

Dice Roll Strategy

Matthew Seguin

Introduction

Imagine you are trapped in a room with the only way to escape being to correctly predict the roll of an n -sided dice. You do not know if this is a fair dice or not, but you are allowed to roll the dice without guessing as many times as you would like.

What is your best strategy for getting out?

Should you hope that the dice is fair?

How many sides should you hope for the dice to have?

How many times should you roll the dice?

The answers to these questions are fairly intuitive, but let's look at the math behind this problem and some simulations of different strategies.

In this document I will show several different strategies for predicting the next roll of an n -sided dice and will analyze which is the most effective. The dice is not assumed to be fair, so some strategies will perform better than others. For each strategy I take a large number of different dice probabilities in order to see how each performs at different levels.

As a side note this can also be used to model a coin flip if you take $n = 2$.

Theory

At some point you have to guess and you will want to maximize the chance that your guess matches the actual roll of the dice.

Assume the dice has some true probability vector $p = (p_1, p_2, p_3, \dots, p_n)$ where p_k is the probability of rolling a k . Further assume that each roll is independent from previous rolls. Then let $X = (k_1, k_2, k_3, \dots, k_n)$ be the respective counts for each side from rolling the dice m times.

Then X by definition follows a multinomial distribution with parameters m and p , that is $X \sim MN(m, p)$.

This means that the expected counts of each roll are $\mathbb{E}(X) = mp = m(p_1, p_2, p_3, \dots, p_n)$.

By the law of large numbers we expect for large m that X will be very close to $\mathbb{E}(X)$.

This means that for large m we expect $\frac{X}{m} = \frac{1}{m}(k_1, k_2, k_3, \dots, k_n)$ to be very close to $\frac{1}{m}\mathbb{E}(X) = (p_1, p_2, p_3, \dots, p_n)$.

So we have found one way to come up with a guess. If you roll the dice many times and count the number of rolls of each side and divide by the total number of rolls you have approximated the probability of each side.

You will then want to make your guess the most likely side to roll, hence you will want to choose the side that has come up the most so far.

This is a rather hand-wavy derivation though it has similar logic to what I shall show next.

A more formal way to do this is to introduce the method of maximum likelihood in order to estimate our probability vector.

This is where given a set of data from a known distribution you compute the parameters for that distribution that would give you the highest likelihood of getting what you observed.

The maximum likelihood estimator has very nice properties in that it is asymptotically unbiased, consistent, and for large sample size achieves the Cramer-Rao lower bound for the variance of an estimator.

Simply put this means that the maximum likelihood estimator will be very close to the true value for large sample size.

Again we know $X \sim MN(m, p)$ so:

$$P(X = (k_1, k_2, k_3, \dots, k_n)) = \frac{m!}{k_1!k_2!k_3!\dots k_n!} \prod_{j=1}^n p_j^{k_j} = L(p)$$

When $\sum k_j = m$ and is otherwise 0. Furthermore we must have that $\sum p_j = 1$.

Proof:

The probability of observing something happen must be 1 so we must have that $\sum p_j = 1$.

There are m observations so $\sum k_j = k_1 + k_2 + \dots + k_n \neq m$ can't happen and $P(X = (k_1, k_2, \dots, k_n)) = 0$ if $\sum k_j \neq m$.

Now assume $\sum k_j = k_1 + k_2 + \dots + k_n = m$ and $P(x_i = k) = p_k$ with each observation independent of previous ones.

Let the ordered (i.e. starting at the first observation and ending at the last) sample be $X = (x_1, x_2, x_3, \dots, x_m)$.

Each of these observations must fall in one of the groups $1, 2, 3, \dots, n$ and $P(x_i = k) = p_k$ for $k \in \{1, 2, 3, \dots\}$.

Therefore if we have k_1 observations of 1, k_2 observations of 2, ..., and k_n observations of n then the probability of getting this exact ordered sample is $(p_1)^{k_1}(p_2)^{k_2}\dots(p_n)^{k_n}$ because every subsequent observation is independent of previous ones.

This is true for any sample with k_1 observations of 1, k_2 observations of 2, ..., and k_n observations of n so each such sample has the same probability above.

Now we know we have m observations total, so there are $\binom{m}{k_1} = \frac{m!}{k_1!(m-k_1)!}$ ways to choose when we observe the 1's.

After that we have $m - k_1$ observations left, so there are $\binom{m-k_1}{k_2} = \frac{(m-k_1)!}{k_2!(m-k_1-k_2)!}$ ways to choose when we observe the 2's.

Following this process then multiplying all the terms to get the total number of ways to observe a sample with k_1 observations of 1, k_2 observations of 2, ..., and k_n observations of n , we get the total possible number of such samples is:

$$\left(\frac{m!}{k_1!(m-k_1)!}\right)\left(\frac{(m-k_1)!}{k_2!(m-k_1-k_2)!}\right)\dots\left(\frac{(m-k_1-k_2-\dots-k_{n-1})!}{k_n!(m-k_1-k_2-\dots-k_{n-1}-k_n)!}\right) = \frac{m!}{k_1!k_2!\dots k_n!}$$

Since there are $\frac{m!}{k_1!k_2!\dots k_n!}$ total ways of observing the sample k_1 observations of 1, k_2 observations of 2, ..., and k_n observations of n each way with probability $(p_1)^{k_1}(p_2)^{k_2}\dots(p_n)^{k_n}$ we get that:

$$P(X = (k_1, k_2, k_3, \dots, k_n)) = \frac{m!}{k_1!k_2!k_3!\dots k_n!} p_1^{k_1} p_2^{k_2} \dots p_n^{k_n} = \frac{m!}{k_1!k_2!k_3!\dots k_n!} \prod_{j=1}^n p_j^{k_j} = L(p)$$

Hence proved \square

We want to maximize this probability so that our observed the sample is the most likely one to have. This is easier if we consider the log of this probability as maximizing one amounts to maximizing the other and the product will become a sum.

$$\log(L(p)) = \log\left(\frac{m!}{k_1!k_2!k_3!\dots k_n!} \prod_{j=1}^n p_j^{k_j}\right) = \log\left(\frac{m!}{k_1!k_2!k_3!\dots k_n!}\right) + \sum_{j=1}^n k_j \log(p_j) = l(p)$$

So we are maximizing $l(p)$ subject to the constraint $C(p) = \sum_{j=1}^n p_j = 1$.

