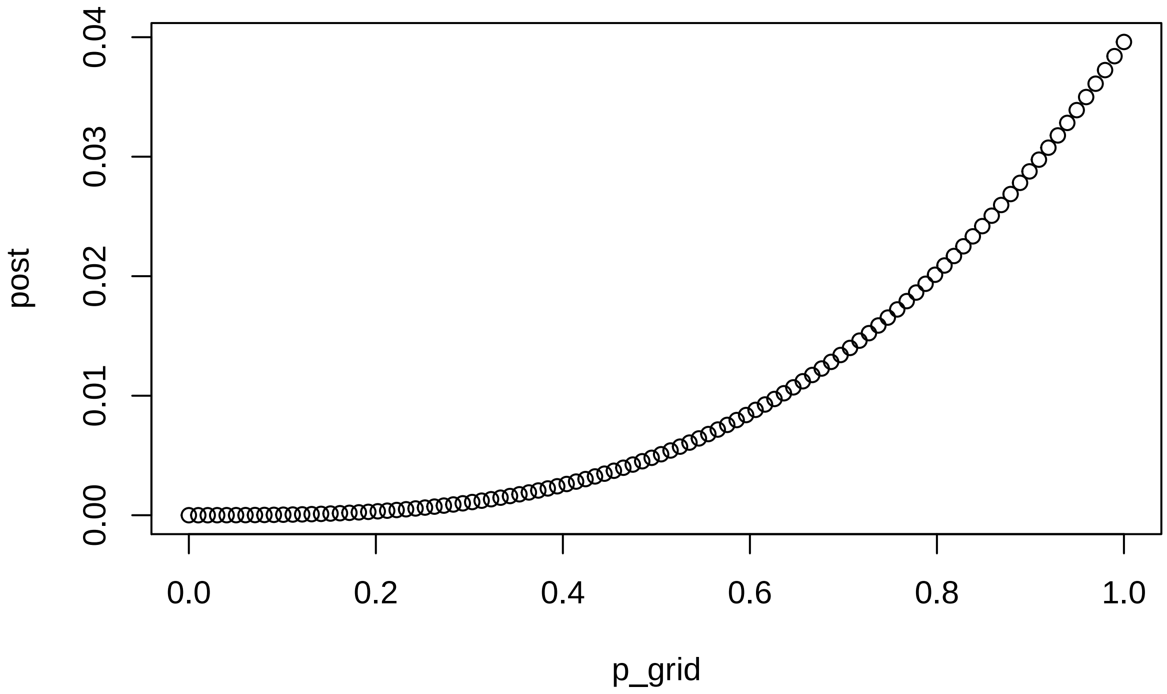
```
## NOTE: As of tidybayes version 1.0, several functions, arguments, and output column names
##       have undergone significant name changes in order to adopt a unified naming scheme.
##       See help('tidybayes-deprecated') for more information.
```

