The ‘Russian Roulette’ problem: a probabilistic variant of the Josephus Problem

Joshua Brulé

2013 May 2

1 Introduction

The classic Josephus problem describes n people arranged in a circle, such that every m^{th} person is removed going around the circle until only one remains. The problem is to find the position $J_{n,m} \in \{1, 2, \ldots, n\}$ at which one should stand in order to be the survivor. [1]

The ‘Russian Roulette’ problem considers n people arranged in a circle, with a (biased) coin passed from person to person. Upon receiving the coin, the k^{th} player flips the coin and is eliminated if and only if it comes up heads. The coin is then passed to the $(k + 1)^{th}$ person continuing until only one person - the survivor - remains. Note that as long as the probability of elimination is non-zero, the game will almost surely terminate.

To give the problem richer structure, we allow the probability of elimination to depend on the number of players still in the game (i.e. $p_k \in (0, 1]$ is the probability of elimination with k players remaining in the game).

1.1 Problem Definition

Given n players and a list of elimination probabilities $P = \{p_n, p_{n-1}, \ldots, p_2\}$, find the probability $R_{n,k}(P)$ that the kth player will be the survivor?

2 Solution as a Recurrence Relation

Let the list of elimination probabilities be $P = \{p_n, p_{n-1}, \ldots, p_2\}$ and for convenience, let $q_n = 1 - p_n$.

Let $L_{n,k}(P)$ be the probability that the k^{th} player in an n player game *will not* be the survivor. (For conciseness, the list P is implied in the following calculations.)

Clearly, $L_{1,1} = 0$ (the case of only one player, who wins by default). In addition, $L_{i,0} = 1$ for all $i \geq 1$ ('Player 0' represents someone who has already been eliminated, thus the probability of them not being the survivor is 1) Then,

$$L_{n,k} = \frac{p_n}{1 - q_n^n} \sum_{i=1}^{n} q_n^{i-1} L_{(n-1),(k-i) \bmod n}$$

or equivalently,

$$L_{n,k} = \frac{p_n}{1 - (1 - p_n)^n} \sum_{i=1}^{n} (1 - p_n)^{i-1} L_{(n-1),(k-i) \bmod n}$$

Proof. Suppose that the i^{th} player is eliminated in an n player game:
For $k > i$ the k^{th} player becomes the $k - i \equiv (k - i) \bmod n$ player in a game with $n - 1$ players and elimination probability p_{n-1}
For $k = i$, the k^{th} player is eliminated, and has probability 1 of *not* surviving the rest of the rounds, regardless of their outcome. By defining $L_{n,0} = 1$, we can consider the eliminated player as the '0^{th}' player in an $n - 1$ player game.
For $k < i$, the k^{th} player becomes the $n - i + k \equiv (k - i) \bmod n$ in a game with $n - 1$ players and elimination probability p_{n-1}
Finally in the case of n consecutive non-eliminations, the probability of each players survival is unchanged (we 'made a complete circle' where no one was eliminated). Thus:

$$
\begin{aligned}
L_{n,k} &= p_n L_{n-1,k-1 \bmod n} + q_n(p_n L_{n-1,k-2 \bmod n} + q_n(\dots q_n(p_n L_{n-1,k-(n-1) \bmod n} + q_n L_{n,k})\dots)) \\
&= p_n(q_n^0 L_{n-1,k-1 \bmod n} + q_n^1 L_{n-1,k-2 \bmod n} + \dots + q_n^{n-1} L_{n-1,k-(n-1) \bmod n}) + q_n^n L_{n,k} \\
&= \frac{1}{1-q_n^n} p_n(q_n^0 L_{n-1,k-1 \bmod n} + q_n^1 L_{n-1,k-2 \bmod n} + \dots + q_n^{n-1} L_{n-1,k-(n-1) \bmod n}) \\
&= \frac{p_n}{1-q_n^n} \sum_{i=1}^{n} q_n^{i-1} L_{n-1,(k-i) \bmod n}
\end{aligned}
$$

□

Since the game almost surely terminates, the probability of the k^{th} player winning in an n player game is simply $1 - L_{n,k}$