Notice that:

$$\frac{\partial}{\partial p_t} l(p) = \frac{\partial}{\partial p_t} \log\left(\frac{m!}{k_1!k_2!k_3!\dots k_n!}\right) + \frac{\partial}{\partial p_t} \sum_{j=1}^n k_j \log(p_j) = 0 + \sum_{j=1}^n \frac{\partial}{\partial p_t} k_j \log(p_j) = 0 + \dots + 0 + \frac{k_t}{p_t} + 0 + \dots + 0 = \frac{k_t}{p_t}$$

Also we know that:

$$\frac{\partial}{\partial p_t} C(p) = \frac{\partial}{\partial p_t} \sum_{j=1}^n p_j = \sum_{j=1}^n \frac{\partial}{\partial p_t} p_j = 0 + \dots + 0 + 1 + 0 + \dots + 0 = 1$$

So using Lagrange multipliers we would have $\nabla l = \lambda \nabla C$ which is expressed below:

$$\left(\frac{\partial}{\partial p_1} l(p), \frac{\partial}{\partial p_2} l(p), \frac{\partial}{\partial p_3} l(p), \dots, \frac{\partial}{\partial p_n} l(p)\right) = \lambda \left(\frac{\partial}{\partial p_1} C(p), \frac{\partial}{\partial p_2} C(p), \frac{\partial}{\partial p_3} C(p), \dots, \frac{\partial}{\partial p_n} C(p)\right)$$

$$\left(\frac{k_1}{p_1}, \frac{k_2}{p_2}, \frac{k_3}{p_3}, \dots, \frac{k_n}{p_n}\right) = \lambda(1, 1, 1, \dots, 1)$$

This implies that for each $j \in \{1, 2, 3, \dots, n\}$ we have $\lambda p_j = k_j$.

Adding up all these equations we get

$$\lambda \sum_{j=1}^n p_j = \lambda = \sum_{j=1}^n k_j = m$$

So for each $j \in \{1, 2, 3, \dots, n\}$ we have that $\hat{p}_j = \frac{k_j}{\lambda} = \frac{k_j}{m}$ is the best estimate for the probability of rolling a j according to such an observed sample.

Hence our best estimate probability vector is $\hat{p} = (\hat{p}_1, \hat{p}_2, \hat{p}_3, \dots, \hat{p}_n) = \frac{1}{m}(k_1, k_2, k_3, \dots, k_n)$.

Since we want to choose the one with the highest estimated probability we want to choose the one that has been observed the most, that is let:

$$\text{Guess} = \{j : k_j = \max\{k_1, k_2, k_3, \dots, k_n\}\}$$

Now that we have our ideal strategy, pick which side has been rolled the most lets examine those questions from earlier.

- Should you want a fair dice or a biased dice?

Normally playing with a biased dice is seen as a disadvantage, this is because in games normally you expect a dice to be fair and if you are playing against an opponent who knows the dice is unfair you are at an information deficit and hence a disadvantage. However, here you are given the opportunity to examine the dice. If the dice truly is fair and you roll it a large number of times, coming to the conclusion that it is fair, then you have a $\frac{1}{n}$ chance of guessing correctly. However, if the dice is biased you will notice so as the number of rolls grows larger, and if rolled enough you will choose the side with the highest probability correctly, which will be the side that has appeared the most. Since not all the sides are equally likely at least one must have more than a $\frac{1}{n}$ chance of turning up (I shall provide a proof of this below). So you will then have more than a $\frac{1}{n}$ chance of guessing correctly. Therefore in this scenario you should hope that the dice is biased even though ultimately your fate is left to chance.

- How many sides should you hope the dice has?

There are two sides to this problem. If the dice is fair then your best chance is obviously if it only has 2 sides, then you have a $\frac{1}{2}$ chance of guessing correctly whereas if it has more than two you have a $\frac{1}{n} < \frac{1}{2}$ chance of guessing correctly. If the dice is not fair then you should also hope that it only has 2 sides though. This is because if it has 3 or more sides then a lack of probability for one side can be result in the other sides evenly sharing the excess probability and only slightly increasing the average probability of the other sides whereas with two sides there is only one side for that remaining probability to go. In general if one side has a negligible probability of appearing (that is it will practically never be rolled) and if the probability that would be for that side is evenly distributed to the other sides you are essentially rolling an $n - 1$ sided dice rather than an n sided one. This will be shown later in the simulation.

- How many times should you roll the dice?

The more you roll the dice the lower the variance of the total sum of rolls for each side gets, so you are decreasing the likelihood that you just got a very unlikely set of rolls each time you roll again. Still, you can't roll forever. The answer to this question also depends on how much risk you want to take. A good rule of thumb would be to do at least 100 rolls though as the variance starts to get very low at that point. Even then 100 rolls of a dice doesn't take long at all so as long as you have some method of recording rolls I would suggest rolling more.

Proof that at least one side must have probability greater than $\frac{1}{n}$ if not all are equal to $\frac{1}{n}$:

Let $p = (p_1, p_2, p_3, \dots, p_n)$. Then consider $p_1, p_2, p_3, \dots, p_{n-1}$.

If one of these is greater than $\frac{1}{n}$ then we are done. Otherwise $p_j \leq \frac{1}{n}$ for each $j \in \{1, 2, 3, \dots, n-1\}$.

Furthermore it must be that at least one of $p_j < \frac{1}{n}$ for $j \in \{1, 2, 3, \dots, n-1\}$.

Otherwise $\sum_{j=1}^{n-1} p_j = \frac{n-1}{n}$ and hence $p_n = 1 - \sum_{j=1}^{n-1} p_j = 1 - \frac{n-1}{n} = \frac{1}{n}$, a contradiction since not all p_j are equal to $\frac{1}{n}$.

So we have that at least one of the $p_j < \frac{1}{n}$ for $j \in \{1, 2, 3, \dots, n-1\}$.

Hence $\sum_{j=1}^{n-1} p_j < \frac{n-1}{n}$ and $p_n = 1 - \sum_{j=1}^{n-1} p_j > 1 - \frac{n-1}{n} = \frac{1}{n} \square$

Simulation

Strategy 1 is where you switch your choice every time. This is not a very good strategy compared to the others, it relies purely on luck.

