

Chapter 2 — Notes

3.1 Sampling from a grid-approximate posterior

- R Code 3.2:

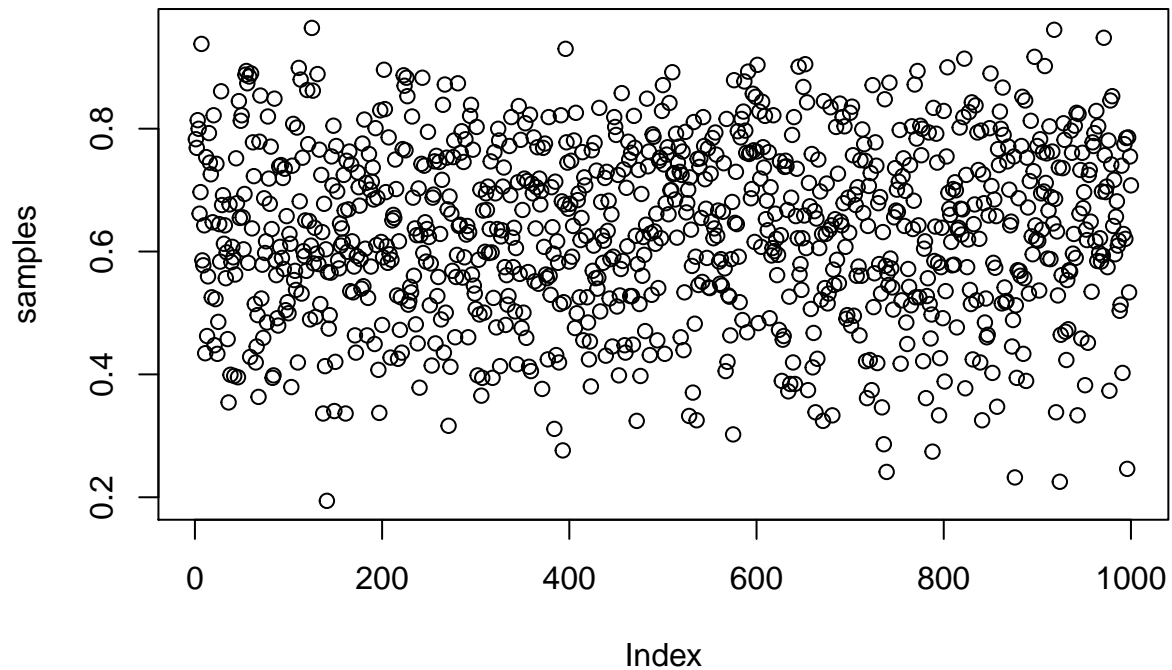
```
n = 1000
p_grid <- seq(from=0, to=1, length.out=n)
prior <- rep(1, n)
likelihood <- dbinom(x=6, size=9, prob=p_grid)
posterior_notnorm <- likelihood * prior
posterior <- posterior_notnorm / sum(posterior_notnorm)
```

Draw 10,000 samples: * R Code 3.3:

```
samples <- sample(p_grid, prob=posterior, size=n, replace=T)
```