Using dynamic programming, $L_{n,k}$ for all $1 \leq k \leq n$ can be computed in $O(n^3)$ time. An example implementation in python:

```
#create diagonal half-matrix for memoization
L = [[0 for k in range(0, n + 1)] for n in range(0, max_players + 1)]

#base cases
for i in range(1, max_players + 1):
    L[i][0] = 1

L[1][1] = 0
L[0][0] = None

#compute the probability of losing
for n in range(2, max_players + 1):
    for k in range(1, n + 1):
        sum = 0
        for i in range(1, n + 1):
            sum = sum + (1 - prob[n])**(i-1) * L[n - 1][mod(k - i, n)]

        L[n][k] = prob[n]/(1 - (1 - prob[n])**n) * sum
```

Where prob[] is an array of the elimination probabilities (prob[i] $= p_i \in (0, 1]$ or a symbolic variable if an appropriate library is available. (The experiments described below were all conducted in Sage [2] which provides a symbolic computing environment.)

2.1 Bounds on the complexity of the solution

In the special case of $p_2 = p_3 = ... = p_n = p$, $L_{n,k}$ is the ratio of two polynomials, each of degree at most $\frac{n(n-1)}{2}$

Proof. Base case: $L_{n,1} = 1$ which is clearly the ratio of two polynomials each of degree at most $\frac{n(n-1)}{2} = 0$.

In general, $L_{n,k} = \frac{p}{1-(1-p)^n} \sum_{i=1}^{n} (1-p)^{i-1} L_{n-1,(k-i) \bmod n}$. Note that $\frac{p}{1-(1-p)^n} = \frac{1}{\mathrm{poly}(n-1)}$ (where poly$(n-1)$ is a polynomial of degree $n-1$). The summation is the sum of polynomials of degree at most $(n-1)$ multiplied by $L_{n-1,j}$ for some j, which is - by the inductive hypothesis - the ratio of two polynomials each of degree at most $(n-2)$. The multiplication of two polynomials of degrees n and m, yields a new polynomial of degree n + m. Thus, $L_{n,k}$ is of degree at most $\sum_{i=0}^{n-1} i = (n-1)n/2$. □

3 Specific solutions and experimental verification

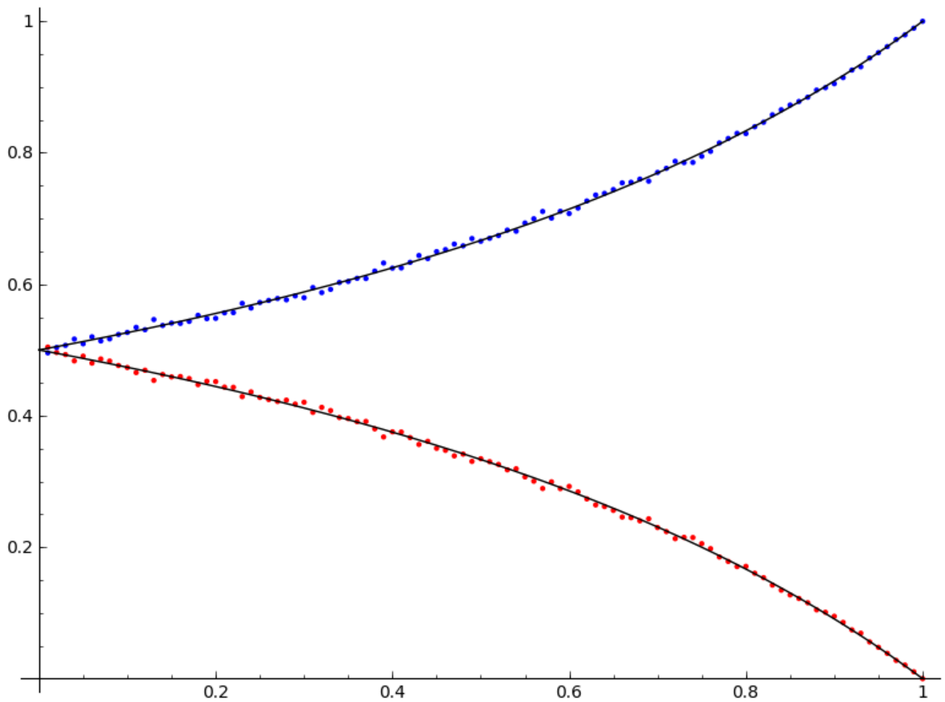
3.1 $n = 2$

The case $n = 2$ has a simple solution that matches well with intuition:

