Chapter-03-assignment.R

Ruijuan

February 23, 2016

just for practice

install packages

```
library(rmarkdown)
library(rstan)
```

```
## Loading required package: ggplot2
```

```
## rstan (Version 2.9.0-3, packaged: 2016-02-11 15:54:41 UTC, GitRev: 05c3d0058b6a)
```

```
## For execution on a local, multicore CPU with excess RAM we recommend calling
## rstan_options(auto_write = TRUE)
## options(mc.cores = parallel::detectCores())
```

```
library(rethinking)
```

```
## Loading required package: parallel
```

```
## rethinking (Version 1.58)
```

R code 3.27

```
p_grid <- seq(0, 1, length.out = 1000)
prior <- rep(1, 1000)
likelihood <- dbinom(6, size = 9, prob = p_grid)
posterior <- likelihood * prior
posterior <- posterior/sum(posterior)
set.seed(100)
samples <- sample(p_grid, prob = posterior, size = 1e4, replace = TRUE)
```

use the values in samples to answer the questions

3E1-3E3 the intervals of defined boundary

3E1. How much posterior probability lies below p=0.2

```
sum(posterior[p_grid < 0.2])
```

```
## [1] 0.0008560951
```

```
sum(samples<0.2)/1e4
```

```
## [1] 5e-04
```

3E2. How much posterior proabbility lies above p=0.8

```
sum(posterior[p_grid > 0.8])
```

```
## [1] 0.1203449
```

```
sum(samples>0.8)/1e4
```

```
## [1] 0.1117
```

3E3. How much posterior probability lies between p=0.2 and p=0.8

```
sum(posterior[p_grid > 0.2 & p_grid<0.8])
```

```
## [1] 0.878799
```

```
sum(samples>0.2 & samples<0.8)/1e4
```

```
## [1] 0.8878
```

The intervals of defined mass

3E4. 20% of the posterior probability lies below which value of p?

```
quantile(samples, 0.2)
```

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

3E5. 20% of the posterior probability lies above which value of p?

```
quantile(samples, 0.8)
```

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

3E6. Which values of p contain the narrowest interval equal to 66% of the posterior probability?

```
HPDI(samples, prob = 0.66)
```

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

3E7. Which values of p contain 66% of the posterior probability, assuming equal posterior proability both below and above the interval?

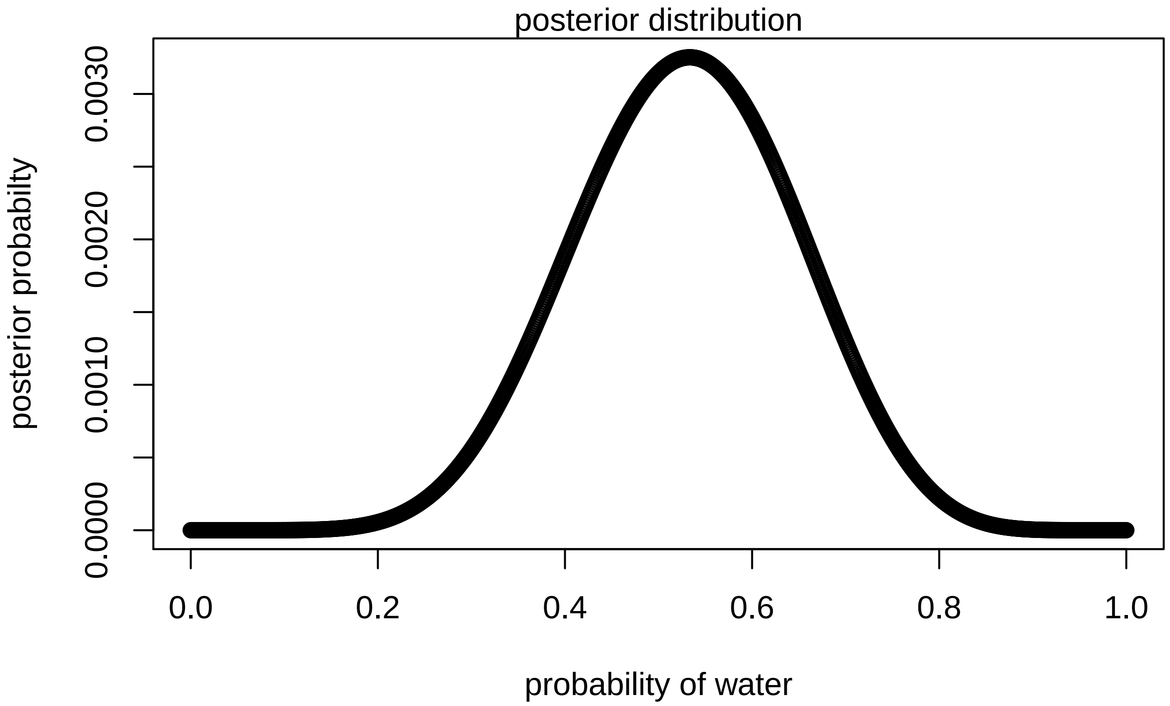
```
PI(samples, prob = 0.66)
```

```
##       17%       83%
## 0.5005005 0.7687688
```

