Chapter 3 Notes and Exercises

Matthias Grenié

19 juillet 2016

Chapter 3

Notes

How to sample from a posterior distribution?

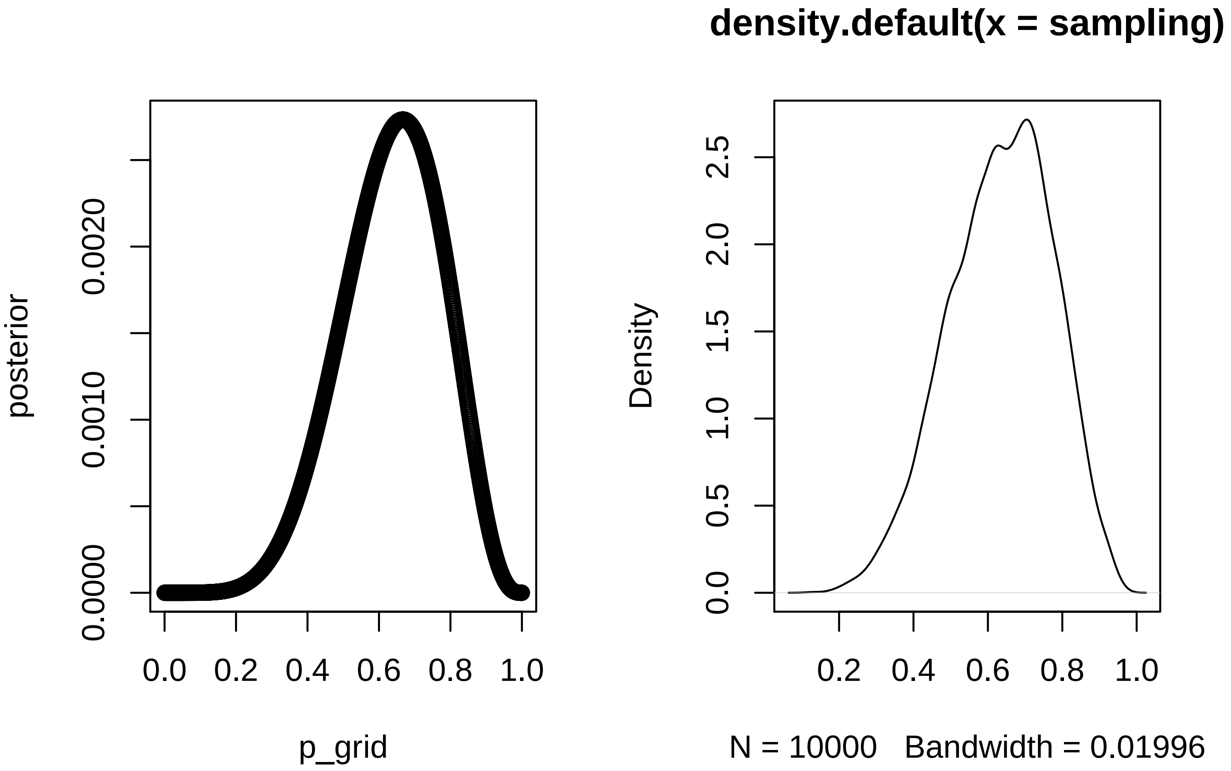
```r
p_grid = seq(from=0, to=1, length.out=1000)
prior = rep(1 , 1000)
likelihood = dbinom(6 , size=9, prob=p_grid)
posterior = likelihood * prior
posterior = posterior / sum(posterior)

sampling = sample(p_grid, prob = posterior, size = 1e4, replace = TRUE)
```

We sample according to the value of the posterior distributions, thus it gives back the posterior distribution:

```r
par(mfrow = c(1, 2))

plot(p_grid, posterior)
plot(density(sampling))
```


density.default(x = sampling)

Practice

Easy

```r
p_grid <- seq(from=0 , to=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)
```

