Computational Statistics & Probability

Neelesh Bhalla

2022-11-15

1. COVID Home Test

A home COVID-19 antigen test developed by BinaxNOW is accepted in the US for travel. [BinaxNOW reports](https://www.globalpointofcare.abbott/en/product-details/binaxnow-covid-19-antigen-self-test-us.html) that in a clinical trial their test correctly gave a positive result 84.6% of the time and correctly gave a negative result 98.5% of the time. We are assuming a PCR result is ground truth: if a PCR test is positive, then the patient has COVID-19; if a PCR test is negative, then the patient does not have COVID-19. Suppose that 10% of the people in your community are currently infected. (This is your base rate for exposure.) Now suppose you take a BimaxNOW COVID-19 antigent test.

a) Given that your BinaxNOW test is positive, what is the probability that you have COVID-19?

```
pr_Pos_givenCov <- 0.846
pr_Neg_givenNoCov <- 0.985
pr_Cov <- 0.1
pr_Pos <- pr_Pos_givenCov * pr_Cov + (1 - pr_Neg_givenNoCov) * (1 - pr_Cov)
pr_Cov_givenPos <- pr_Pos_givenCov * pr_Cov / pr_Pos
pr_Cov_givenPos
```

```
## [1] 0.8623853
```

b) Given that your BinaxNOW test is negative, what is the probability that you have COVID-19?

```
pr_NoCov = 1 - pr_Cov
pr_Neg_givenCov <- 1 - pr_Pos_givenCov
pr_Cov_givenNeg <- pr_Neg_givenCov * pr_Cov / pr_NoCov
pr_Cov_givenNeg
```

```
## [1] 0.01711111
```

c) Suppose instead the base rate for infection is 30%. This is extreme: The peak daily number of cases in Germany (so far) was 250,000 in April 2022. People are considered infectious for 10 days. So, no more than 2.5M people were infected with COVID in Germany during this peak, which is 3% of the population. Do you think a negative test result of a BinaxNOW home test is sufficient to conclude that you do not have COVID-19? Why or why not?

```
# For this, we evaluate "the conditional probability
# of actually having Covid-19 given the test is negative"
new_pr_Cov <- 0.3
pr_Neg <- (1 - pr_Pos_givenCov) * new_pr_Cov + pr_Neg_givenNoCov * (1 - new_pr_Cov)
new_pr_Cov_givenNeg <- pr_Neg_givenCov * new_pr_Cov / pr_Neg
new_pr_Cov_givenNeg
```

```
## [1] 0.06279734
```

```
# Thus there is 6.3% chance of actually having Covid-19
# despite being tested negative"
```

2. Swing Voters

Imagine a country where there are only two political parties, Red and Blue, which divide the electorate equally. One difference between registered Blue voters and registered Red voters is their willingness to vote for the opposing party's candidate. Blue voters vote Red 20% of the time, otherwise they vote Blue. Red voters vote Blue 10% of the time, otherwise they vote Red. Voters who switch are called swing voters. Smith was a swing voter in the last election but you do not know whether he is Red or Blue. (Nobody changes parties.) What is the probability that Smith will be a swing voter in the next election?

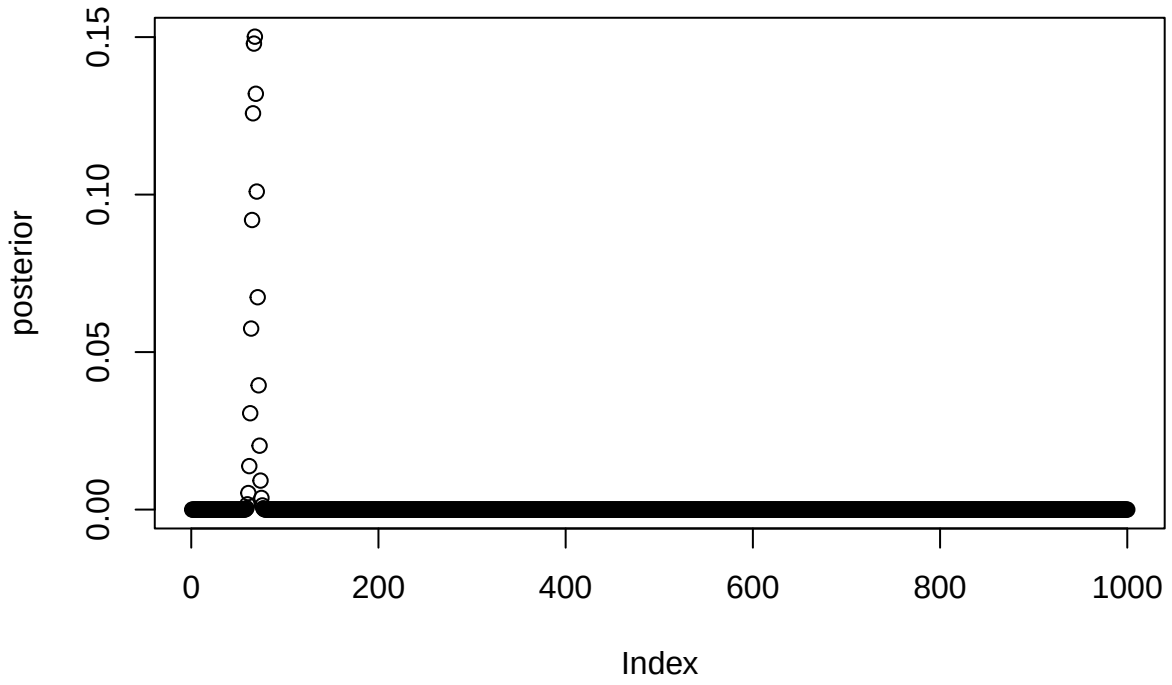
```
pr_SV_givenB <- 0.2
pr_SV_givenR <- 0.1
# Given Smith swinged last year

pr_Smith_is_R <- pr_SV_givenR / (pr_SV_givenR + pr_SV_givenB)
pr_Smith_is_B <- pr_SV_givenB / (pr_SV_givenR + pr_SV_givenB)

pr_Smith_swings_again <- (pr_Smith_is_R * pr_SV_givenR) + (pr_Smith_is_B * pr_SV_givenB)
pr_Smith_swings_again
```

```
## [1] 0.1666667
```

3. More Precision

Suppose you want a very precise estimate of the proportion of the Earth's surface that is covered in water. Specifically, suppose you would like the 99% percentile interval of the posterior distribution of p(the estimated proportion of water) to have a width of no greater than 0.05 – that is, the distance between the lower and upper bound on pshould be no greater than 0.05. How many times must you toss the globe to achieve this precision? An exact count is unnecessary. I am primarily interested in your approach.

```
p_grid <- seq( from=0 , to=1 , length.out=1000 )
prior <- rep( 1 , 1000 )
likelihood <- dbinom( 600 , size=9000 , prob=p_grid )
posterior <- likelihood * prior
posterior <- posterior / sum(posterior)
plot(posterior)
```




```
quantile( posterior , 0.8 )
```

```
##          80%
## 1.82037e-305
```

```
sum( posterior < 0.99 ) / 1e4
```

```
## [1] 0.1
```