Chapter 2 Homework

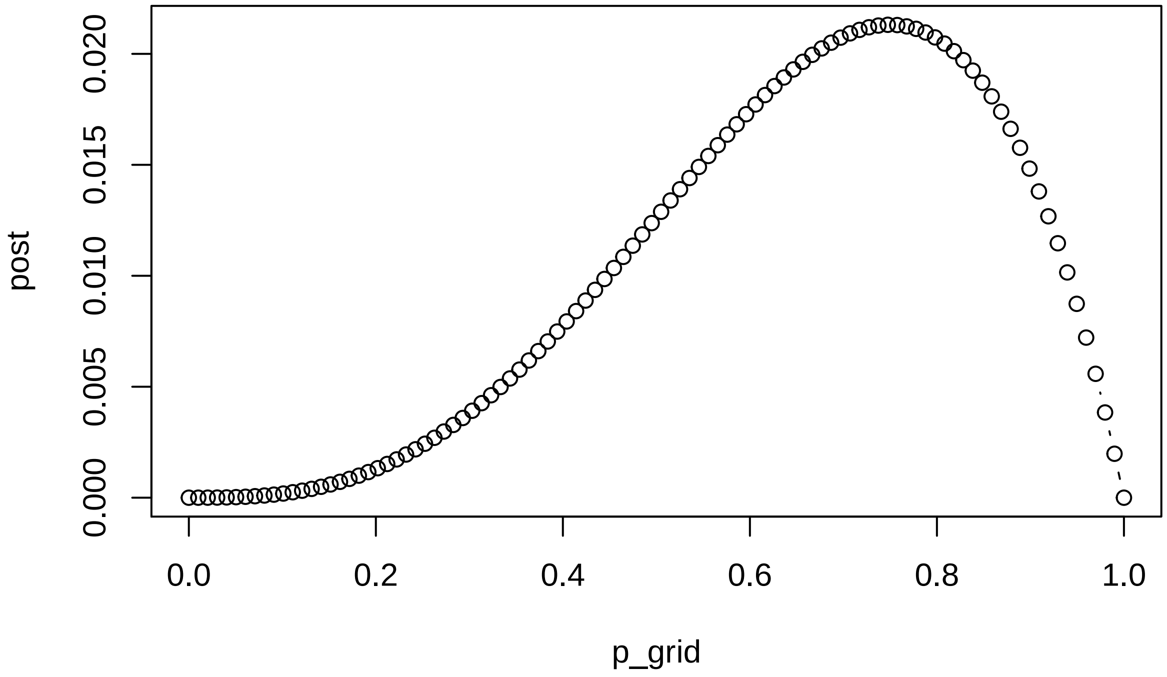
Medium Problems

2M1

```
#WWW
              #define grid
              p_grid <- seq(from = 0, to = 1, length.out = 100)

              #define prior
              prior <- rep(1, 100)
              #prior <- exp(-5*abs(p_grid - 0.5))

              #compute likelihood at each value in the grid
              lh <- dbinom(3, size = 3, prob = p_grid)

              #compute product of lh & prior
              upost <- lh * prior

              #standardize post
              post <- upost/sum(upost)

              plot(p_grid, post, type = "b")
```

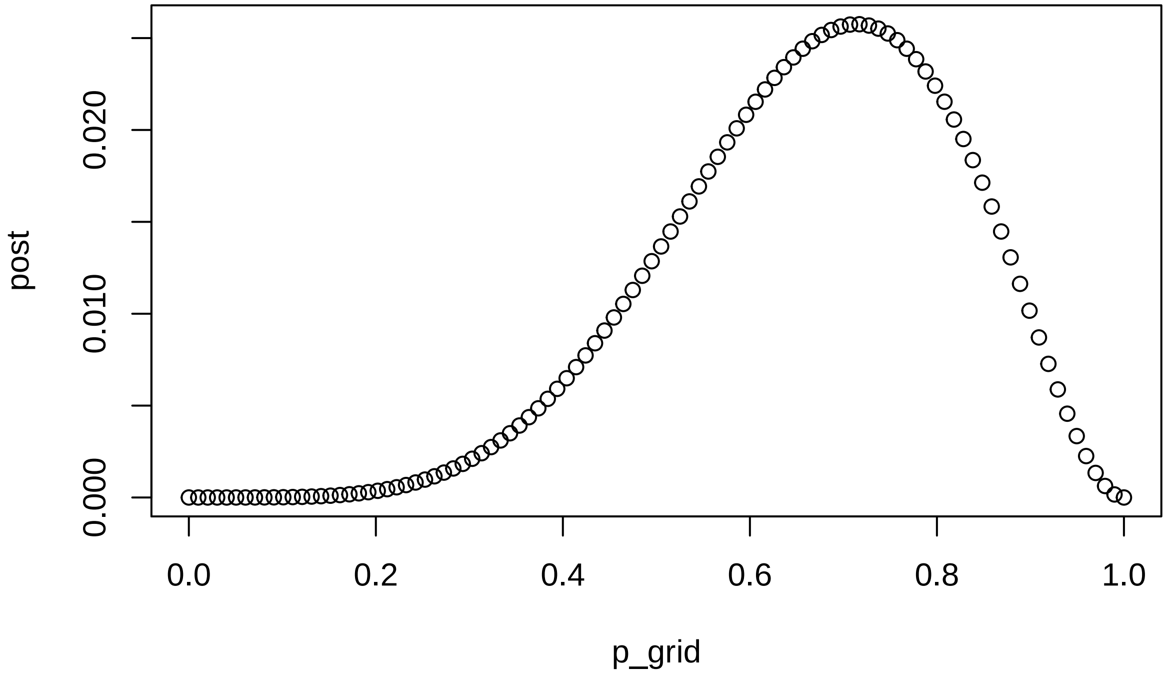


```
#WWWL
          #define grid
          p_grid <- seq(from = 0, to = 1, length.out = 100)

          #define prior
          prior <- rep(1, 100)
          #prior <- exp(-5*abs(p_grid - 0.5))

          #compute likelihood at each value in the grid
          lh <- dbinom(3, size = 4, prob = p_grid)

          #compute product of lh & prior
          upost <- lh * prior

          #standardize post
          post <- upost/sum(upost)

          plot(p_grid, post, type = "b")
```



```
#LWWLWWW
          #define grid
          p_grid <- seq(from = 0, to = 1, length.out = 100)

          #define prior
          prior <- rep(1, 100)
          #prior <- exp(-5*abs(p_grid - 0.5))
```

