20/20

Statistics I — Quiz 1 (2024-09-09)

Name: **Roll number:**

A group of 20-year old students are all given 100 rupees. They have the option of investing the money for ten years in two different schemes.

In scheme A, they get a fixed interest rate of 25% every year, so that they have 125 rupees at the end of year one, 156.25 rupees at the end of year two, and so on, to have 931.32 rupees at the end of ten years.

In scheme B, they have to invest all their money in a high-risk high-reward game, where at the end of every year, their money is either doubled (with probability 0.5) or becomes half (with probability half). In other words, if they start with 100 rupees at the beginning of a year, they either have 200 rupees or 50 rupees (with equal probability) to invest at the beginning of the next year.

Answer the following questions in the question paper itself:

1. In a *large* group of students who choose scheme B, what is the average amount (total amount divided by number of students) each student will have after ten years?
   (a) Make a guess (before doing any computations): 500
   (b) Simulate scheme B 10000 times and report the observed mean: ≈ 953.157
2. What is the probability that someone choosing scheme B will have at least 100 rupees at the end of 10 years?
   (a) Guess (before doing any computations): 0.5
   (b) Answer (after suitable computations): 0.6236
3. What is the probability that someone choosing scheme B will have at least 931.32 rupees at the end of 10 years?
   (a) Guess (before doing any computations): ~~0.5~~ 0.2
   (b) Answer (after suitable computations): 0.17216
4. If 10000 students all choose scheme B, then at the end of 10 years, roughly what proportion of the total wealth will 10% of the richest students have?
   Your answer: ≈ 74%.

On the other side of this page, *very* briefly outline the approach you have taken to compute your answers.

To solve this following problem, we first define a function, which calculates the return for 10 years for B. What is does it, generates a number [0,1] and doubles if the number less than 0.5 or ~~0~~ halves it if the number is more than 0.5. This ~~coding~~ generating a random number will maintain the 50-50 minor loss factor in scheme B. After that we replicate it for 10 000 times and get the results.

1) Ans:- Here to solve the and simulate 10k times, we use the following R code.

Rscript:

```
# It's a function to calculate the increase by creating a random number and taking help of the randomness to solve the problem.
# The function is defined as following:—
simulate_student <- function() {
  money <- 100
  for (year in 1:years) {
    if (runif(1) < 0.5) {
      money <- money * 2 # if generated number is less than 0.5,
    } else {                # double the money
      money <- money / 2 # else reduce it by 2.
    }
  }
  return (money)
}
# Note num_students = 10000 and years = 100.
outcomes <- replicate (num_students, simulate_student()) # replication of the event for 10k times
avg <- mean(outcomes) # → Has the output average.
```

Function we define

2) Ans It's the continuous extension of Q1.

We will calculate the outcomes using our specially design function and then calculate the probability

probability <- sum(results >= 100) / 10000

Sum will count all the elements in results which are over >= 100.

Note, sum is generally used for addition, but by appling it to boolen data.

results >= 100 will change results = [10, 1000 ----] give after results >= 100, [0, 1 ....] it will a boolen which will say which ones are more than 100 or not and adding them will give me the sum as TRUE = 1 & FALSE = 0

3) Ans:- Same way, we will change the results >=100 to results >= 931.32 and compute the code. and get the probability as 0.17216

4) Ans:- Here again, we calculate the outcomes and then we will sort them in decreasing order, then get the first 10% elements, sum all their wealth and the calculate the percentage of wealth they hold.

R code →

```
sorted <- sort(outcomes, decreasing = T
topgroup <- sum(sorted[1:1000])
percen <- topgroup/sum(outcomes
cat(percen)
```

percen ≈ 74.15528%.