3E1

```r
sum(samples < 0.2)/length(samples)
```

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

3E2

```r
sum(samples > 0.8)/length(samples)
```

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

3E3

```r
sum(samples > 0.2 & samples < 0.8)/length(samples)
```

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

3E4

```r
quantile(samples, probs = c(0.2))
```

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

3E5

```r
quantile(samples, probs = c(0.8))
```

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

3E6

```
hpdi_66 = rethinking::HPDI(samples, prob = 0.66)
hpdi_66
```

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

```
# Verification, it should contain 66% of the values
sum(samples > min(hpdi_66) & samples < max(hpdi_66))/length(samples)
```

```
## [1] 0.6575
```

3E7

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

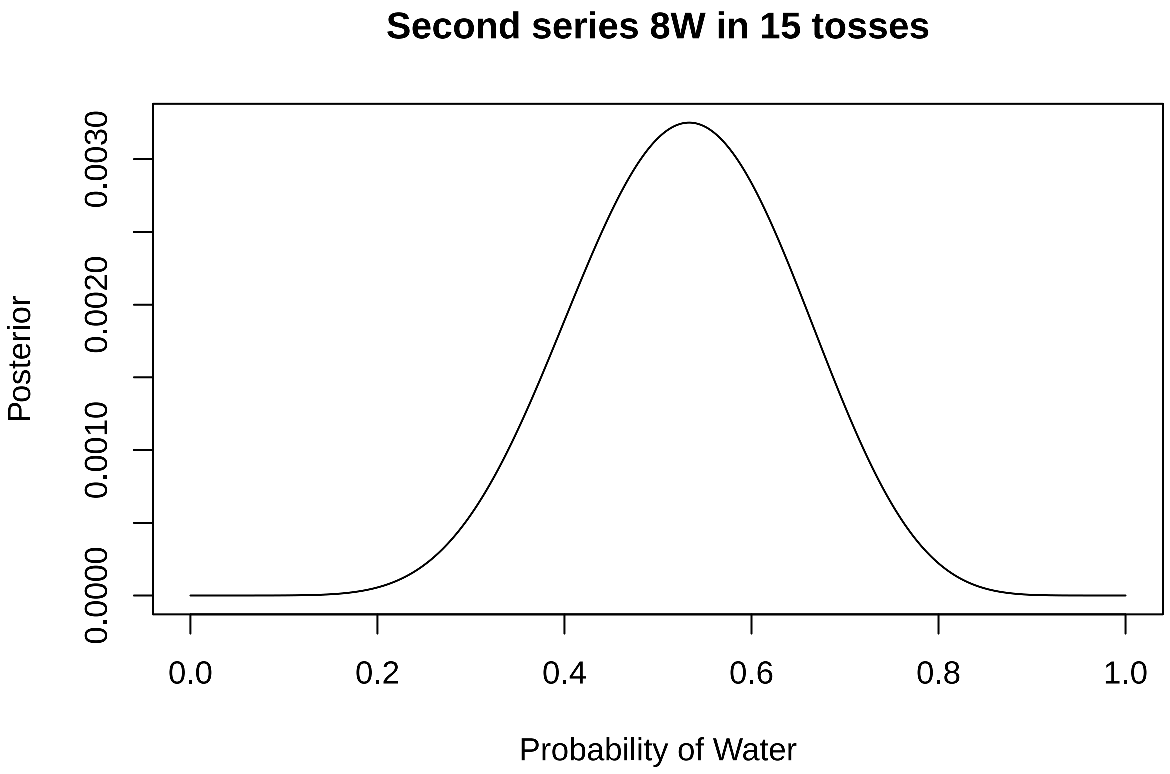
```
##       17%       83%
## 0.5005005 0.7687688
```

Medium

3M1

```
like_2 <- dbinom(8, size=15, prob=p_grid)
post_2 <- like_2 * prior
post_2 <- post_2 / sum(post_2)
set.seed(100)
samples_2 <- sample(p_grid, prob=post_2, size=1e4, replace=TRUE)
```

```
par(mfrow = c(1, 1))
plot(p_grid, post_2, type = "l", xlab = "Probability of Water",
     ylab = "Posterior", main = "Second series 8W in 15 tosses")
```



3M2

```
hdpi_2 = rethinking::HPDI(samples_2, 0.9)
```

3M3

```
prob_8_15 = rbinom(1e4, size = 15, prob = samples_2)
sum(prob_8_15 == 8)/1e4
```

```
## [1] 0.1475
```