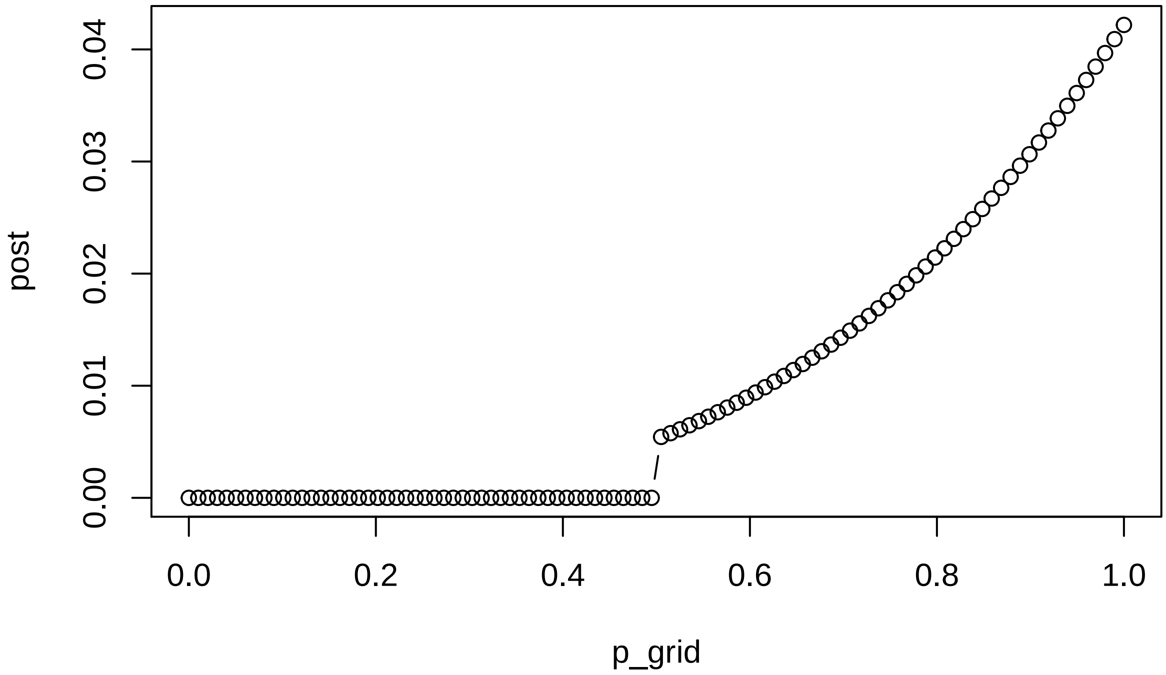
```
              #compute likelihood at each value in the grid
              lh <- dbinom(5, size = 7, prob = p_grid)

              #compute product of lh & prior
              upost <- lh * prior

              #standardize post
              post <- upost/sum(upost)

              plot(p_grid, post, type = "b")
```



