Problem1

April 14, 2021

1 Problem 1: Bayesian COVID Pool Testing

We seek to determine what the probability of being COVID positive if your weekly pool test shows a positive result.

Bayes Theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- $P(A|B)$: Posterior Probability - Probability of event A occuring given B is true - $P(B|A)$: Conditional Probability - Probability of event B occuring given A is true - $P(A)$: A priori probability that event A occurs - $P(B)$: A priori probability that event B occurs

For the COVID Pool test: - $P(A)$: Probability of an individual having COVID - $P(B)$: Probability of a pool test returning positive result - $P(A|B)$: Probability of an individual having COVID given a positive pool test - $P(B|A)$: Probability of a pool test being positive given a COVID positive student in the pool.

We don't have values for $P(B)$ so we rewrite Bayes theorem as,

$$P(A|B) = \frac{P(B|A)P(A)}{P(B|A)P(A) + P(B|A^c)P(A^c)}$$

where, - $P(A^c)$: Probability of an individual **not** have COVID - $P(B|A^c)$: Probability of a pool test returning a positive result given the individual does not have COVID. This implies some or all of the other people in the pool are COVID positive *except* the individual A.

We assume, - $P(A_B) = 1$ -> if at least one person in the pool is positive, the pool test should return a positive result - $P(A) = 1/76$ -> 1 in every 76 people in Erie County is infected with COVID (19andMe), and - $P(A^c) = 1 - 1/76$.

The last unknown probability that is needed for this calculation $P(B|A^c)$. This can be calculated with a Binomial Distribution,

$$\begin{aligned} P(x; p, n) &= \binom{n}{x} (p)^x (1-p)^{n-x} \quad for\, x = 0, 1, 2, ..., n \\ &= \frac{n!}{x!(n-x)!} (P(A))^x (1 - P(A))^{n-x} \end{aligned}$$

In each pool, there are 12 samples so we set n=12.

Alternatively, the website 19andMe has a personal risk calculator. For me, the risk of contracting COVID is, - $P(A)_{alt} = 0.0002$.

We will use both values for $P(A)$

References: - https://19andme.covid19.mathematica.org/

[15]:
```
import numpy
import tensorflow as tf
import tensorflow_probability as tfp
```

1.0.1 P(A|B) for Average Erie Citizen

[46]:
```
n = 12 # Number of people in a test pool
P_A = 1/76 # Probability of an individual testing positive for Covid
P_B = 1 - P_A
P_BA = 1
```

[47]:
```
Bin = tfp.distributions.Binomial(n, logits=None, probs=P_A)

P_BAc = 0
for i in range(1,n-1):
    P_BAc = P_BAc + Bin.prob(i)

P_AB = (P_BA * P_A )/((P_BA*P_A)+(P_BAc*P_B))
```

1.0.2 P(A|B) for Me

[31]:
```
P_A_alt = 0.0002
P_B_alt = 1 - P_A_alt
```

[36]:
```
Bin_alt = tfp.distributions.Binomial(n, logits=None, probs=P_A_alt)
P_BAc_alt = 0
for i in range(1,n-1):
    P_BAc_alt = P_BAc_alt + Bin_alt.prob(i)

P_AB_alt = (P_BA * P_A_alt )/((P_BA * P_A_alt) + (P_BAc_alt * P_B_alt))
```

1.1 Results

[45]:
```
print("Average Erie County Resident:")
tf.print("P(A) = 1.32[%]: P(A|B) = ",P_AB*100,"[%]\n")
print("Personal Risk:")
tf.print("P(A) = 0.02[%]: P(A|B) = ",P_AB_alt*100,"[%]")
```

```
Average Erie County Resident:
P(A) = 1.32[%]: P(A|B) =  8.31835365 [%]
```

```
Personal Risk:
P(A) = 0.02[%]: P(A|B) =  7.70153618 [%]
```