3M4

```
prob_6_9 = rbinom(1e4, size = 9, prob = samples_2)
sum(prob_6_9 == 6)/1e4
```

```
## [1] 0.1766
```

3M5

Let's start over everything using a different prior:

```
prior_3 <- prior
prior_3[p_grid < 0.5] <- 0
like_3 <- dbinom(8, size=15, prob=p_grid)
```

```r
post_3 <- like_3 * prior_3
post_3 <- post_3 / sum(post_3)
set.seed(100)
samples_3 <- sample(p_grid, prob = post_3, size=1e4, replace = TRUE)

# Compare HDPIs
hdpi_3 = rethinking::HPDI(samples_3, prob = 0.9)
c("flat_prior" = hdpi_2, "better_prior" = hdpi_3)
```

```
##   flat_prior.|0.9   flat_prior.0.9| better_prior.|0.9 better_prior.0.9|
##         0.3243243         0.7157157         0.5005005         0.7077077
```

```r
# Compare prediction of probability of obtaining 8W in 15 tosses
prob_8_15_3 = rbinom(1e4, size = 15, prob = samples_3)
c("flat_prior" = sum(prob_8_15 == 8)/1e4,
  "better_prior" = sum(prob_8_15_3 == 8)/1e4)
```

```
##   flat_prior better_prior
##       0.1475       0.1617
```

```r
# Now using posterior compare chances of obtaining 6W in 9 tosses
prob_6_9_3 = rbinom(1e4, size = 9, prob = samples_3)
c("flat_prior" = sum(prob_6_9 == 6)/1e4,
  "better_prior" = sum(prob_6_9_3 == 6)/1e4)
```

```
##   flat_prior better_prior
##       0.1766       0.2376
```

Hard

Introduction

```r
library(rethinking)
```

```
## Loading required package: rstan
```

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

```
## Loading required package: StanHeaders
```

```
## rstan (Version 2.10.1, packaged: 2016-06-24 13:22:16 UTC, GitRev: 85f7a56811da)
```

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

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

```
## rethinking (Version 1.59)
```