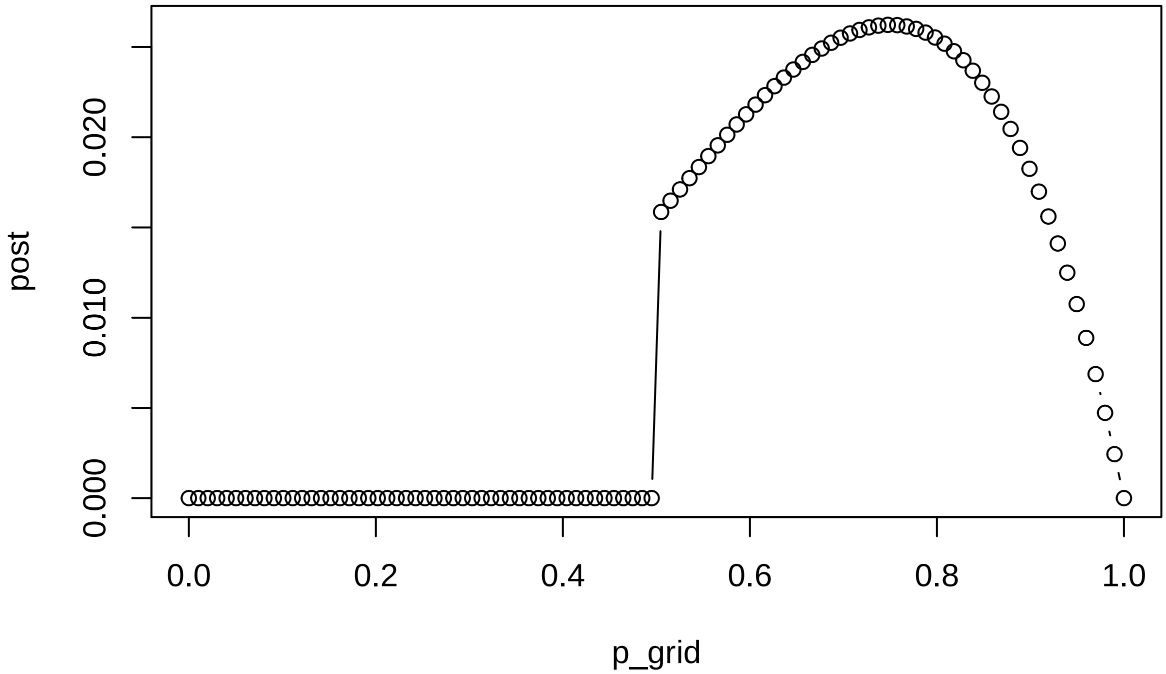
2M2

```
#WWW
              #define grid
              p_grid <- seq(from = 0, to = 1, length.out = 100)

              #define prior
              prior <- ifelse(p_grid < 0.5, 0, 2)

              #compute likelihood at each value in the grid
              lh <- dbinom(3, size = 3, prob = p_grid)

              #compute product of lh & prior
              upost <- lh * prior
```