3M1. Suppose the globe tossing data had turned out to be 8 water in 15 tosses.

Construct the posterior distribution, using grid approximation. Use the same flat prior as before

```
p_grid <- seq(0, 1, length.out = 1000)
prior <- rep(1, 1000)
likelihood <- dbinom(8, size = 15, prob = p_grid)
posterior <- likelihood * prior
posterior <- posterior/sum(posterior)
```



3M2. Draw 10,000 samples from the grid approximation from above.

Then use the samples to calculate the 90% HPDI() for p.

```
set.seed(8808)
samples <- sample(p_grid, prob = posterior, size = 1e4, replace = TRUE)
# the 90% highest posterior density (narrowest) interval
HPDI(samples, prob = 0.9)
```

```
##      |0.9      0.9|
## 0.3413413 0.7317317
```

STOP AFTER 3M2 FOR 02/25 ASSIGNMENT_

3M3. construct a posterior predictive check for this model and data. This means simulate the distribution of samples, averaging over the posterior uncertaninty in p (all possible p).

What is the probability of observing 8 water in 15 tosses?

```
# 1e5 random results from size of 15 (15 tosses, based on the prob of samples generated above)
dummy_w_eight_fifteen <- rbinom(1e5, size = 15, prob = samples)
sum(dummy_w_eight_fifteen==8)/length(dummy_w_eight_fifteen)
```

```
## [1] 0.14654
```

3M4. Using the posterior distribution constructed from the new (8/15) data, now calcualte the probability of observing 6 water in 9 tosses.

```
dummy_w_six_nine <- rbinom(1e5, size = 9, prob = samples)
sum(dummy_w_six_nine==6)/length(dummy_w_six_nine)
```

```
## [1] 0.17677
```

3M5. Start over at 3M1, but now use a prior that is zero below p=0.5 and a constant above p=0.5, This correspondes to prior information that a majority of the Earth's surface is water. Repeat each problem above and compare the inferences. What difference does the better prior make? If it helps, compare inferences (using both priors) to the true value p=0.7

```
# do p_grdi approximation and draw samples
p_grid <- seq(0, 1, length.out = 1000)
prior <- ifelse(p_grid<0.5, 0, 1)
likelihood <- dbinom(8, size = 15, prob = p_grid)
posterior <- likelihood * prior
posterior <- posterior/sum(posterior)
set.seed(100)
samples2 <- sample(p_grid, prob = posterior, size = 1e4, replace = TRUE)
HPDI(samples2, prob = 0.9)
```

```
##      |0.9      0.9|
## 0.5005005 0.7077077
```

```
# sampling to simulate prediction 8/15
dummy_w_eight_fifteen_new <- rbinom(1e5, size = 15, prob = samples2)
sum(dummy_w_eight_fifteen_new==8)/length(dummy_w_eight_fifteen_new)
```

```
## [1] 0.15802
```

```
# 6/9, with this original 8/15 posterior
dummy_w_six_nine_new <- rbinom(1e5, size = 9, prob = samples2)
sum(dummy_w_six_nine_new==6)/length(dummy_w_six_nine_new)
```

```
## [1] 0.23236
```

```
# with true value p=0.7 for simulation
dummy_w_eight_fifteen_new_new <- dbinom(1e5, size = 15, prob = 0.7)
sum(dummy_w_eight_fifteen_new_new==8)/length(dummy_w_eight_fifteen_new_new)
```

```
## [1] 0
```

```
dummy_w_six_nine_new_new <- rbinom(1e5, size = 9, prob = 0.7)
sum(dummy_w_six_nine_new_new==6)/length(dummy_w_six_nine_new_new)
```

```
## [1] 0.2679
```

```
# result from six_nine with new prior is closer to the true value "0.7" result, although
# eight_fifteen with flat prior is closer to the true value "0.7" result. Why?
```