$$L_{2,1} = 1/(2 - p_2)$$
$$L_{2,2} = (1 - p_2)/(2 - p_2)$$

As the probability of elimination tends to 1, the probability that player 1 will not be the survivor also tends to 1. With a probability of elimination near 0, the probability of each player being the survivor is very nearly 1/2.

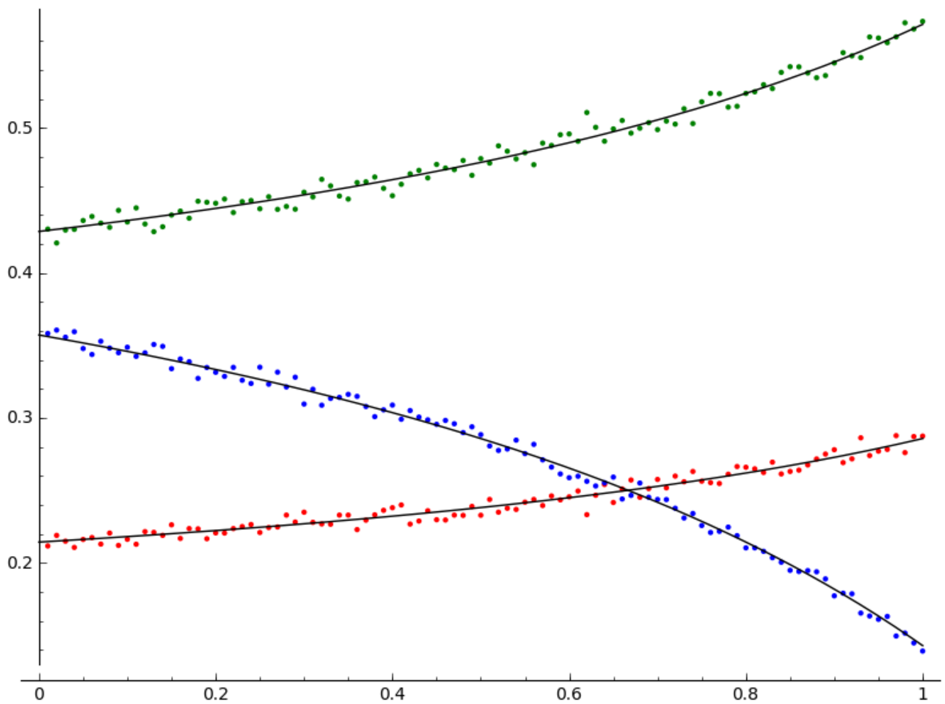
To experimentally verify, a simulation of the game was run and the results plotted against a graph of the probability of surviving vs. probability of elimination.



(Each point represents the percent games lost out of 10000 trials.)

3.2 n=3

With $n \geq 3$ more interesting results are possible - a demonstrative example is with p_3 fixed at 1/2 and the probability of survival for each player (red=1, blue=2, green=3) plotted against p_2 below.