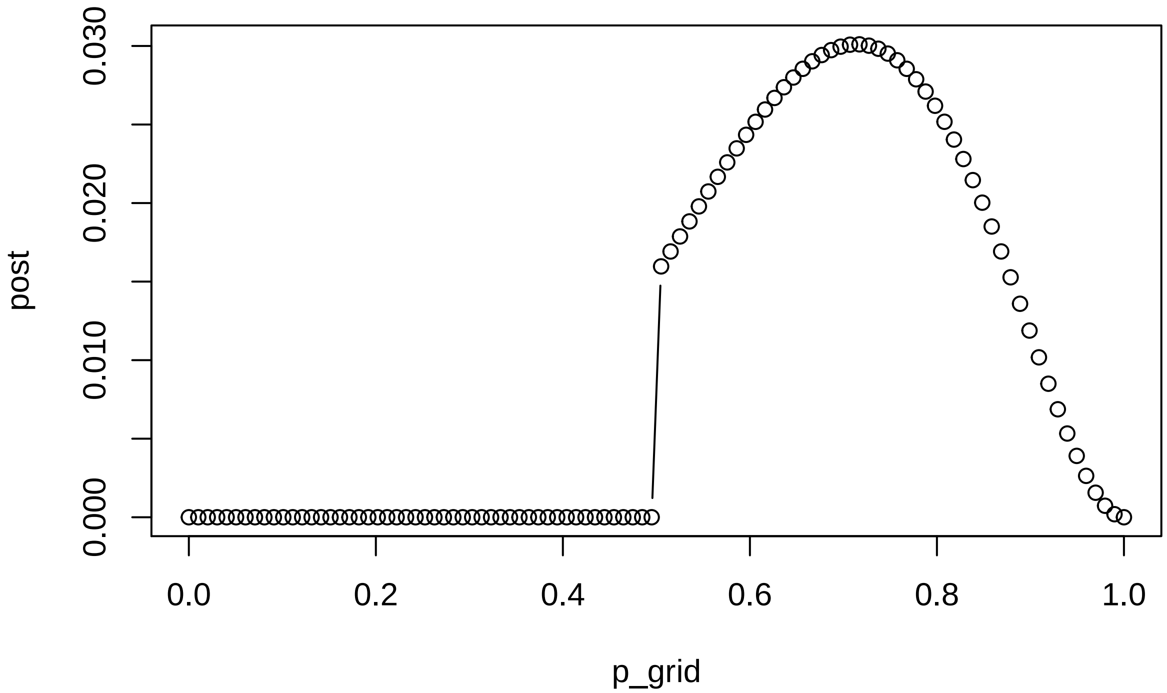
```
#standardize post
post <- upost/sum(upost)

plot(p_grid, post, type = "b")
```



```
#WWWL
#define grid
p_grid <- seq(from = 0, to = 1, length.out = 100)

#define prior
prior <- ifelse(p_grid < 0.5, 0, 2)

#compute likelihood at each value in the grid
lh <- dbinom(3, size = 4, prob = p_grid)

#compute product of lh & prior
upost <- lh * prior

#standardize post
post <- upost/sum(upost)

plot(p_grid, post, type = "b")
```



```r
#LWWLWWW
#define grid
p_grid <- seq(from = 0, to = 1, length.out = 100)

#define prior
prior <- ifelse(p_grid < 0.5, 0, 2)
#prior <- exp(-5*abs(p_grid - 0.5))

#compute likelihood at each value in the grid
lh <- dbinom(5, size = 7, prob = p_grid)

#compute product of lh & prior
upost <- lh * prior

#standardize post
post <- upost/sum(upost)

plot(p_grid, post, type = "b")
```



2M3

```
pwe <- 0.7
plm <- 1
lh <- 0.3/1
priorodds <- 1
postodds <- priorodds*lh
(post <- postodds/(postodds + 1))
```

```
## [1] 0.2307692
```

The probability of the globe was Earth, given seeing land is 0.2307692.

2M4

```
ww <- 0
bw <- 1
bb <- 2

(p <- bb/(bb + bw + ww))
```

```
## [1] 0.6666667
```

Given the fact that a card with two white sides cannot produce a black side facing up, a black and white card can produce a black side facing up 1 way, and a card with two black sides can produce a black side facing up two ways, the probablity that the other side of a card with a black face up is 2/3.

2M5

```r
ww <- 0
bw <- 1
bb <- 2*2

(p <- bb/(bb + bw + ww))
```

```
## [1] 0.8
```

If there are two cards that have black on both sides, the probability that a card with a black side facing up also has black on the other side is now 4/5.

2M6

```r
ww <- 0*3
bw <- 1*2
bb <- 2

(p <- bb/(bb + bw + ww))
```

```
## [1] 0.5
```

If there are two ways to pull out a black and white card and 3 ways to pull out a white and white card for every way to pull out a black and black card, the probability that the other side is black is now 1/2.

2M7

```r
ww <- 0 * (1 + 0)
bb <- 2 * (2 + 1)
bw <- 1 * (0 + 2)

(p <- bb/(bb + bw + ww))
```

```
## [1] 0.75
```

If a second card is drawn with a white side face up, the probability that the first card with the black side facing up has black on the other side, is now 0.75.

