Rajan Puri — Shameem Ahmed Khan — Orkun Akyol — Gabriel Brown
Statistics — Winter 2024
Homework 1

1 A medical test for disease D has outcomes + (positive) and - (negative). We assume that

- **the probability for an individual to have the disease is 0.01**
- **the probability of a positive test, given that the individual has the disease, is 0.9**
- **the probability of a negative test, given that the individual does not have the disease, is 0.9**

Compute the probability that an individual has the disease, given that the individual has tested positive. Comment on the quality of the test.

P(has disease | positive test) = P(positive test | has disease) P(has disease) / P(positive test)
$= 0.9 * 0.01/$ P(positive test)
P(positive test) = P(positive test | has disease) P(has disease) + P(positive test | doesn't have disease) P(doesn't have disease)
= P(positive test | has disease) P(has disease) + (1 - P(negative test | doesn't have disease)) P(doesn't have disease)
$= 0.9 * 0.01 + (1 - 0.9) * .99 = 0.108$
Therefore, the probability that a patient has the disease given a positive test $= 0.9 * 0.01/0.108 = 0.0833$ or 8.33%
Therefore, the test is not very useful in diagnosing a patient.

2 The Monty Hall problem is a probability puzzle based on a game show scenario:

- **There are three doors: behind one door is a prize (e.g., a car) and behind the other two doors are goats.**
- **A contestant chooses one of the three doors.**
- **The host, who knows what is behind each door, opens one of the two remaining doors with a goat behind it.**
- **The contestant is then given the option to either stick with their initial choice or switch to the other unopened door.**

Your task is to write a program (using Python or any other programming language of your choice) to simulate the probability of winning the prize if the contestant always switches their initial choice after the host reveals a goat. You can model the scenario as follows:

- **The prize is randomly assigned to one of the three doors (labeled 1,2 or 3) in a uniformly random manner.**
- **The contestant randomly selects one of the doors (1, 2 or 3) with equal probability.**
- **One of the unchosen doors that does not have the prize behind it is eliminated.**
- **The contestant switches their initial choice to the remaining unopened door.**

Simulate the above process 105 times and track the number of times the contestant wins the prize by switching their initial choice. Based on this data, compute the proportion of times the contestant wins using this strategy. In general, do you have a better chance at winning the prize by changing your initial choice?

```python
import random as rm

def checkProb(total):
    correct_answer = 0
    # loop for total number
    for i in range(total):
        door = [1, 0, 0]
        # shuffle door every time
        rm.shuffle(door)
        print(f"Here are the option for door: {door}")
        # taking one value out of 3 options
        person_choice = rm.randint(0, 2)
        print(f"Here is person_choice: {person_choice}")

        # person has chosen, and now host will open one
        host_choose = [i for i in range(3) if i != person_choice and door[i] != 1]
        print(f"value after host taken one out {host_choose}")

        # person now has to take one out of the left 2 choices
        # shuffling again
        rm.shuffle(host_choose)

        # host takes out the first
        host_choice = host_choose[0]

        # checking if it is correct
        value = [i for i in range(len(door)) if i != host_choice and i != person_choice]
        print(f"value is {value[0]}")

        if door[value[0]] == 1:
            correct_answer += 1

    print(f"correct_answer is {correct_answer}")

    return correct_answer / total

if __name__ == "__main__":
    # total number of times 100000
    total = 10**5

    result = checkProb(total)

    print(f"total prob is {result}")
```

3 Let X be a continuous real-valued random variable with the probability density function $f : \mathbb{R} \to \mathbb{R}$,

$$f(x) = \begin{cases} 0 & \textbf{if } x < 0 \\ \frac{4xe^{-x^2}}{(1+e^{-x^2})^2} & \textbf{if } x \geq 0 \end{cases}$$

(a) **Compute the cumulative distribution function $F(x) = \mathbb{P}(X \leq x)$ of X for $x \in \mathbb{R}$**

For all $x < 0$, $F(x) = 0$ trivially

For all $x \geq 0, F(x) = \int_0^x f(u)du$

$= \int_0^x \frac{4ue^{-u^2}}{(1+e^{-u^2})^2} du$

$= \frac{-2}{e^{u^2}+1}|_0^x = \frac{e^{x^2}-1}{e^{x^2}+1} = \tanh \frac{x^2}{2}$

Therefore, $F(x) = \begin{cases} 0 & x < 0 \\ \tanh(\frac{x^2}{2}) & x \geq 0 \end{cases}$

(b) **Solve the quantile function of X**

Given $q \in (0,1), \exists x$ such that $F(x) = q$

$\implies \tanh(\frac{x^2}{2}) = q, \forall q \in (0,1)$

$\implies x = \sqrt{2\tanh^{-1}(q)}$

$\implies F_X^{-1}(q) = \sqrt{2\tanh^{-1}(q)}$ for $q \in (0,1)$

(c) **Compute the probability $\mathbb{P}(0 < X < 1)$. Which value $a \in \mathbb{R}$ satisfies $\mathbb{P}(X \leq a) = 0.95$**

$P(0 < X < 1) = F(1) - F(0) = \tanh(1/2) \approx 0.462$

by the definition of the quantile function, $a = F_X^{-1}(.95) = \sqrt{2\tanh^{-1}(.95)} \approx 1.91404$

4

(a) The sample space is the following set with four members:

$$\omega = \{(0,0),(1,0),(0,1),(1,1)\}$$

By simple enumeration, it follows:

$$p_X(x) = \begin{cases} 1/4 & \text{if } x = 0 \\ 1/2 & \text{if } x = 1 \\ 1/4 & \text{if } x = 2 \end{cases}$$

(b) Trivially, $F(x) = 0$ for $x < 0$ and $F(x) = 1$ for $x \geq 2$. By cumulatively summing the probabilities in the probability mass function, we get:

$$F(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1/4 & \text{if } 0 \leq x < 1 \\ 3/4 & \text{if } 1 \leq x < 2 \\ 1 & \text{if } x \geq 2 \end{cases}$$

(c) $F^{-1}(q) = \begin{cases} 0 & \text{if } 0 < q \leq 1/4 \\ 1 & \text{if } 1/4 < q \leq 3/4 \\ 2 & \text{if } 3/4 < q < 1 \end{cases}$