3H1

R code 3.29, load data into R

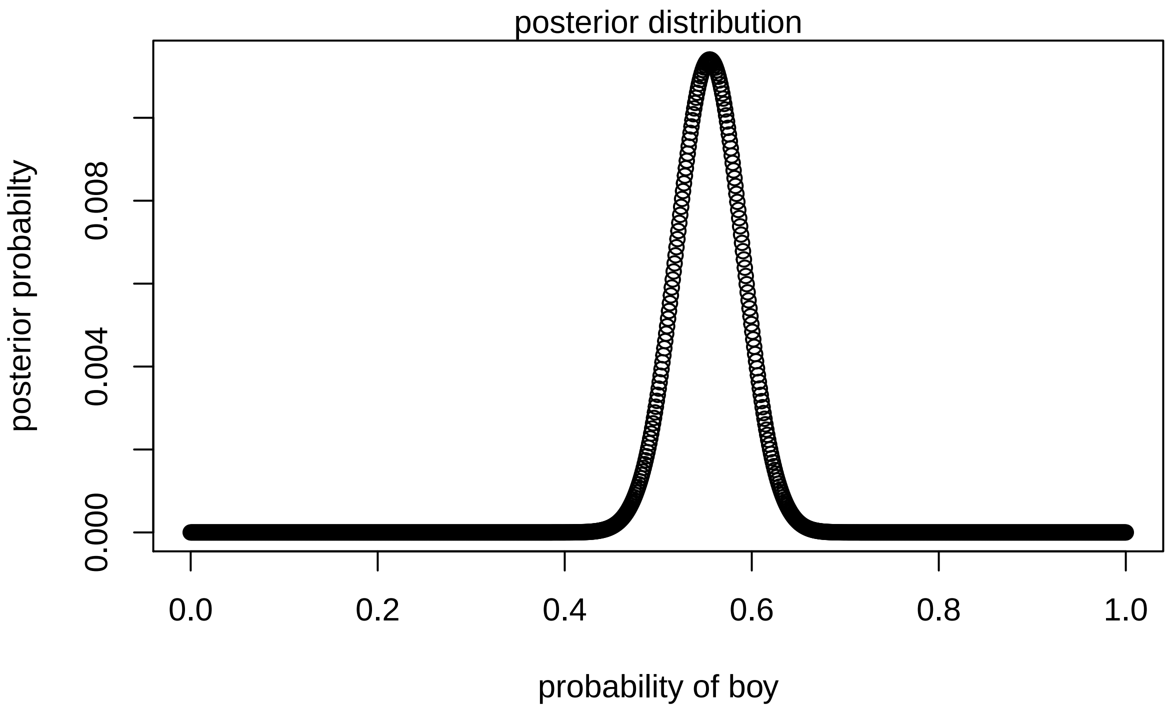
```
library(rethinking)
data(homeworkch3)
```

R code 3.30, to compute the total number of boys born across all of these births

```
sum(birth1) + sum(birth2)
```

```
## [1] 111
```

```
# do p_grid approximation, to compute the posterior distribution of a birth being a boy, with
# uniform prior.
p_grid <- seq(0, 1, length.out = 1000)
prior <- rep(1, 1000)
likelihood <- dbinom(111, size = 200, prob = p_grid)
posterior <- likelihood * prior
posterior <- posterior/sum(posterior)
```



```
## [1] 0.5545546
```