Chapter 3 Homework

Easy Problems

```r
p_grid <- seq(from = 0, to = 1, length.out = 1000)
prior <- rep(1, 1000)
lh <- dbinom(6, size = 9, prob = p_grid)
post <- lh * prior
post <- post/sum(post)
set.seed(100)
samples <- sample(p_grid, prob = post, size = 1e4, replace = T)
samples <- as_tibble(samples)
```

```
## Warning: Calling `as_tibble()` on a vector is discouraged, because the behavior is likely to change
## This warning is displayed once per session.
```

3E1

```
less20 <- samples %>% filter(value < 0.2) %>%
            summarise(sum = n()/1e4)
```

There is 0.005 posterior probability that lies below p = 0.2.

3E2

```
more80 <- samples %>% filter(value > 0.8) %>%
            summarise(sum = n()/1e4)
```

There is 0.0112 posterior probability that lies above p = 0.8.

3E3

```
bw2080 <- samples %>% filter(value > 0.2 & samples < 0.8) %>%
            summarise(sum = n()/1e4)
```

There is 0.0888 posterior probability that lies between p = 0.2 and p = 0.8.

3E4

```
quantile(samples$value, p = 0.2)
```

```
##       20%
## 0.5195195
```

20% of the posterior probability lies below p = 0.52.

3E5

```
quantile(samples$value, p = 0.8)
```

```
##       80%
## 0.7567568
```

20% of the posterior probability lies above p = 0.757.

3E6

```
HPDI(samples$value, p = 0.66)
```

```
##     |0.66     0.66|
## 0.5205205 0.7847848
```

The values of p which contain the narrowest interval equal to 66% of the posterior probability are 0.521 and 0.785.

3E7

```
quantile(samples$value, p = 0.83)
```

```
##       83%
## 0.7687688
```

```r
quantile(samples$value, p = 0.17)
```