Strategy 2 is where you switch to whatever the last roll was. This is a much better strategy than the first one clearly as we don't exactly know whether the dice is fair or not so this will perform well when it isn't.

Strategy 3 is where you pick what has been rolled the most so far, and if it is tied then you switch from your previous choice.

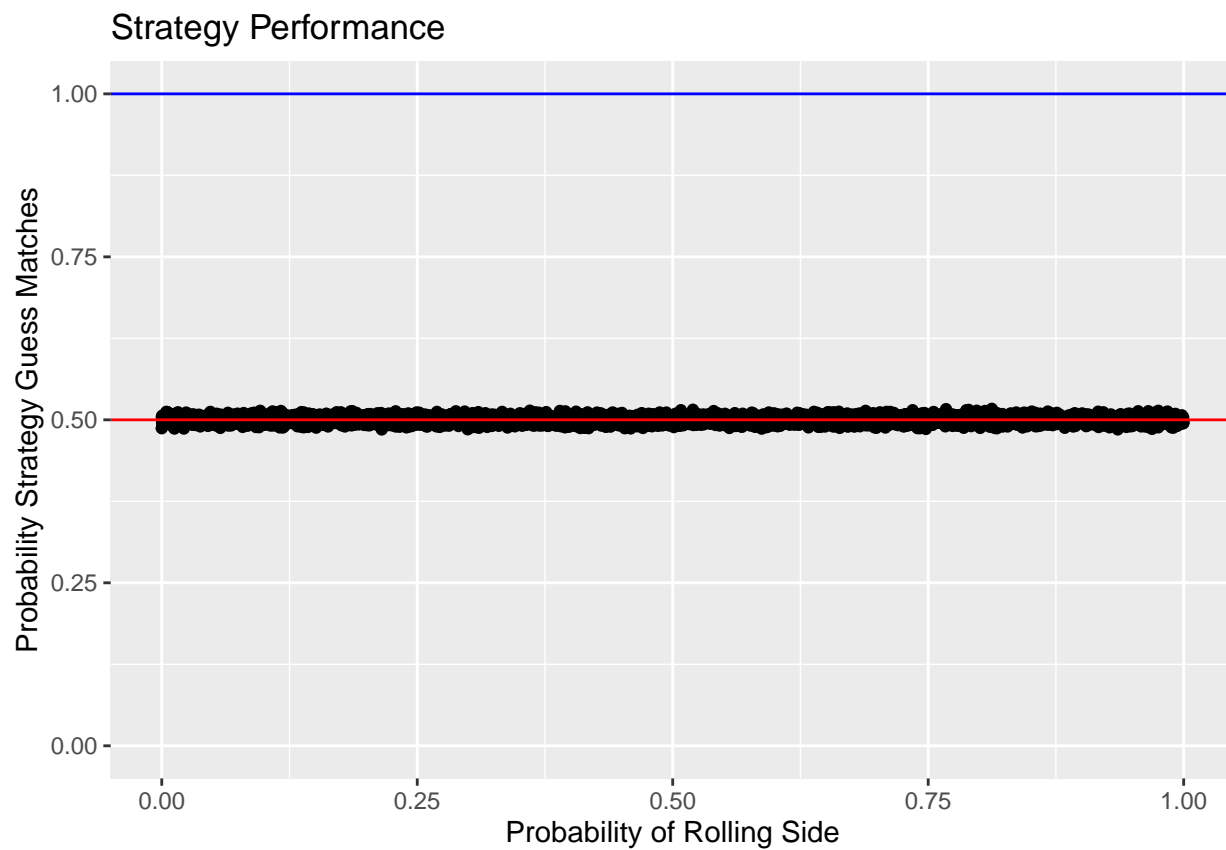
Strategy 4 is where you pick what has been rolled the most so far, and if it is tied then you don't switch from your previous choice.

When $N > 2$ it is harder to visualize the graphs, so for the sake of simplicity each graph will have the x axis as the probability of rolling a side (not any specific side) for example rolling a 1, but without loss of generality it is the same as for any other side. So it will simply be the probability of rolling a side, and the y axis will be the probability the strategy guess matches the roll. The blue horizontal line will be at $\frac{1}{N-1}$ and the red horizontal line will be at $\frac{1}{N}$. So the blue line represents the chance of randomly guessing the correct side for an $N - 1$ sided dice, and the red line will represent the chance of randomly guessing the correct side for an N sided dice.

Results for $n = 2$:

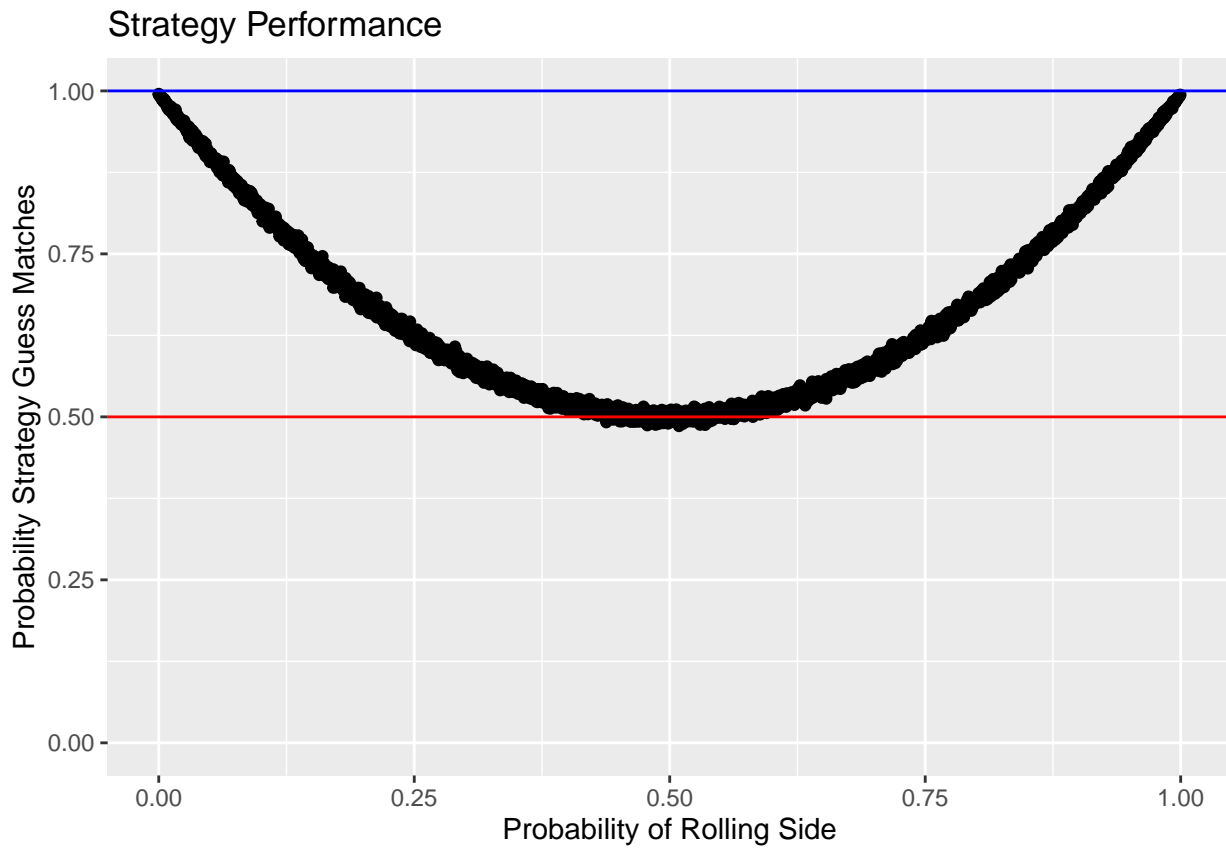
Strategy 1

```
grapher(N = 2, strat = strategy_1)
```