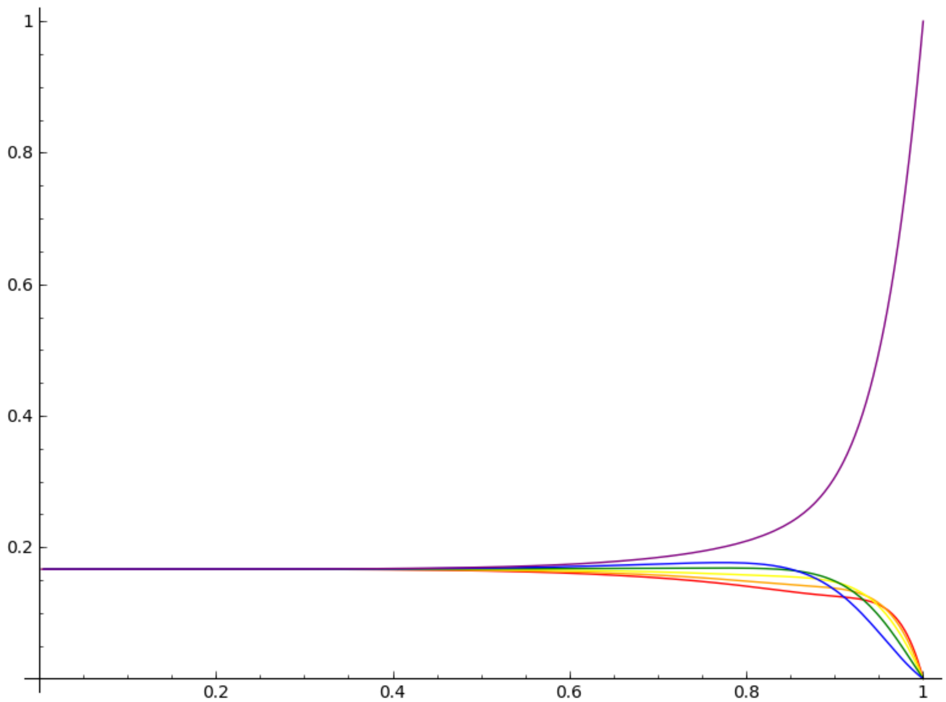


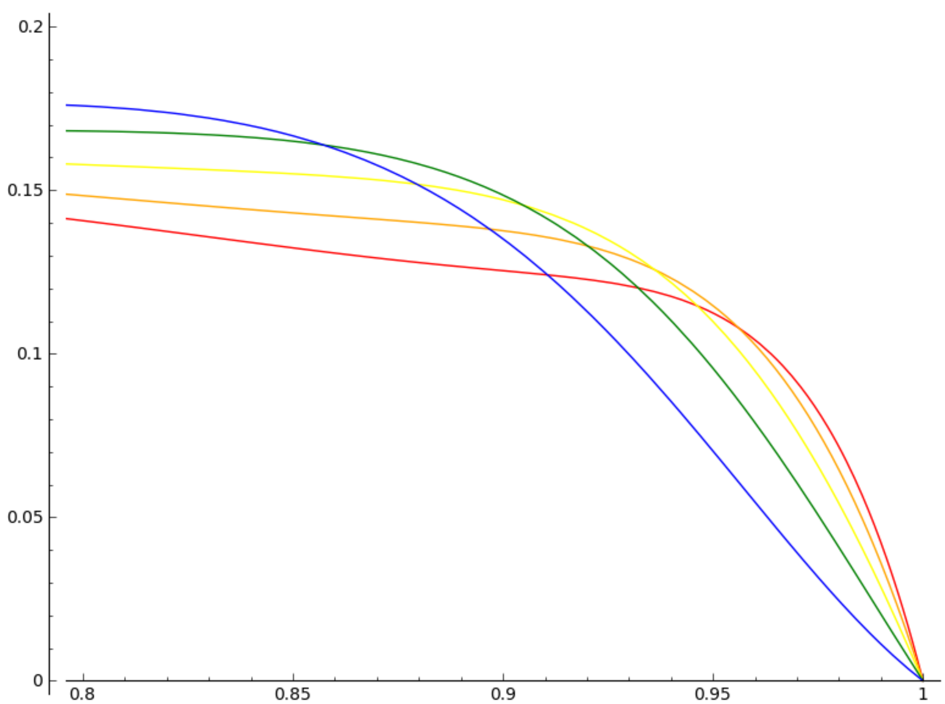
(Each point represents 10000 simulations.)

As can be seen, for certain elimination probabilities, player 1 actually has a better chance of winning then player 2! However, it appears (and indeed, can be proven) that player 3 always has the best probability of winning, regardless of which elimination probabilities are used.)