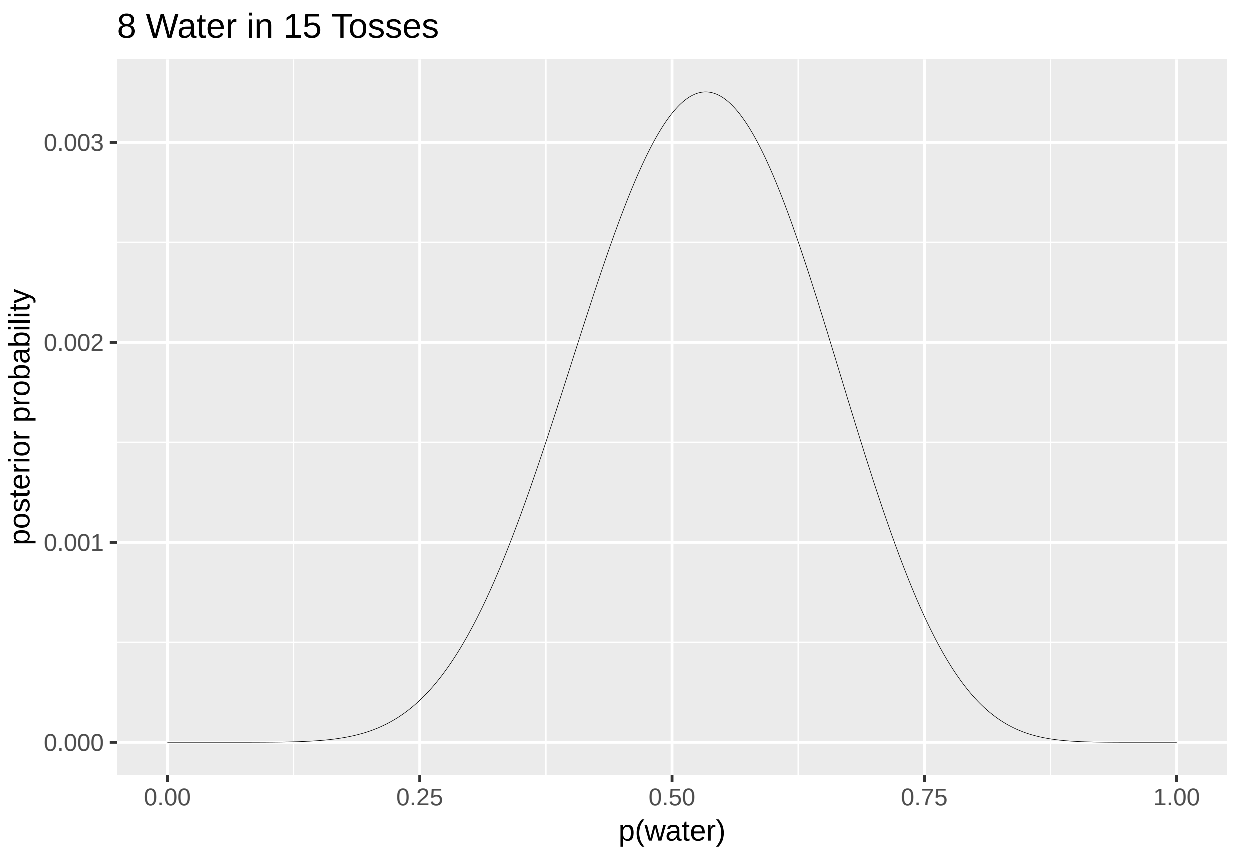
```
##       17%
## 0.5005005
```

The values of p which contain 66% of the posterior probability, assuming equal posterior probability both below and above the interval are 0.500 and 0.769.

Medium Problems

3M1

```r
n <- 1000
n_success <- 8
n_trials <- 15

d <- tibble(p_grid = seq(from = 0, to = 1, length.out = n),
            prior = 1) %>%
     mutate(lh = dbinom(n_success, size = n_trials, prob = p_grid),
            post = lh * prior,
            post = post/sum(post))

d %>%
            ggplot(aes(x = p_grid, y = post)) +
            geom_line(size = 1/10) +
            labs(x = "p(water)",
                 y = "posterior probability") + ggtitle("8 Water in 15 Tosses")
```



3M2

```
#drawing samples
samples <- tibble(samples = sample(d$p_grid, prob = d$post, size = 10000, replace = T)) %>%
           mutate(sample_n = 1:n())
head(samples)
```

```
## # A tibble: 6 x 2
##   samples sample_n
##     <dbl>    <int>
## 1   0.666        1
## 2   0.703        2
## 3   0.376        3
## 4   0.381        4
## 5   0.456        5
## 6   0.521        6
```

```
#90% HPDI
HPDI(samples$samples, p = 0.9)
```

```
##      |0.9      0.9|
## 0.3383383 0.7317317
```

The 90% HPDI for p is 0.338-0.731.

3M3

```r
ppc <- tibble(sample = rbinom(1e4, size = 15, prob = samples$samples))
p8 <- ppc %>% filter(sample == 8) %>%
              summarise(sum = n()/1e4)
```

There is a p8 probability of getting 8 tosses out of 15.

