FiveThirtyEight's March 11, 2022 Riddler

Emma Knight

March 13, 2022

This week's riddler, from Ed Carl, is about a game with dice:

Question 1. *We're playing a game where you have to pick four whole numbers. Then I will roll four fair dice. If any two of the dice add up to any one of the numbers you picked, then you win! Otherwise, you lose.*

To maximize your chances of winning, which four numbers should you pick? And what are your chances of winning?

There aren't *that* many options for what to pick, so it's not too hard to hit this with a computer. Since I set this up with some generality, here is a pair of tables that show the best guesses and the probability of winning respectively:

Guesses	1	2	3	4	5	6
2	$\{7\}$	$\{6,7\}$	$\{6,7,8\}$	$\{5,6,7,8\}$	$\{5,6,7,8,9\}$	$\{4,5,6,7,8,9\}$
3	$\{7\}$	$\{6,7\}$	$\{6,7,8\}$	$\{4,6,8,10\}$	$\{2,4,6,8,10\}$	$\{2,4,6,8,10,12\}$
4	$\{7\}$	$\{6,7\}$	$\{6,7,8\}$	$\{4,6,8,10\}$	$\{2,4,6,8,10\}$	$\{2,4,6,8,10,12\}$
5	$\{7\}$	$\{6,7\}$	$\{6,7,8\}$	$\{4,6,8,10\}$	$\{2,4,6,8,10\}$	$\{2,4,6,8,10,12\}$

Probabilities	1	2	3	4	5	6
2	1/6	11/36	4/9	5/9	2/3	3/4
3	5/12	139/216	85/108	49/54	103/108	1
4	139/216	361/432	149/162	79/81	1283/1296	1
5	115/144	7211/7776	419/432	1931/1944	485/486	1

Before talking about the tables, there is a minor point: when the best guesses aren't symmetric about 7, then the set where you replace every guess with 14 minus that guess is also an optimal guess. However, there are never any other optimal guesses.

The optimal strategy is as follows: if you have at least 4 guesses and I roll at least 3 dice, then you should guess in order $6, 8, 4, 10, 2, 12, e, \Gamma(3+2i), \sin(2-5\sqrt{3}i), \ldots$. Otherwise, you should guess in order $7, 6, 8, 5, 9, 4, 10, 3, 11, 2, 12, \pi, \tan(3+4i), \log(2-5\sqrt{3}i), \ldots$. The centrally located numbers make sense; the key observation about 3 dice is that, once I roll 3 dice, I'm guaranteed to have a pair of dice sum to an even number. This is why you get more utility out of guessing $\{4, 6, 8, 10\}$ than $\{5, 6, 7, 8\}$ when I roll 3 or more dice.

Finally, here is the code[1]:

```
import itertools
import math

##This funtion takes in a guess and a set of rolls,
##and outputs whether that guess wins for that set
##of rolls.

def win(g, r):
    for a in g:
        for b in itertools.combinations(r, 2):
            if b[0] + b[1] == a:
                return(True)
    return(False)

##This function takes in a guess and a set of sets
##of rolls, and determines how many of the rolls
##the guess wins for.

def score(g, rs):
    score = 0
    for r in rs:
        if win(g, r):
            score += 1
    return(score)

##gu is the number of guesses, ro is the number of rolls,
##and si is the number of sides of the die.

gu = 4
ro = 4
si = 6

##This part loops over all the (sensible) guesses and finds
##the best guess(es) and the worst guess(es), and prints out
##both how good they are and what they are.  In addition, it
##finds out how good a random guess is on average.

guesses = itertools.combinations(range(2, 2*si+1), gu)

tot = 0
best = 0
```

[1]There are some other things I looked at in the code but didn't discuss above; you're free to play around with it.

```
bests = []
worst = si**ro
worsts = []

for guess in guesses:
    rolls = itertools.product(range(1, si+1), repeat=ro)
    s = score(guess, rolls)
    tot += s
    if s == best:
        bests.append(guess)
    if s > best:
        best = s
        bests = [guess]
    if s == worst:
        worsts.append(guess)
    if s < worst:
        worst = s
        worsts = [guess]

print(best, best/(si**ro), bests)
print(worst, worst/(si**ro), worsts)
print(tot/(math.comb(2*si-1, gu)*(si**ro)))
```