hw1_S2023_pt2

January 27, 2023

1 Homework 1 Part 2

This is an individual assignment.

1.1 Description

Create or edit this Jupyter Notebook to answer the questions below. Use simulations to answer these questions. An analytical solution can be useful to check if your simulation is correct but analytical solutions alone will not be accepted as a solution to a problem.

Problem 9 Consider repeatedly rolling a fair 4-sided die.

1. Create a simulation to compute the probability that the top face will be 4 at least once on four rolls of the die?
2. Create a simulation to compute the probability that the top face will be 4 at least once on 20 rolls of the die?
3. Create a simulation to compute how many rolls of the die would you have to do to be 90% confident that you would see at least one 4?
4. Using the formula you have computed in problem 2 part 4, make a Python function that takes in the target value p and outputs the required number of rolls of an integer.
    1. Find the values for $p = 0.95$ and $p = 0.99$.
    2. Use your simulation to verify that the number of rolls you specified is sufficient to achieve $p \geq 0.95$.

[1]:
```
import random

# 1. Simulation that top face will be 4 at least once on four rolls of die
num_sims = 10000

hasFour = 0
for sim in range(num_sims):
    for roll in range(4):
        if random.choice(range(1,5)) == 4:
            hasFour += 1
            break
print("The probability that 4 will appear at least once on four rolls of the␣
 ↪die is", hasFour / num_sims)
```

```
The probability that 4 will appear at least once on four rolls of the die is
0.6695
```

[2]:
```
# 2. Simulation that top face will be 4 at least once on 20 rolls of die

num_sims = 10000

hasFour = 0
for sim in range(num_sims):
    for roll in range(20):
        if random.choice(range(1,5)) == 4:
            hasFour += 1
            break
print("The probability that 4 will appear at least once on twenty rolls of the␣
 ↪die is", hasFour / num_sims)
```

```
The probability that 4 will appear at least once on twenty rolls of the die is
0.998
```

[3]:
```
# 3. Simulation to be 90% confident a 4 is seen at least once

numRolls = 0
p = 0
while p < 0.9:
    numRolls += 1
    p = 1 - pow(0.75, numRolls)
print(numRolls, "rolls are required to be 90% confident a 4 is seen at least␣
 ↪once")
```

```
9 rolls are required to be 90% confident a 4 is seen at least once
```

[4]:
```
# 4. Function to find number of rolls necessary for a specific confidence␣
 ↪percentage

def rolls_required(confidence):
    numRolls = 0
    p = 0
    while p < confidence:
        numRolls += 1
        p = 1 - pow(0.75, numRolls)
    return numRolls

print(rolls_required(0.95), "rolls are required to be 95% confident a 4 is seen␣
 ↪at least once")
print(rolls_required(0.99), "rolls are required to be 99% confident a 4 is seen␣
 ↪at least once")
```

```
11 rolls are required to be 95% confident a 4 is seen at least once
```

```
17 rolls are required to be 99% confident a 4 is seen at least once
```

1.2 Problem 10

Create a simulation function where you will roll a fair 6-sided die twice. Use simulation to find out the probability of getting a 4,5, or 6 on the first toss and a 1,2,3 on the second toss.

[5]:
```
num_sims = 10000

eventCount = 0
for sim in range(num_sims):
    roll1 = random.choice(range(1,7))
    roll2 = random.choice(range(1,7))
    if roll1 > 3 and roll2 <= 3:
        eventCount += 1

print("The probability of this event occuring is", eventCount / num_sims)
```

```
The probability of this event occuring is 0.2513
```

1.3 Problem 11

Suppose that you have a bag with 3 coins. One of them is a fair coin, but the others are biased trick coins. When flipped, the three coins come up heads with probability $\frac{1}{2}$, $\frac{1}{3}$, and $\frac{1}{4}$, respectively.

Consider the experiment where you pick one coin at random and flip it three times. Let H_i be the event that the coin comes up heads on flip i. What is the probability of the outcome $H_1 \cap H_2 \cap \overline{H_3}$?

With small modification in your code, find out the probability of the outcome $H_1 \cap \overline{H_2} \cap \overline{H_3}$.

Use simulation to find out the probability.

[6]:
```
num_sims = 10000

eventCount1 = 0
eventCount2 = 0
event = ["H","H","T"]
for sim in range(num_sims):
    seq = []
    coin = random.choice(range(1,4))
    if coin == 1:
        seq = ["H", "T"]
    elif coin == 2:
        seq = ["H", "T", "T"]
    else:
        seq = ["H", "T", "T", "T"]
    flip1 = random.choice(seq)
    flip2 = random.choice(seq)
    flip3 = random.choice(seq)
    if flip1 == "H" and flip2 == "H" and flip3 != "H":
```

```
        eventCount1 += 1
    if flip1 == "H" and flip2 != "H" and flip3 != "H":
        eventCount2 += 1

print("The probability of  1 2 -H3 is", eventCount1 / num_sims)
print("The probability of  1 -H2 -H3 is", eventCount2 / num_sims)
```

```
The probability of  1 2 -H3 is 0.0802
The probability of  1 -H2 -H3 is 0.1406
```

2 Submit Your Solutions

Confirm that you've successfully completed the assignment.

Along with the Notebook, include a PDF of the notebook with your solutions.

`add` and `commit` the final version of your work, and `push` your PDF file to your GitHub repository.

Submit the URL of your GitHub Repository as your assignment submission on Canvas.