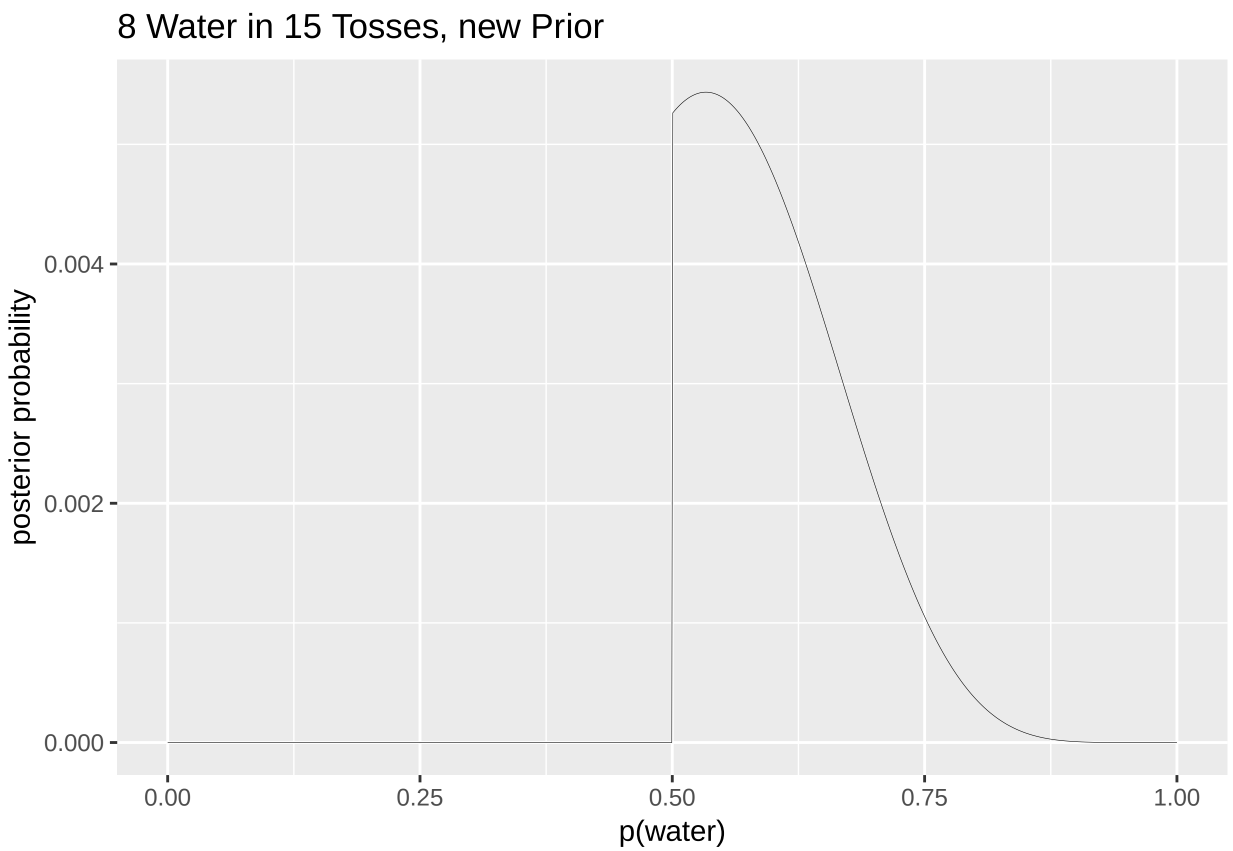
3M4

```r
newsim <- tibble(sample = rbinom(1e4, size = 9, prob = samples$samples))
p69 <- newsim %>% filter(sample == 6) %>%
                  summarise(sum = n()/1e4)
```

There is a p69 probability of getting 6 tosses out of 9.

3M5

```r
n <- 1000
n_success <- 8
n_trials <- 15

d <- tibble(p_grid = seq(from = 0, to = 1, length.out = n),
            prior = ifelse(p_grid < 0.5, 0, 2)) %>%
     mutate(lh = dbinom(n_success, size = n_trials, prob = p_grid),
            post = lh * prior,
            post = post/sum(post))

d %>%
            ggplot(aes(x = p_grid, y = post)) +
            geom_line(size = 1/10) +
            labs(x = "p(water)",
                 y = "posterior probability") + ggtitle("8 Water in 15 Tosses, new Prior")
```


8 Water in 15 Tosses, new Prior

```
#drawing samples
samples <- tibble(samples = sample(d$p_grid, prob = d$post, size = 10000, replace = T)) %>%
           mutate(sample_n = 1:n())

#90% HPDI
HPDI(samples$samples, p = 0.9)
```

```
##      |0.9      0.9|
## 0.5005005 0.7097097
```

```
#posterior predictive check
ppc <- tibble(sample = rbinom(1e4, size = 15, prob = samples$samples))
p8 <- ppc %>% filter(sample == 8) %>%
              summarise(sum = n()/1e4)
print(p8)
```

```
## # A tibble: 1 x 1
##     sum
##   <dbl>
## 1 0.159
```

```
#probability of observing 6 water in 9 tosses
newsim <- tibble(sample = rbinom(1e4, size = 9, prob = samples$samples))
p69 <- newsim %>% filter(sample == 6) %>%
                  summarise(sum = n()/1e4)
print(p69)
```

```
## # A tibble: 1 x 1
##     sum
##   <dbl>
## 1 0.236
```