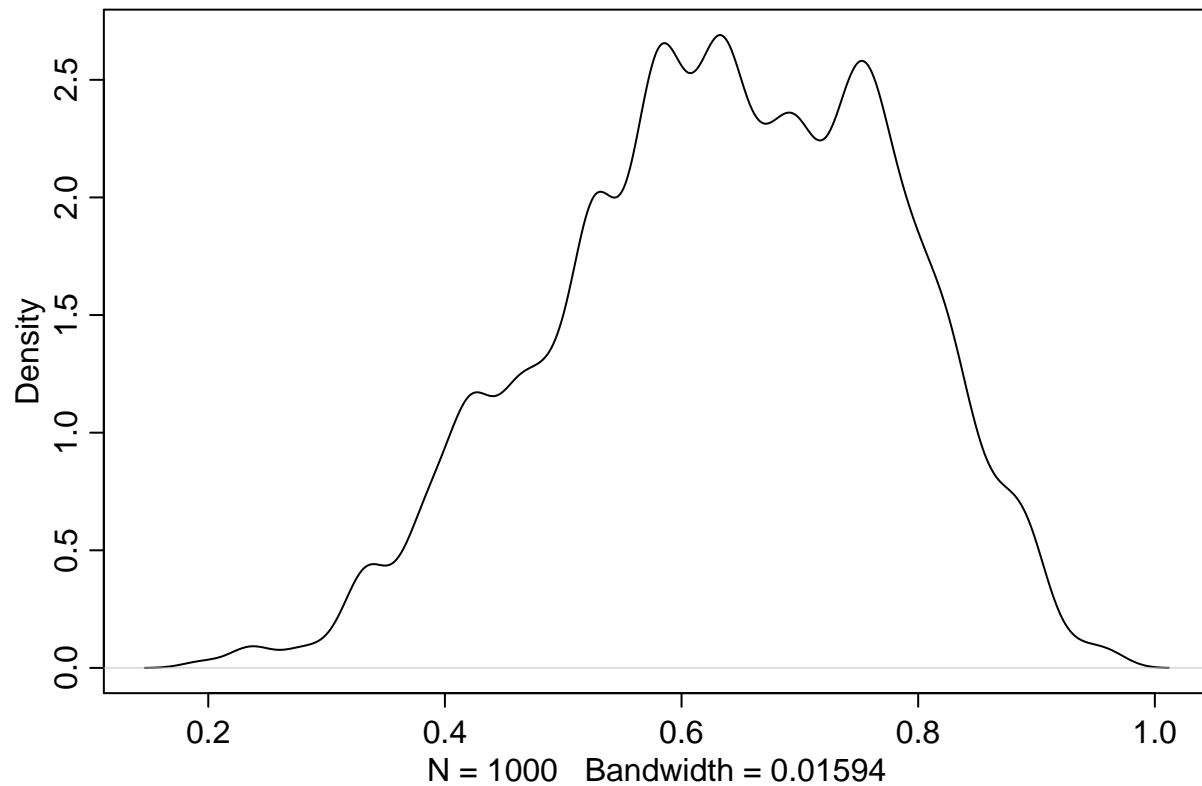
- 3.4:

```
plot(samples)
```



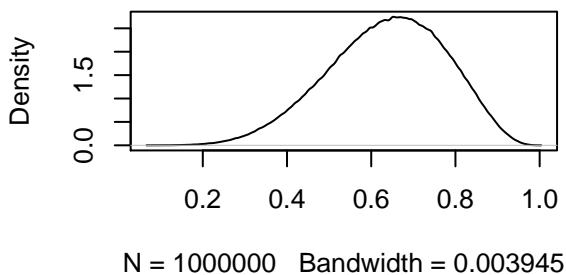
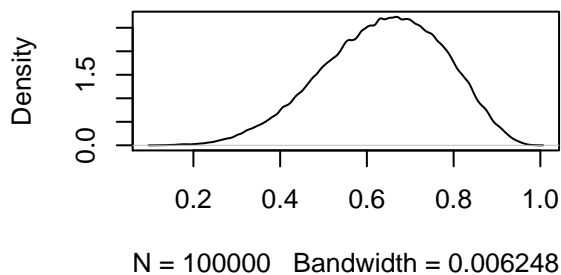
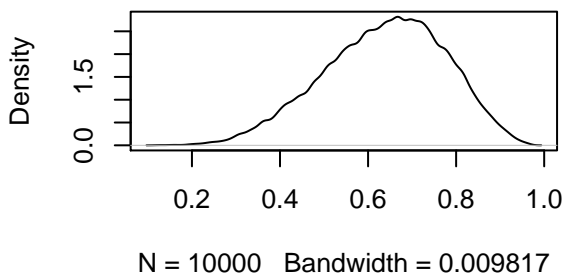
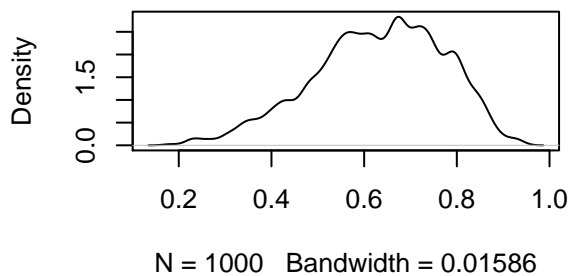
- 3.5:

```
dens(samples)
```



