STATISTICAL RETHINKING 2025
WEEK 1 SOLUTIONS

1. On the surface, this is the same calculation as for the proportion of water on the globe, described in lecture and the textbook. Under the surface, it is tricky to think through how an outcome relates to "honesty" in the population of participants. As with many of these solution sets, I want to give you a solution that is thorough and has a lot of explanation. Your own solutions can be simpler and partial and still satisfy me that you understand the material. If there are things here you didn't see, that's fine. Don't be hard on yourself.

You can use grid approximation or just the Beta distribution directly. I'll show both. The grid approximation can reuse the code from the chapter. Here is a simple implementation.

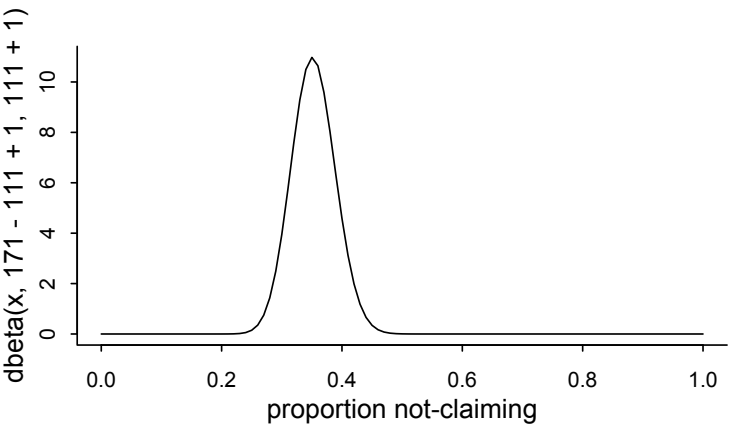
```
compute_posterior <- function( Y , N , grid ) {
    ways <- sapply( grid , function(q) q^Y * (1-q)^N )
    post <- ways/sum(ways)
    data.frame( grid , ways , post=round(post,3) )
}
compute_posterior( 171-111 , 111 , grid=seq(from=0,to=1,len=21) )
```

```
   grid          ways  post
1  0.00  0.000000e+00 0.000
2  0.05  2.920925e-81 0.000
3  0.10  8.335248e-66 0.000
4  0.15  5.382404e-58 0.000
5  0.20  2.017383e-53 0.000
6  0.25  1.019062e-50 0.007
7  0.30  2.711176e-49 0.197
8  0.35  7.541630e-49 0.549
9  0.40  3.150564e-49 0.229
10 0.45  2.360529e-50 0.017
11 0.50  3.340956e-52 0.000
12 0.55  8.479113e-55 0.000
13 0.60  3.294079e-58 0.000
14 0.65  1.466744e-62 0.000
15 0.70  4.638117e-68 0.000
16 0.75  4.731695e-75 0.000
17 0.80  3.978586e-84 0.000
```

```
18 0.85  2.047695e-96 0.000
19 0.90 1.797010e-114 0.000
20 0.95 1.774544e-146 0.000
21 1.00  0.000000e+00 0.000
```

Using the Beta distribution means we have a mathematical expression for it. You can plot the posterior with:

```
curve( dbeta(x,171-111+1,111+1) , from=0 , to=1 ,
    xlab="proportion not-claiming" )
```



Or use `dbeta(x,171-111+1,111+1)` to compute the posterior probability of any value `x`.

However you computer the posterior, the posterior probability of not-claiming the prize is maximized around 0.35. And the posterior mass is very concentrated around that value—no less than about 0.30 and no more than about 0.45. It should be about 0.50, if everyone were honest.

So if the posterior mass is concentrated around 0.35, what does that mean? It is reasonable to wonder and feel a little confused. As always, if you are confused, that just means you are paying attention. Confusion is what thinking feels like.

I like to address my confusion by considering generative stories about the data. Suppose for example a population in which half the participants are cheats. The cheats always claim the prize, whatever the result of the die toss. The rest of the population is honest. What proportion of participants should we expect to claim the prize? The half that are cheats claim it, and then half of the other half claim it (in expectation). So we expect $0.5 \times 1 + 0.5 \times 0.5 = 0.75$ of the participants to claim the prize. Or 0.25 not to claim it. So what does 0.35 mean? It means fewer than 50% are cheats. But how many fewer? You can mess around with the calculation above and find that a proportion 0.30

of cheats in the population yields an expectation of 0.35 of the sample not claiming the prize.