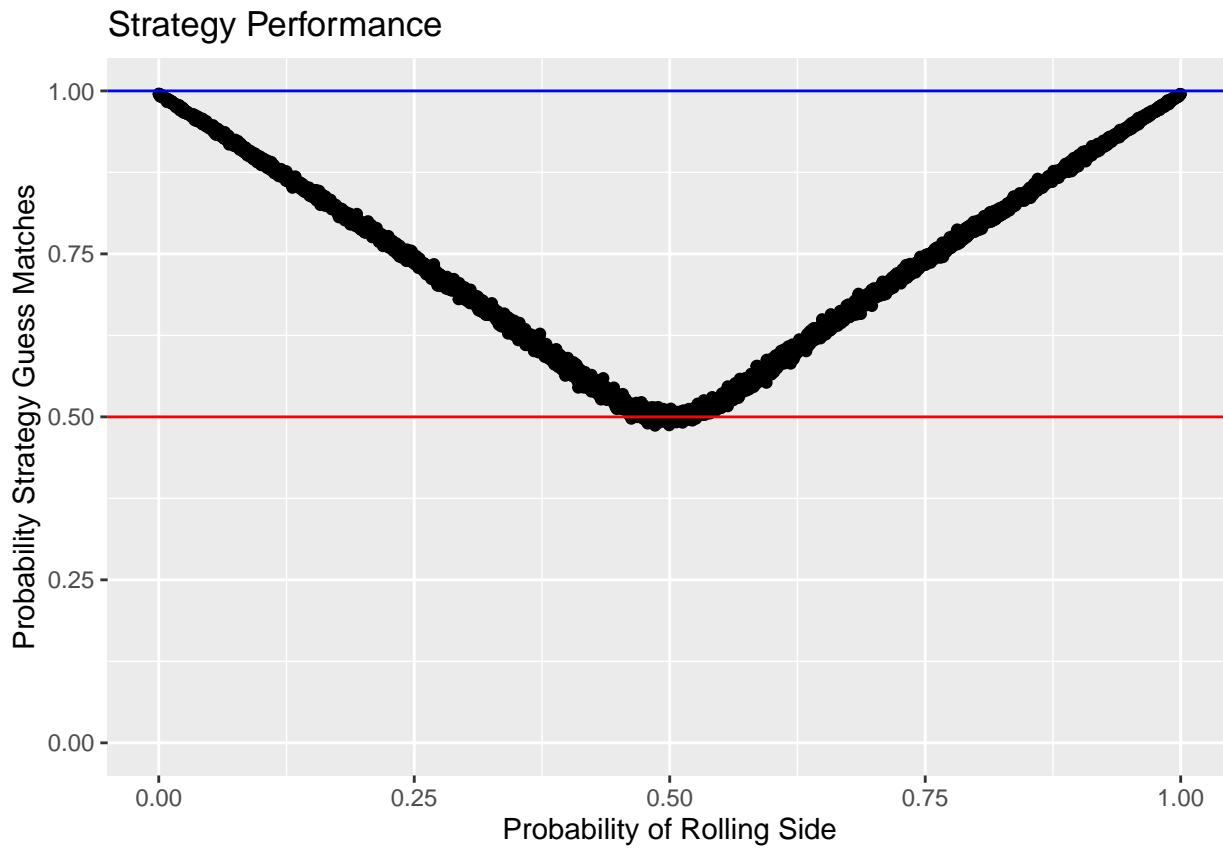
Strategy 2

```
grapher(N = 2, strat = strategy_2)
```



