Monster Gems Puzzle Solution

May 21, 2016

1 Solution to the 538 "Riddler," Week of May 20, 2016

1.1 William Gilpin

For $N+1$ consecutive monster slayings require that only two gems are observed for the first N slayings. The probability of collecting n of a type of gem of type "I" in a set of N slayings is,

$$p_{I;N} = \binom{N}{n} p_I^n p_J^{N-n}$$

where $I, J \in \{A, B, C\}, I \neq J$. We note that this formula is the celebrated binomial distribution, and $\binom{N}{n}$ is "N choose n" or the binomial coefficient. The total probability of obtaining at least one gem of type I and one gem of type J in a set of N is given by summation of these coefficients,

$$p_{I,J>0;N} = \sum_{n=1}^{N-1} \binom{N}{n} p_I^n p_J^{N-n}$$

where p_I, p_J are the probabilities of the monster dropping a gem of type I or J, and the bounds of the summation $1, N-1$ enforce that at least one gem each of type I and K is collected. The total number of monsters slayed.

In order to find the expected total number of monsters that need to be slayed $\mathcal{N}_K$ before obtaining a gem of type $K \neq I \neq J$, we take another summation over sequences of monster slayings of different lengths that terminate in the first collection of gem K,

$$\mathcal{N}_K = \sum_{N=2}^{\infty} p_{I,J>0;N}(N+1) = \sum_{N=2}^{\infty} (N+1) \sum_{n=1}^{N-1} \binom{N}{n} p_I^n p_J^{N-n}$$

where the bounds of the summation are determined by by the fact that the minimum number of slayings is 3 monsters, and the $(N+1)$ term comes from requiring that we slay N monsters and obtain only I, J, but we then slay monster $N+1$ and obtain gem K.

Finally, in order to obtain the average number of slain monsters across all values of K,

$$\mathcal{N} = \sum_{K=1}^{3} \mathcal{N}_K p_K = \sum_{K=1}^{3} p_K \sum_{N=2}^{\infty} (N+1) \sum_{n=1}^{N-1} \binom{N}{n} p_I^n p_J^{N-n}$$

At this point, we insert the known values of $p_1 = 3/6, p_2 = 2/6, p_3 = 1/6$ and solve this expression,

$$\mathcal{N} = 7.3 \approx 7$$

Because 7.3 is the expected number of slayings, the expected number of the most common gem is

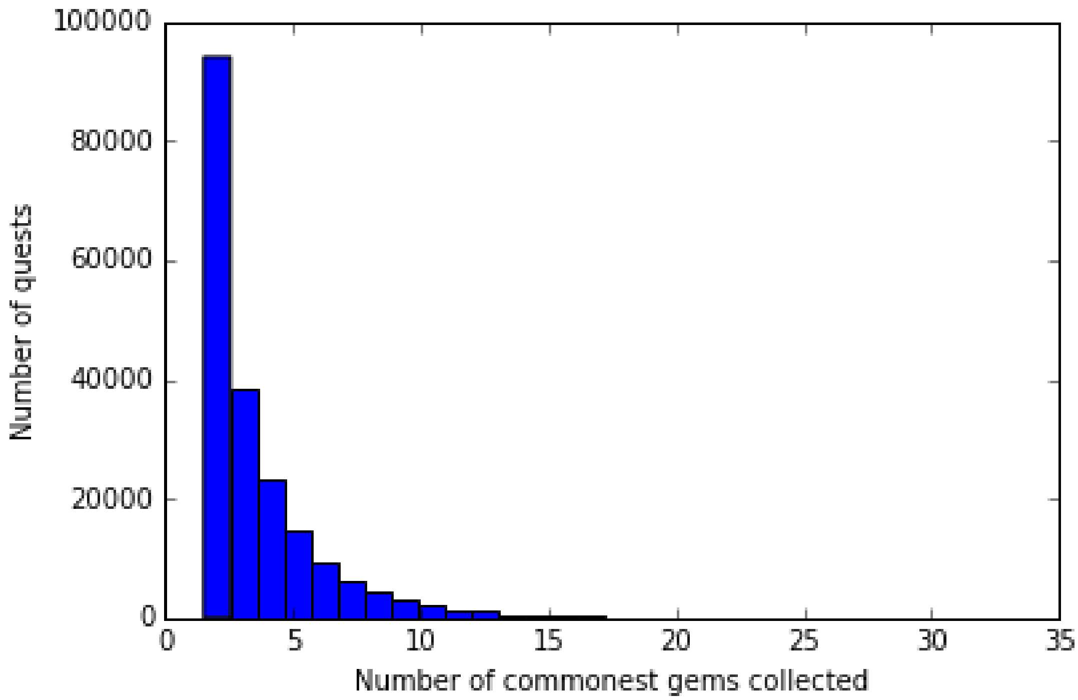
$$G_1 = p_1\mathcal{N} = 7.3(3/6) = 3.65 \approx 4\ gems$$

Below, we confirm this result with a numerical simulation.

```
In [3]: %matplotlib inline
        from matplotlib.pyplot import *
        from numpy import *
        from random import choice

        GEM_TYPES = [1,2,3]
        GEM_FREQS = [1,2,2,3,3,3]
        N_TRIALS = 200000

        all_comm_counts = []
        all_run_lengths = []

        for ii in range(N_TRIALS):

            gems_collected = []

            while not all([item in gems_collected for item in GEM_TYPES]):
                gems_collected.append(choice(GEM_FREQS))

            all_comm_counts.append(gems_collected.count(3))
            all_run_lengths.append(len(gems_collected))


        print("Predicted: "+ str(3.65))
        print("Observed: "+ str(mean(all_comm_counts)))
        hist(array(all_run_lengths)*.5,30);
        ylabel("Number of quests")
        xlabel("Number of commonest gems collected")
```

```
Predicted: 3.65
Observed: 3.65431
```

```
Out[3]: <matplotlib.text.Text at 0x10dd300f0>
```


100000
80000
60000
40000
20000
0
0
5
10
15
20
25
30
35
Number of quests
Number of commonest gems collected

```
In [ ]:
```