3H2.

```
# randomly draw 1e4 samples and caculate 50%, 89%, and 97% highest posterior density intervals
set.seed(100)
samples <- sample(p_grid, prob = posterior, size = 1e4, replace = TRUE)
HPDI(samples, prob = 0.5)
```

```
##      |0.5      0.5|
## 0.5315315 0.5765766
```

```
HPDI(samples, prob = 0.89)
```

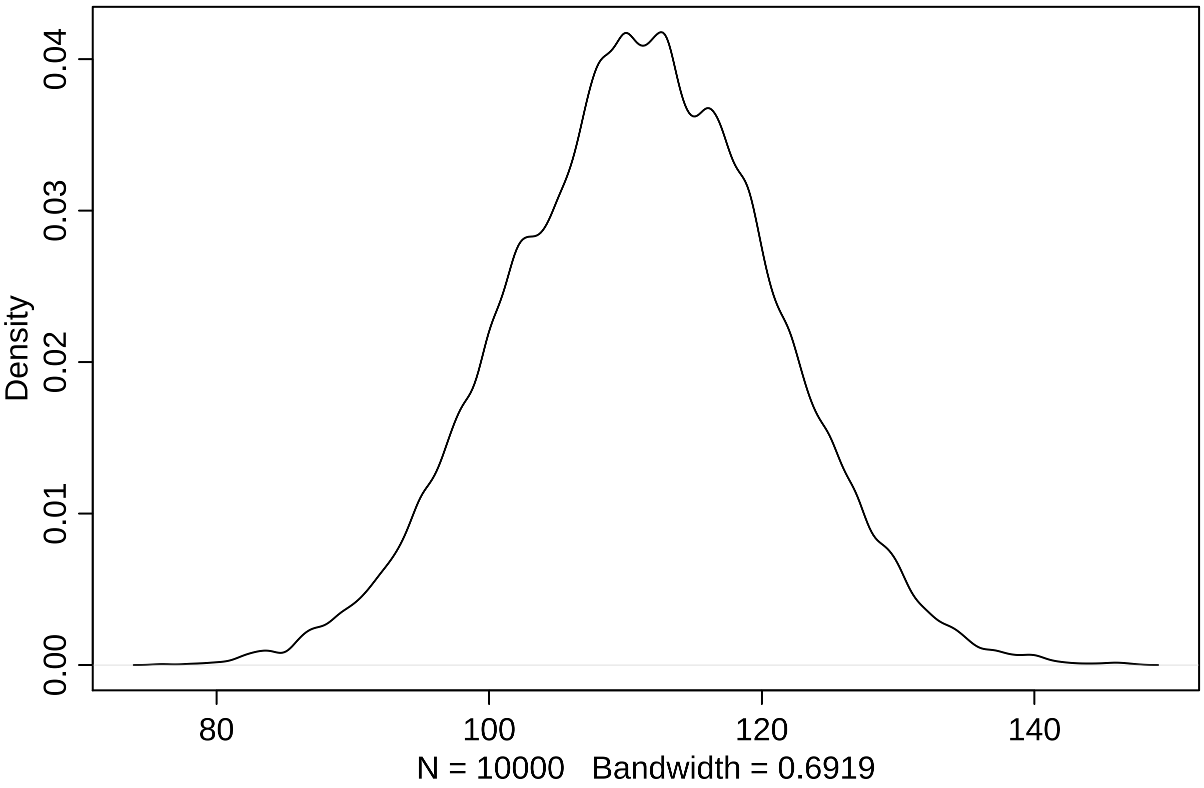
```
##     |0.89     0.89|
## 0.4974975 0.6076076
```

```
HPDI(samples, prob = 0.97)
```

```
##     |0.97     0.97|
## 0.4774775 0.6276276
```

3H3.

```
# make simulated predctions
set.seed(100)
dummy_boy <- rbinom(1e4, size = 200, prob = samples)
# check the central of the distribution
dens(dummy_boy)
```