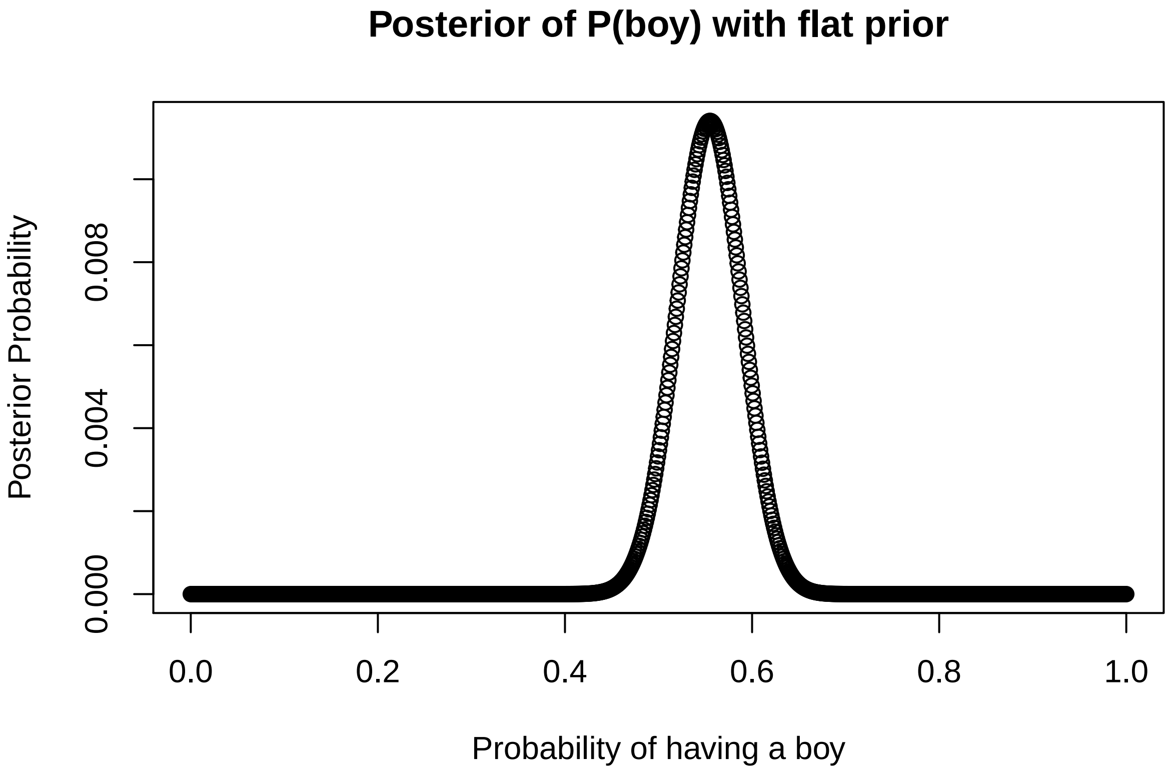
```
data(homeworkch3)
```

3H1

Compute posterior probability of being a boy. Consider each birth in each family as being independent. And using a flat prior

```
child_prior = rep(1, 1000)
boy_like = dbinom(sum(birth1) + sum(birth2), size = 200, prob = p_grid)
boy_post = child_prior * boy_like
boy_post = boy_post/sum(boy_post)
```

```
plot(p_grid, boy_post, xlab = "Probability of having a boy",
     ylab = "Posterior Probability", main = "Posterior of P(boy) with flat prior")
```



```
p_grid[which.max(boy_post)]
```

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

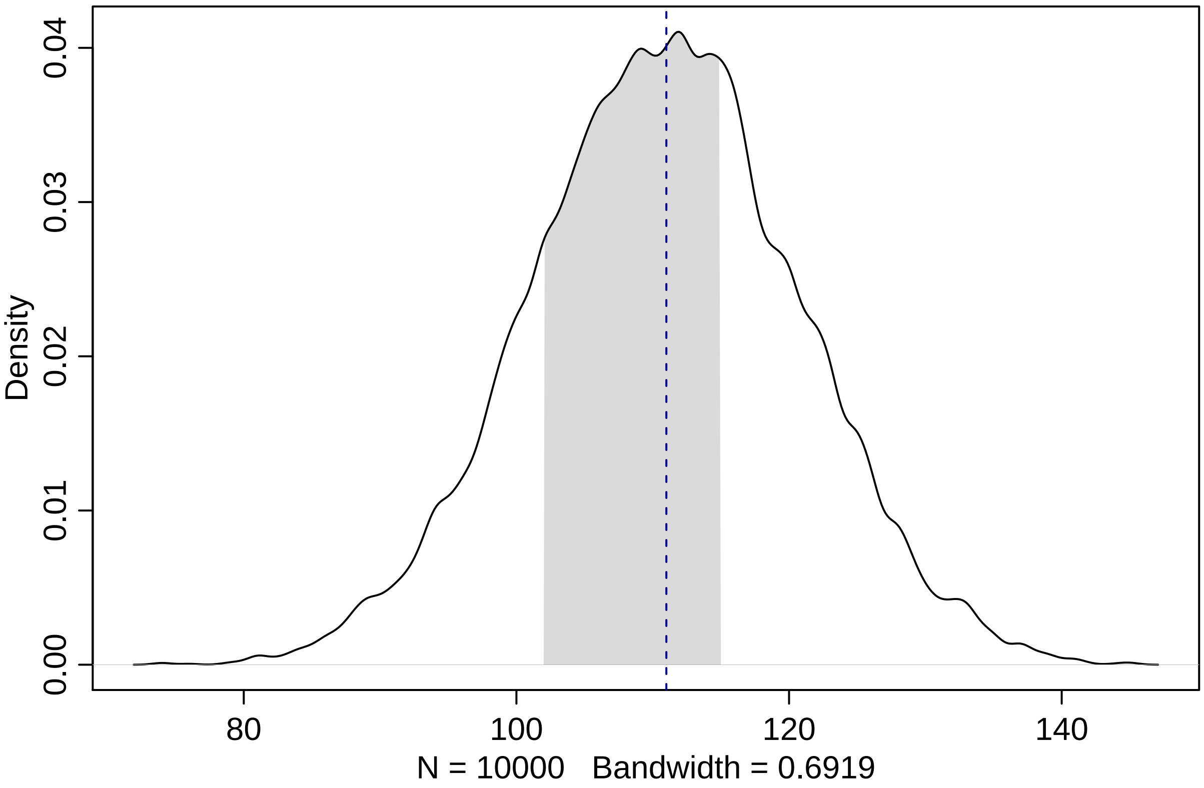
3H2

```
boy_samples = sample(p_grid, prob = boy_post, size = 1e4, replace = TRUE)
boy_hpdi = HPDI(boy_samples, c(0.5, 0.89, 0.97))
boy_hpdi
```

```
##     |0.97     |0.89      |0.5      0.5|     0.89|     0.97|
## 0.4824825 0.4994995 0.5305305 0.5765766 0.6106106 0.6326326
```