Let's try rewriting the calculation so it estimates what we want, the proportion of cheats, and not the proportion not claiming the prize. The way to do this is to construct the probability of observing a claim as $p = q + (1-q) \times 0.5$, where q is the proportion of cheats. As code:

```
compute_posterior2 <- function( Y , N , grid ) {
    ways <- sapply( grid , function(q)
     {p <- q+(1-q)*0.5; return(p^Y * (1-p)^N)} )
    post <- ways/sum(ways)
    data.frame( grid , ways , post=round(post,3) )
}
compute_posterior2( 111 , 171-111 , grid=seq(from=0,to=1,len=21) )
```

```
   grid         ways  post
1  0.00 3.340956e-52 0.000
2  0.05 3.461778e-51 0.001
3  0.10 2.360529e-50 0.009
4  0.15 1.062837e-49 0.039
5  0.20 3.150564e-49 0.115
6  0.25 6.089168e-49 0.222
7  0.30 7.541630e-49 0.274
8  0.35 5.830697e-49 0.212
9  0.40 2.711176e-49 0.099
10 0.45 7.202452e-50 0.026
11 0.50 1.019062e-50 0.004
12 0.55 6.973328e-52 0.000
13 0.60 2.017383e-53 0.000
14 0.65 2.035500e-55 0.000
15 0.70 5.382404e-58 0.000
16 0.75 2.385035e-61 0.000
17 0.80 8.335248e-66 0.000
18 0.85 5.564269e-72 0.000
19 0.90 2.920925e-81 0.000
20 0.95 4.528067e-98 0.000
21 1.00 0.000000e+00 0.000
```

And there is the 0.30 estimate, but now with the entire posterior so the uncertainty is correctly quantified. So about 0.30 of the population is estimated to be cheats. So $1 - 0.30 = 0.70$ are honest.

It turns out we can convert from the raw proportion of claims to this estimate using an ancient technology known as algebra. The expected proportion of claims p is:

$$p = q + (1 - q)\frac{1}{2}$$

We can solve this equation for q, the proportion of cheaters:

$$q = 2p - 1$$

Plugging in our previous estimate of $p = 1 - 0.35$:

$$q = 2(1 - 0.35) - 1 = 0.30$$

The code I wrote above does this algebra for you, but maybe it helps to understand that there is no magic.

This analysis isn't completely general though. Suppose there are other types of people in the population. Suppose there are people who never claim the prize. You might think these people don't exist, but participants do refuse rewards, so it's not impossible. Now if say 30% the population are cheats and 40% are honest and the remaining 30% never claim the prize, we expect $0.3 + 0.4 \times 0.5 + 0.3 \times 0 = 0.8$ claiming the prize. So which strategies are in the population needs to be reflected in our model, in order to get the right inference.

All of this is meant to be an example of the importance of thinking hard about the data generating model. This helps us interpret and justify the estimates we produce. No estimate can be easily interpreted in the absence of a data generating model.

2. This one is easier, because once you have the estimate of the proportion of claims, you can just make predictions of claims in the next 10 participants. There is no worry about which of these claims are honest—we can't tell anyway, and I didn't ask you to decide. Let's sample from the Beta distribution and then simulate claims:

```
p_samples <- rbeta(1e4,111+1,171-111+1)
Y_sim <- rbinom(1e4,size=10,p=p_samples)
```

I used the rbinom() function, but you could use sample() and then tally the water points. The resulting distribution is approximated by the counts in W_sim. You can view the distribution with:

```
plot(table(Y_sim))
```

But the text table works just as well:

```
Y_sim
   0    1    2    3    4    5    6    7    8    9   10
   2    5   49  267  749 1547 2318 2403 1722  786  152
```

So we expect 5 to 8 claims in the next 10 participants, although more extreme numbers are possible. This is important for example in planning the amount of money needed to continue the experiment.