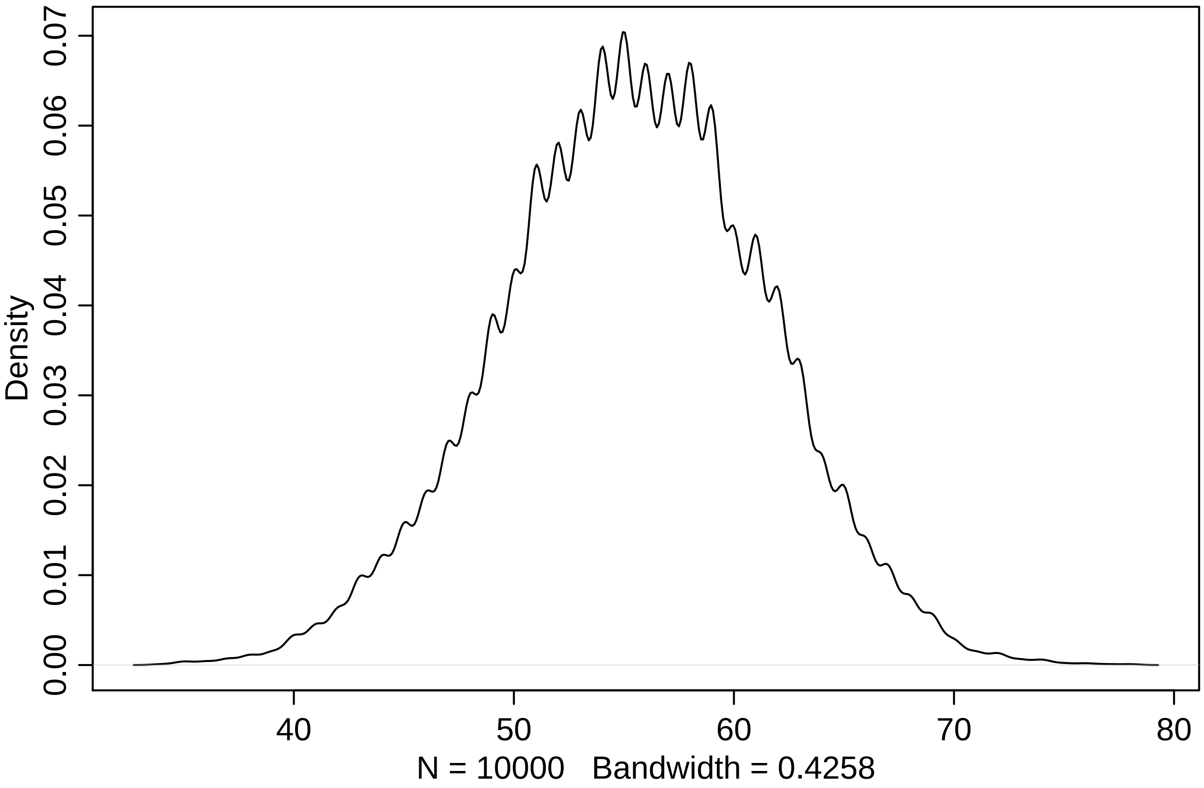
```r
HPDI(dummy_boy, prob = 0.95)
```

```
## |0.95 0.95|
##    93   130
```

```r
# 111 is in the center, the model looks good
```

3H4. Now compare 10,000 counts of boys from 100 simulated first borns only to the number of boys in the first births, birth1. How does the model look in this light? (don't understand this Q)

```r
# simulate 10,000 counts of boys from 100 simulated first borns
set.seed(100)
dummy_boy2 <- rbinom(1e4, size = 100, prob = samples)
dens(dummy_boy2)
```



```
HPDI(dummy_boy2, prob = 0.95)
```

```
## |0.95 0.95|
##    43    66
```

```
sum(birth1)
```

```
## [1] 51
```

```
# still looks good, the number is in the center, 95% probability, a little worse
```

3H5. The model assumes that sex of first and second births are independent. To check this assumption, focus now on second births that followed female first borns. Compare 10,000 simulated counts of boys to only those second births that followed girls. To do this correctly, you need to count the number of first borns who were girls and simulate that many births, 10,000 times. Compare the counts of boys in your simulations to the actual observed count of boys following girls. How does the model look in this light? Any guesses what is going on in these data?