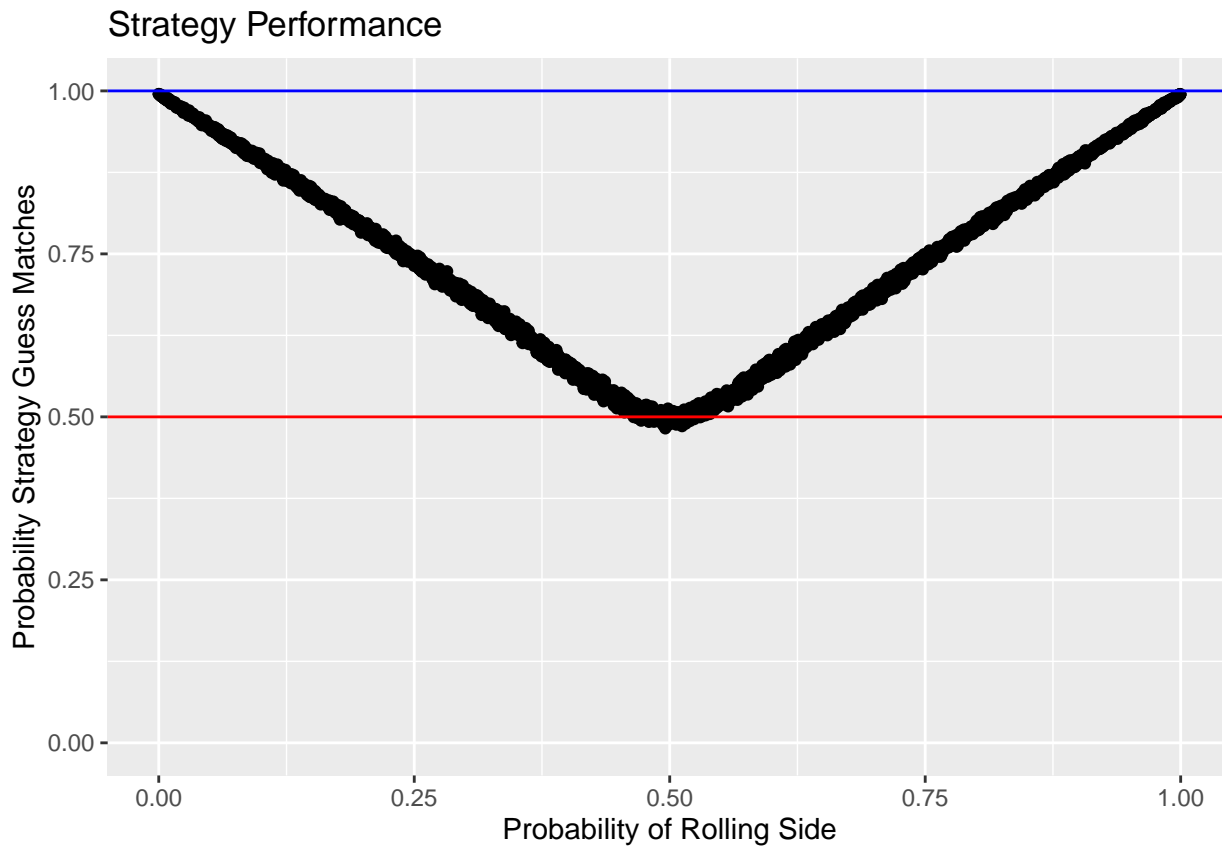
Strategy 3

```
grapher(N = 2, strat = strategy_3)
```



Strategy 4

```
grapher(N = 2, strat = strategy_4)
```

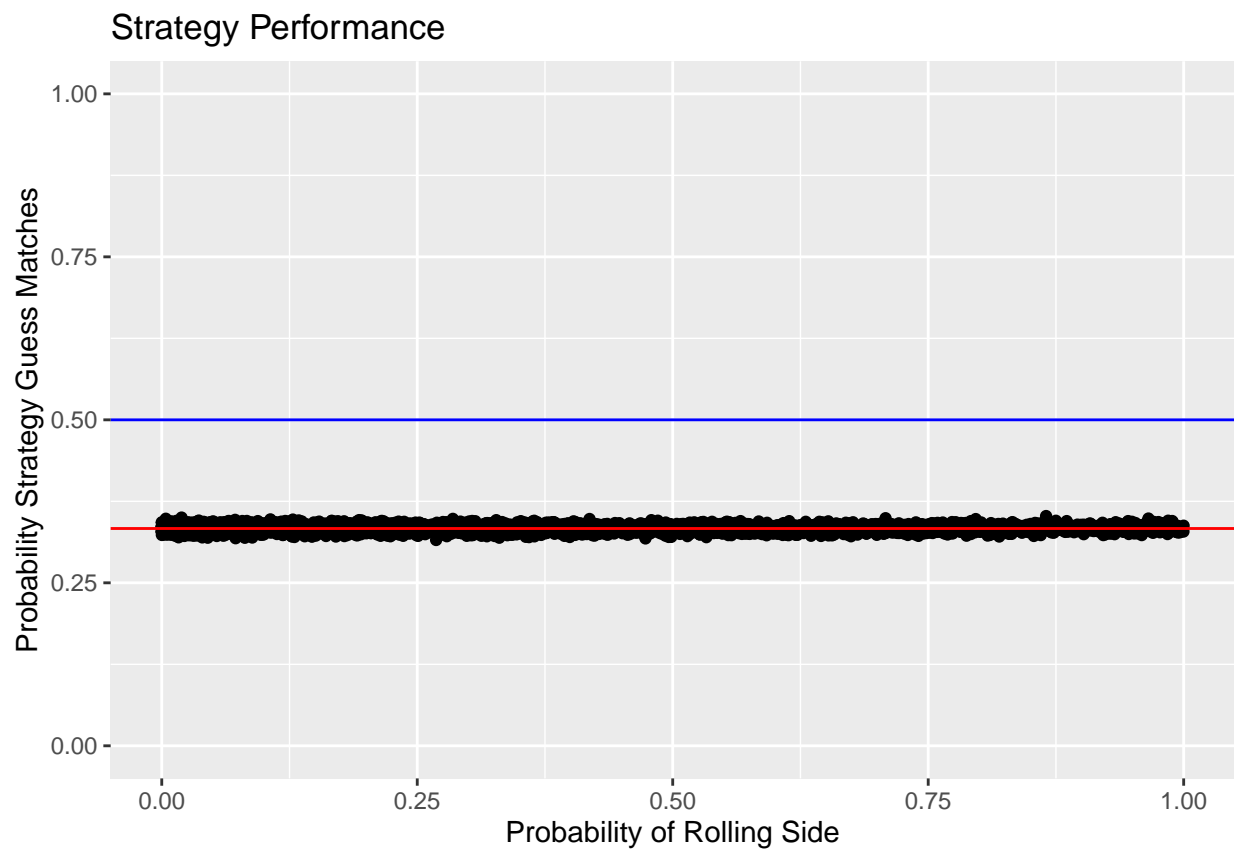


This produces exactly the results you would expect. The first strategy is level and matches half of the time no matter the probability for the “dice”, the second strategy performs very well at high and low probabilities as it will guess what the previous “roll” was which will be dominated by one side in these edge cases. That is if the probability of one side is very high then it is likely to appear much more than the other side, and so Strategy 2 will repeatedly pick it. Strategy 3 and 4 are practically identical and perform the best, their effectiveness grows faster than strategy 2 as the “dice” becomes more unfair. They work better than Strategy 2 because they carry more information than Strategy 2 does on each roll. Strategy 2 only records the previous “roll” whereas Strategy 3 and 4 record the total number of “rolls” of each side. They all guess correctly about half the time when the “dice” is fair.