Let's try more samples:

```
par(mfrow=c(2, 2))
dens(sample(p_grid, prob=posterior, size=1e3, replace=T))
dens(sample(p_grid, prob=posterior, size=1e4, replace=T))
dens(sample(p_grid, prob=posterior, size=1e5, replace=T))
dens(sample(p_grid, prob=posterior, size=1e6, replace=T))
```



3.2 Sampling to Summarize

3.2.1. Intervals of defined boundaries.

The posterior probability that the proportion of water is less than 0.5:

- 3.6:

```
p_grid < 0.5
```

```
##      [1]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##     [12]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##     [23]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##     [34]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##     [45]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##     [56]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##     [67]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##     [78]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##     [89]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##    [100]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##    [111]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##    [122]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##    [133]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##    [144]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##    [155]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##    [166]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

[illegible]

```
## [771] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [782] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [793] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [804] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [815] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [826] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [837] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [848] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [859] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [870] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [881] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [892] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [903] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [914] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [925] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [936] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [947] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [958] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [969] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [980] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [991] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

```
sum(posterior[p_grid < 0.5])
```

```
## [1] 0.1718746
```

Samples array:

```
head(samples, 100)
```

```
## [1] 0.7827828 0.7687688 0.8138138 0.7997998 0.6616617 0.6966967 0.9379379
## [8] 0.5855856 0.5775776 0.6426426 0.4344344 0.7537538 0.4624625 0.5595596
## [15] 0.7927928 0.7447447 0.7257257 0.5255255 0.6476476 0.8218218 0.4474474
## [22] 0.5225225 0.7427427 0.4334334 0.4854855 0.6446446 0.5835836 0.8608609
## [29] 0.6756757 0.6916917 0.6126126 0.6426426 0.5565566 0.5985986 0.4574575
## [36] 0.3543544 0.6766767 0.3993994 0.5825826 0.6076076 0.5905906 0.3973974
## [43] 0.5625626 0.7517518 0.6786787 0.3953954 0.8448448 0.6546547 0.8128128
## [50] 0.8218218 0.6556557 0.6036036 0.6936937 0.8878879 0.8938939 0.8738739
## [57] 0.5815816 0.8848849 0.4284284 0.8898899 0.6376376 0.7777778 0.7227227
## [64] 0.5145145 0.4194194 0.4454454 0.4964965 0.3633634 0.7787788 0.8538539
## [71] 0.5235235 0.5785786 0.4594595 0.6866867 0.5975976 0.6166166 0.4844845
## [78] 0.8198198 0.7187187 0.6776777 0.7707708 0.6366366 0.3943944 0.3983984
## [85] 0.8488488 0.5855856 0.5615616 0.4914915 0.4794795 0.7387387 0.6076076
## [92] 0.7407407 0.5895896 0.7197197 0.5715716 0.7347347 0.5045045 0.6576577
## [99] 0.5175175 0.4994995
```

The same calculation using samples. Add up all samples that lie in the grid < 0.5 , and divide by the total number of samples to get the frequency \sim probability:

- 3.7:

```
n = 1e4
samples = sample(p_grid, prob=posterior, size=n, replace=T)
sum(samples < 0.5) / n
```

```
## [1] 0.1691
```

How much probability lies between 0.5 and 0.75: * 3.8:

```
sample_points = sum(samples > 0.5 & samples < 0.75)
sample_points
```

```
## [1] 6061
```

```
sample_points / n
```

```
## [1] 0.6061
```

3.2.2. Intervals of defined mass.

Boundaries of the lower 80% posterior probability lies:

- 3.9:

```
quantile(samples, probs = .8)
```

```
##          80%
```

```
## 0.7607608
```

Middle 80%, i.e. lying between 10% and 90%:

```
# 3.10
```

```
quantile(samples, probs = c(0.1, 0.9))
```

```
##          10%          90%
```

```
## 0.4494494 0.8108108
```