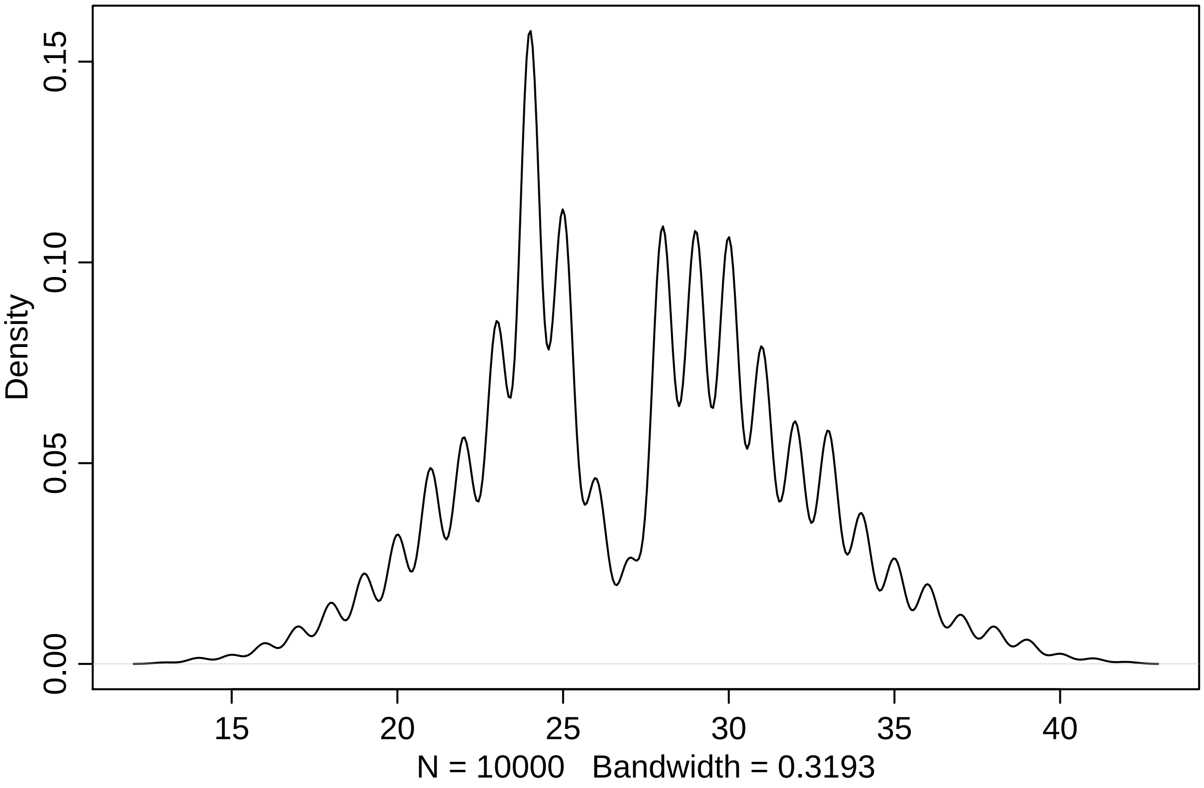
```
# count the number of second birth boy after first birth girls
sum(birth1==0) # first birth girls
```

```
## [1] 49
```

```
sum(birth1==0 & birth2==1)
```

```
## [1] 39
```

```
# simulate 10,000 times using the paramters generated from H1
set.seed(100)
dummy_boy3 <- rbinom(1e4, size = 49, prob = samples)
dens(dummy_boy3)
```



```
# compare to the observation
HPDI(dummy_boy3, prob = 0.95)
```

```
## |0.95 0.95|
##    17    35
```

```
# the model doesn't look right, not indepent of first birth and second birth gender.
```