Results for $n = 3$:

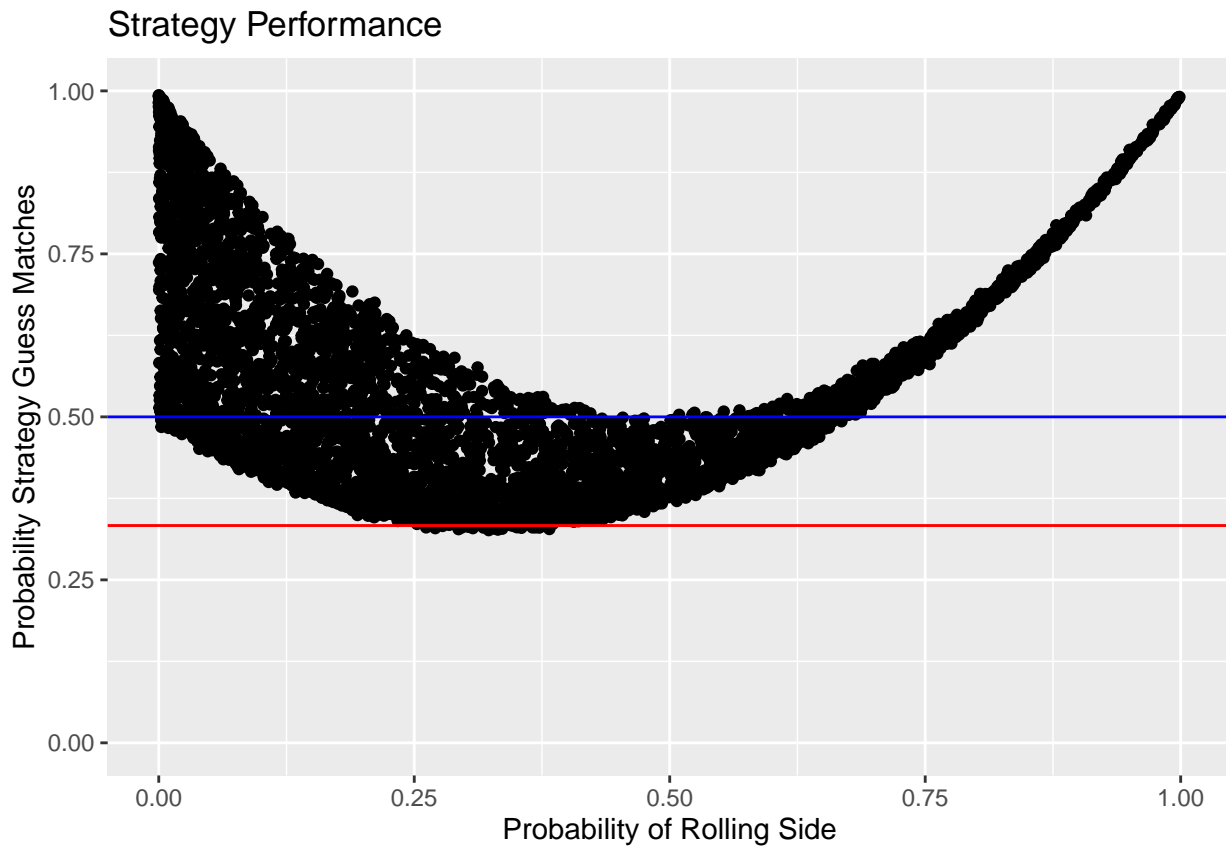
Strategy 1

```
grapher(N = 3, strat = strategy_1)
```



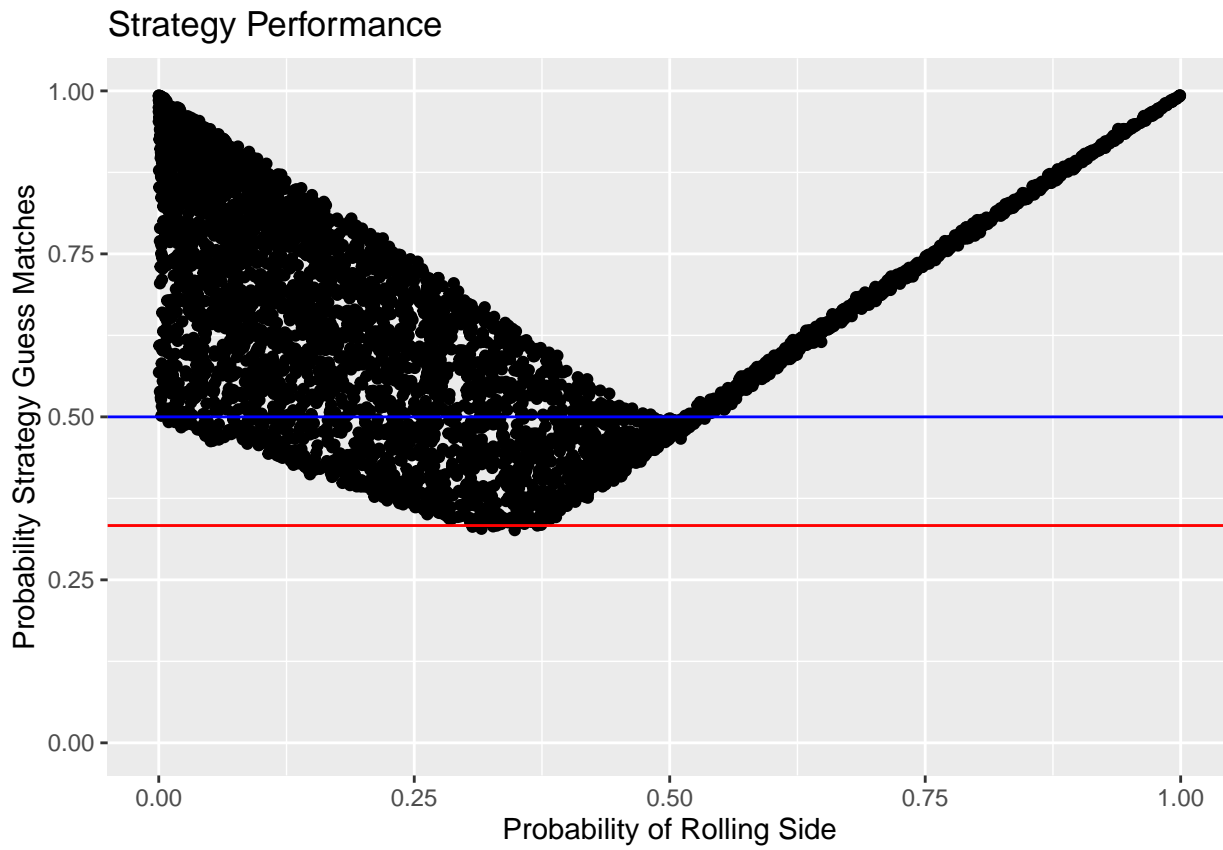
Strategy 2

```
grapher(N = 3, strat = strategy_2)
```