The above are PERCENTILE INTERVALS. Percentiles can be misleading if the distribution is highly skewed.

```
# 3.11
```

```
n <- 1000
```

```
p_grid <- seq(0, 1, length.out = n)
```

```
prior <- rep(1, n)
```

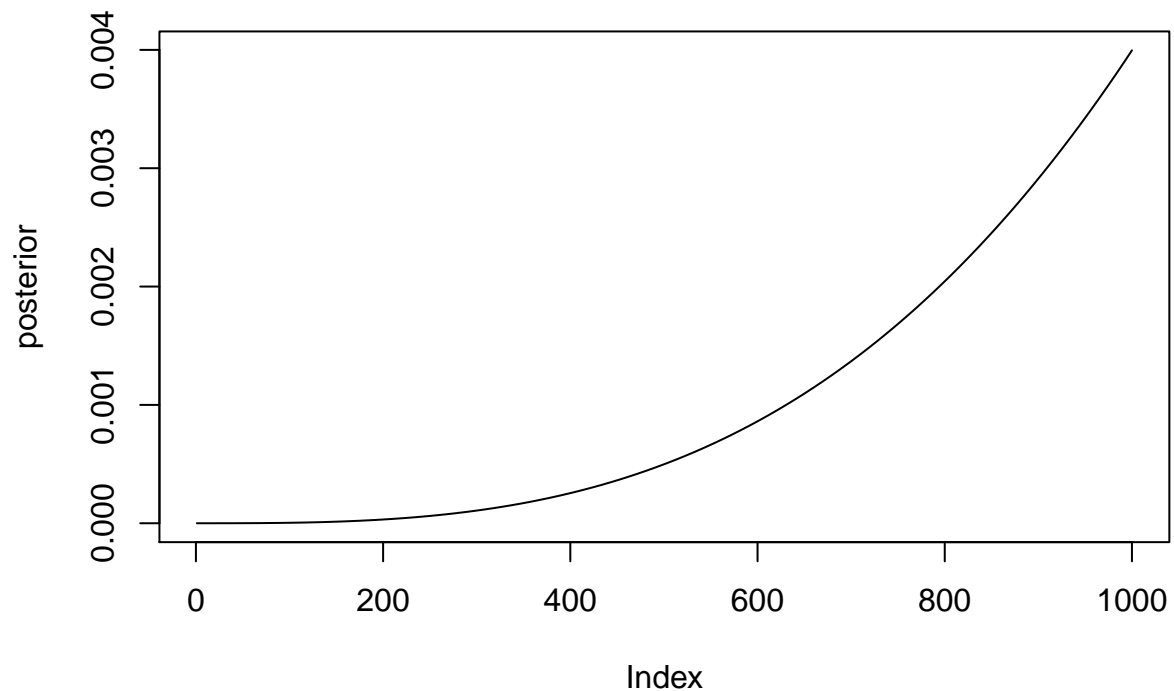
```
likelihood <- dbinom(3, size=3, prob=p_grid)
```

```
posterior_notnorm <- likelihood * prior
```

```
posterior <- posterior_notnorm / sum(posterior_notnorm)
```

```
samples <- sample(p_grid, size=1e4, replace=T, prob=posterior)
```

```
plot(posterior, type='l')
```



```
# 3.12
```

```
PI(samples, prob=0.5)
```

```
##      25%      75%
## 0.7087087 0.9329329
```

Highest Posterior Density Interval described the distribution better. It's the *narrowest* interval containing the specified probability mass, e.g. 50%.

```
# 3.13
```

```
HPDI(samples, prob=0.5)
```

```
##      |0.5      0.5|
## 0.8438438 1.0000000
```

3.2.3. Point Estimates

A parameter with the highest posterior probability is called a *maximum a posteriori* estimate, or *MAP*.

```
# 3.14
```

```
which.max(posterior)
```

```
## [1] 1000
```

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

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

Use samples to get the same (or similar) result:

```
# 3.15  
chainmode(samples, adj=0.01)
```

```
## [1] 0.9722671
```

```
# 3.16  
mean(samples)
```

```
## [1] 0.8009554
```

```
median(samples)
```

```
## [1] 0.8438438
```