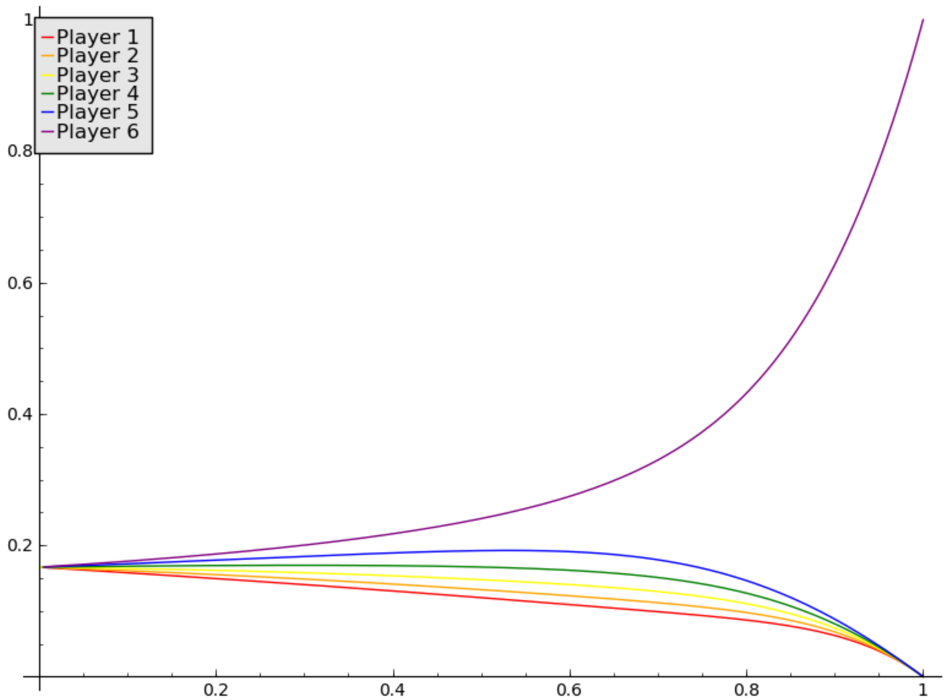
3.3 $n > 3$

Setting $p_{n-1} = p_n^2$, $p_{n-2} = p_n^3 \ldots$ and plotting probability of survival (red=1, orange=2, yellow=3, …) against p_n gives the following results:


1
0.8
0.6
0.4
0.2
0.2
0.4
0.6
0.8
1


0.2
0.15
0.1
0.05
0
0.8
0.85
0.9
0.95
1

Setting $p_n = p_{n-1} = \ldots = p_2$ and plotting probability of survival against p_n gives the following results:



(Similar behavior is observed for $n < 6$).

These results (and additional experimentation with various lists of elimination probabilities) are very suggestive and motivate several conjectures.

3.4 Conjectures

$L_{n,n} \leq L_{n,k}$ for all k, n, and p_i's.

Given $p_n = p_{n-1} = \ldots = p_2$, $L_{n,n} \leq L_{n,n-1} \leq \ldots \leq L_{n,1}$ for all n and p_n.

For all permutations $k_1, k_2, \ldots, k_{n-1}$ of $1, 2, \ldots, n-1$ there exists p_i's such that $L_{n,k_1} < L_{n,k_2} < \cdots < L_{n,k_{n-1}}$ In other words, there always exists a list of elimination probabilities such that sorting the players by $L_{n,k}$ yields every possible permutation of players - excluding player n who always has the lowest probability of losing.

With computer assistance, this last conjecture was directly proven up to $n = 6$ by grid searching the space of elimination probability lists. Every permutation of players' probability of survival was proven to be possible by finding a specific list of elimination probabilities that yields that permutation.

Unfortunately, naive grid search is exponential in n, and is computationally intractable for $n \geq 7$. A formal proof for all n remains elusive.

4 Acknowledgment

I would like to thank Dr. William Gasarch for his support and guidance while conducting this research.

References

[1] Weisstein, Eric W. "Josephus Problem." *MathWorld* http://mathworld.wolfram.com/JosephusProblem.html

[2] William A. Stein et al. Sage Mathematics Software http://www.sagemath.org