3H3

```
boy_random = rbinom(1e4, size = 200, prob = boy_samples)
dens(boy_random, show.HPDI = 0.5)
abline(v = 111, lty = 2, col = "darkblue", main = "Total Model")
```

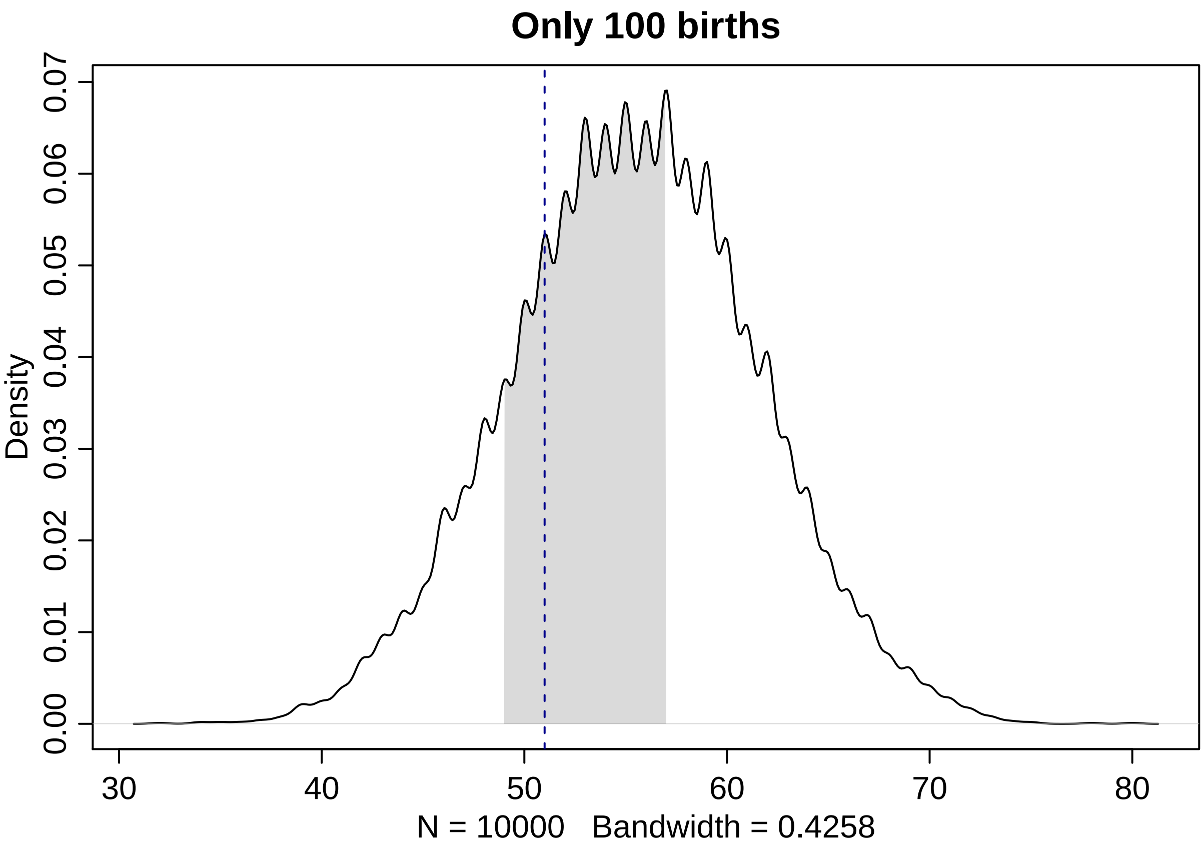


The posterior distribution contains the observed values, and it looks a likely model.