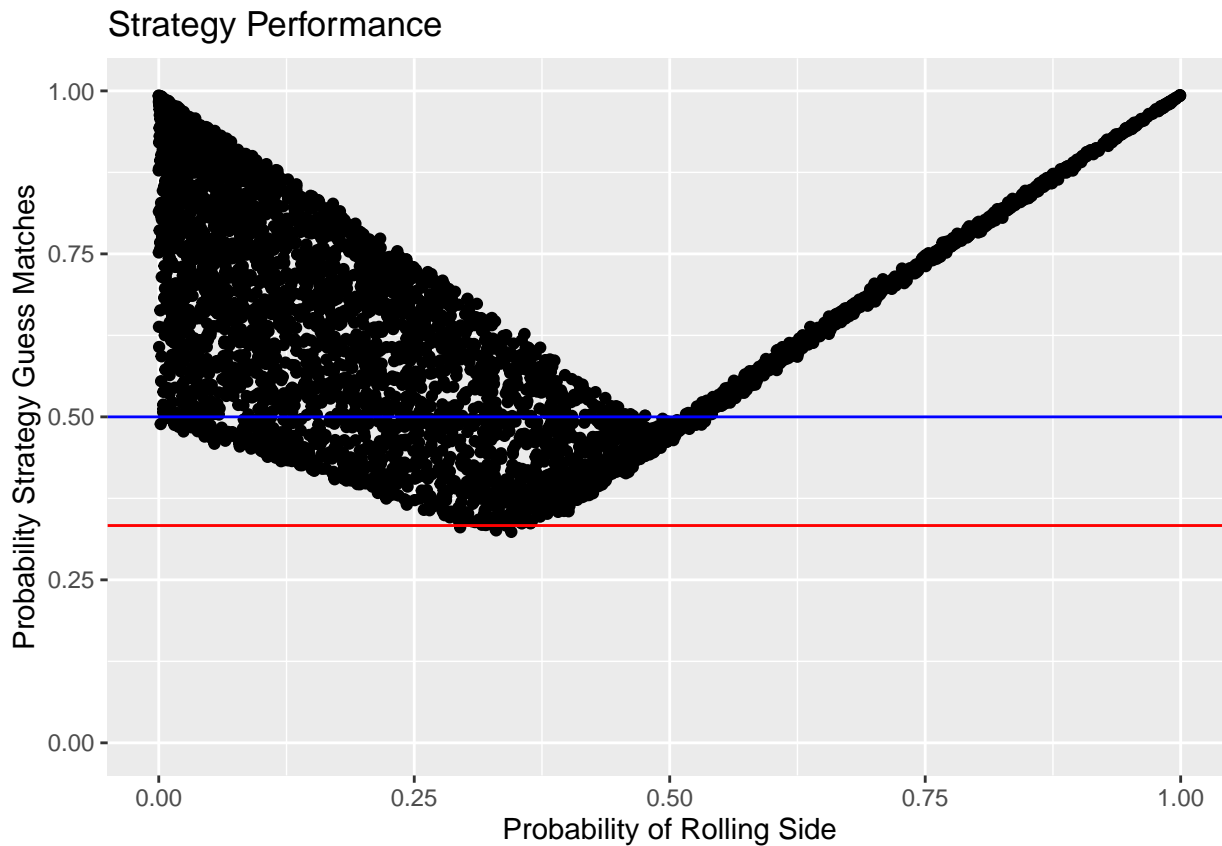
Strategy 3

```
grapher(N = 3, strat = strategy_3)
```



