# STATISTICAL RETHINKING WINTER 2019
# HOMEWORK, WEEK 1 SOLUTIONS

**1.** Really all you need is to modify the grid approximation code in Chapter 3. If you replace 6 with 8 and 9 with 15, it'll work:

```
p_grid <- seq( from=0 , to=1 , length.out=1000 )
prior <- rep( 1 , 1000 )
prob_data <- dbinom( 8 , size=15 , prob=p_grid )
posterior <- prob_data * prior
posterior <- posterior / sum(posterior)
set.seed(100)
samples <- sample( p_grid , prob=posterior , size=1e4 , replace=TRUE )
```

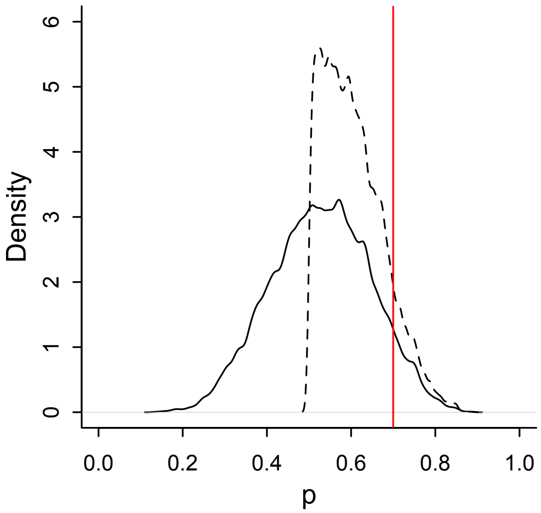
The posterior mean should be about 0.53 and the 99% percentile interval from 0.24 to 0.81.

**2.** Modifying only the prior:

```
p_grid <- seq( from=0 , to=1 , length.out=1000 )
prior <- c( rep( 0 , 500 ) , rep( 1 , 500 ) )
prob_data <- dbinom( 8 , size=15 , prob=p_grid )
posterior <- prob_data * prior
posterior <- posterior / sum(posterior)
set.seed(100)
samples2 <- sample( p_grid , prob=posterior , size=1e4 , replace=TRUE )
```

The posterior mean should be about 0.61 and the 99% interval 0.50 to 0.82. This prior yields a posterior with more mass around the true value of 0.7. This is probably easier to see in a plot:

```
dens( samples , xlab="p" , xlim=c(0,1) , ylim=c(0,6) )
dens( samples2 , add=TRUE , lty=2 )
abline( v=0.7 , col="red" )
```



With the impossible values less than 0.5 ruled out, the second model piles up more plausibility on the higher values near the true value. The data are still misleading it to think that values just above 0.5 are the most plausible. But the posterior mean of 0.63 is much better than 0.53 from the previous problem.

Informative priors, when based on real scientific information, help. Here, the informative prior helps because there isn't much data. That is common in a lot of fields, ranging from astronomy to paleontology.

**3.** One way to approach this problem is to try a range of sample sizes and to plot the interval width of each. Here's some code to compute the posterior and get the interval width:

```
set.seed(100)
N <- 20
p_true <- 0.7
W <- rbinom( 1 , size=N , prob=p_true )
p_grid <- seq( from=0 , to=1 , length.out=1000 )
prior <- rep( 1 , 1000 )
prob_data <- dbinom( W , size=N , prob=p_grid )
posterior <- prob_data * prior
posterior <- posterior / sum(posterior)
samples <- sample( p_grid , prob=posterior , size=1e4 , replace=TRUE )
PI99 <- PI( samples , 0.99 )
as.numeric( PI99[2] - PI99[1] )
```

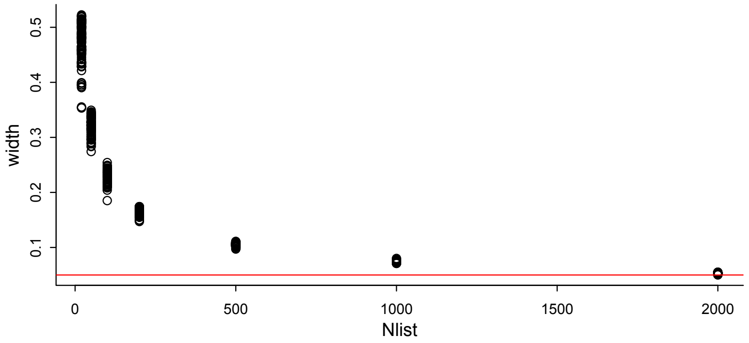
There are other ways to compute the interval width. But the above is closest to the code in the book. Now since we want to do this for different values of N, it's nice to make this into a function:

```
f <- function( N ) {
    p_true <- 0.7
    W <- rbinom( 1 , size=N , prob=p_true )
    p_grid <- seq( from=0 , to=1 , length.out=1000 )
    prior <- rep( 1 , 1000 )
    prob_data <- dbinom( W , size=N , prob=p_grid )
    posterior <- prob_data * prior
    posterior <- posterior / sum(posterior)
    samples <- sample( p_grid , prob=posterior , size=1e4 , replace=TRUE )
    PI99 <- PI( samples , 0.99 )
    as.numeric( PI99[2] - PI99[1] )
}
```

Now if you enter f(20), you get an interval width for 20 globe tosses. Now notice that the interval width varies across simulations. Try f(20) a few times to see what I mean. But as you increase N, this variation shrinks rapidly. This is because as the sample size increases, the differences between samples shrink. So if you ignore the sample to sample variation in interval width, that's okay in this example. But in the code below, I'll account for it.

Now we need to run simulations across a bunch of different sample size to find where the interval shrinks to 0.05 in width. I'll use sapply to run 100 simulations at each of 7 sample sizes:

```
Nlist <- c( 20 , 50 , 100 , 200 , 500 , 1000 , 2000 )
Nlist <- rep( Nlist , each=100 )
width <- sapply( Nlist , f )
plot( Nlist , width )
abline( h=0.05 , col="red" )
```



What are we looking at in this plot? The horizontal is sample size. The points are individual interval widths, one for each simulation. The red line is drawn at a width of 0.05. Looks like we need more than 2000 tosses of the globe to get the interval to be that precise.

The above is a general feature of learning from data: The greatest returns on learning come early on. Each additional observation contributes less and less. So it takes very much effort to progressively reduce our uncertainty. So if your application requires a very precise estimate, be prepared to collect a lot of data. Or to change your approach.