3H4

```
only_100 = rbinom(1e4, size = 100, prob = boy_samples)

dens(only_100, show.HPDI = 0.5, main = "Only 100 births")
abline(v = sum(birth1), lty = 2, col = "darkblue")
```



The model is pretty bad looking at the first birth serie!

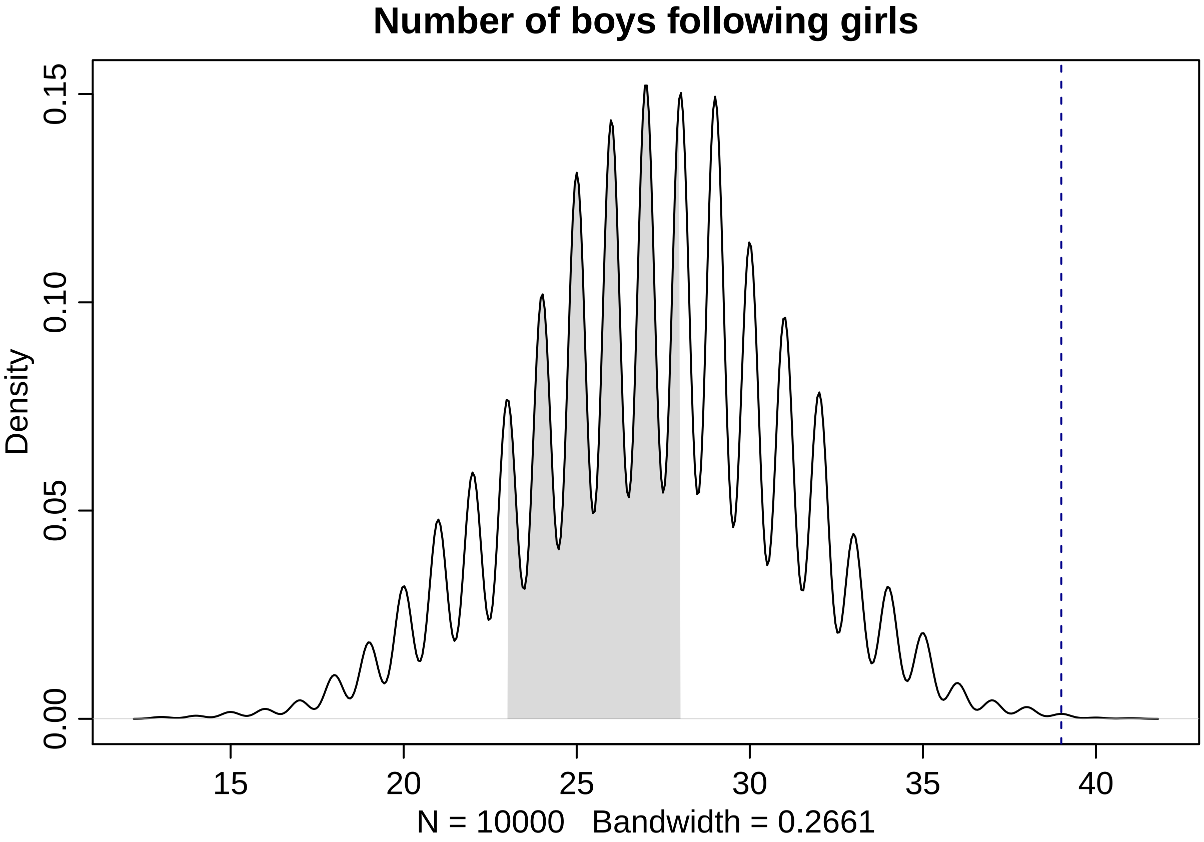
3H5

```
boy_following_girl = birth2[birth1 == 0]
sum(boy_following_girl == 1)
```

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

```
random_following = rbinom(1e4, size = 49, prob = boy_samples)

dens(random_following, show.HPDI = 0.5, main = "Number of boys following girls")
abline(v = sum(boy_following_girl == 1), lty = 2, col = "darkblue")
```



The model completely misses the point, maybe there is abortion when people first have a girl then want a boy, if the second birth would be a girl?