Strategy 4

```
grapher(N = 3, strat = strategy_4)
```

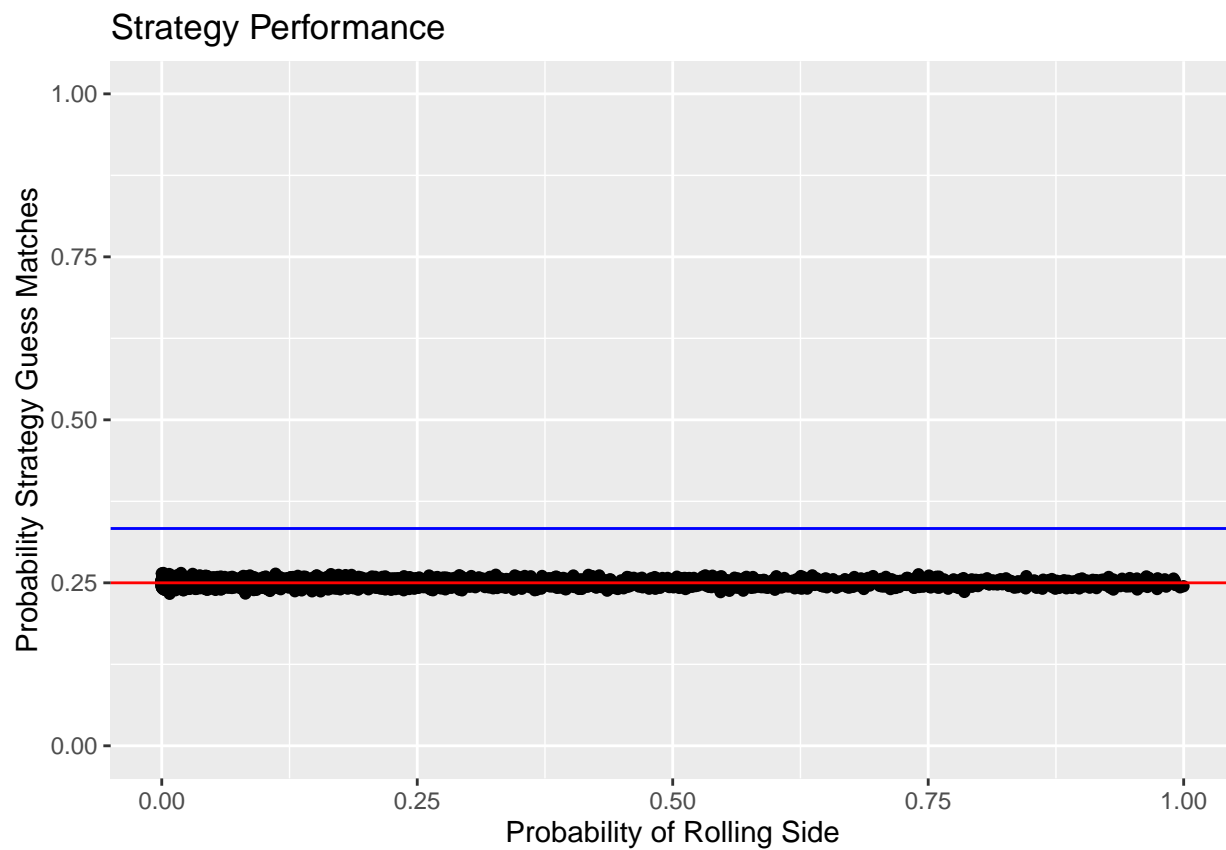


This is a little different than for $n = 2$. If the probability of rolling a side is high then the good strategies recognize that and are able to accurately guess rolls. However, if the probability of rolling a side is low then the leftover probability can either be given all to one other side (which makes it easier for the good strategies to guess) or spread more evenly (which makes it more difficult for the good strategies to guess). For example if the probability of rolling a 1 is $p_1 = 0.01$ then $p_2 + p_3 = 0.99$, if $p_2 = 0.98$ then the strategy will easily come to predict what will happen. However, since there are two sides what can happen is $p_2 = p_3 = \frac{0.99}{2}$ and so it is much more difficult to predict what happens. This can be seen in how spread out the left side of the graph is for the good strategies, namely the probability our guess matches can vary widely in this range. In general as the probability of rolling a side becomes arbitrarily small, if the leftover probability is spread evenly to the other sides then the probability the good strategies guess correctly is about $\frac{1}{n-1}$ (represented by the blue line), this was discussed before since you are essentially rolling a fair $n - 1$ sided dice. This can be seen in the graph as near the left side the points stay above the lower limit of the blue line at $\frac{1}{n-1}$.

Results for $n = 4$:

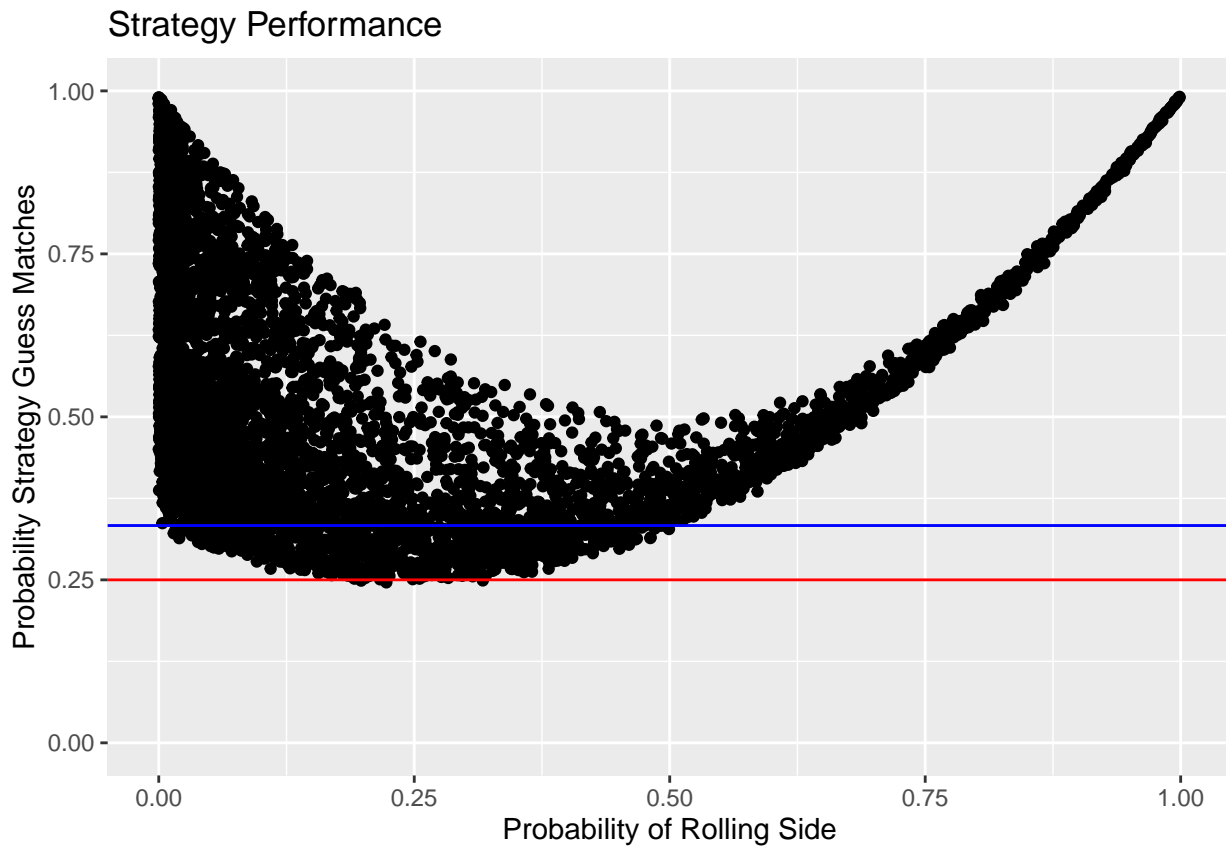
Strategy 1

```
grapher(N = 4, strat = strategy_1)
```



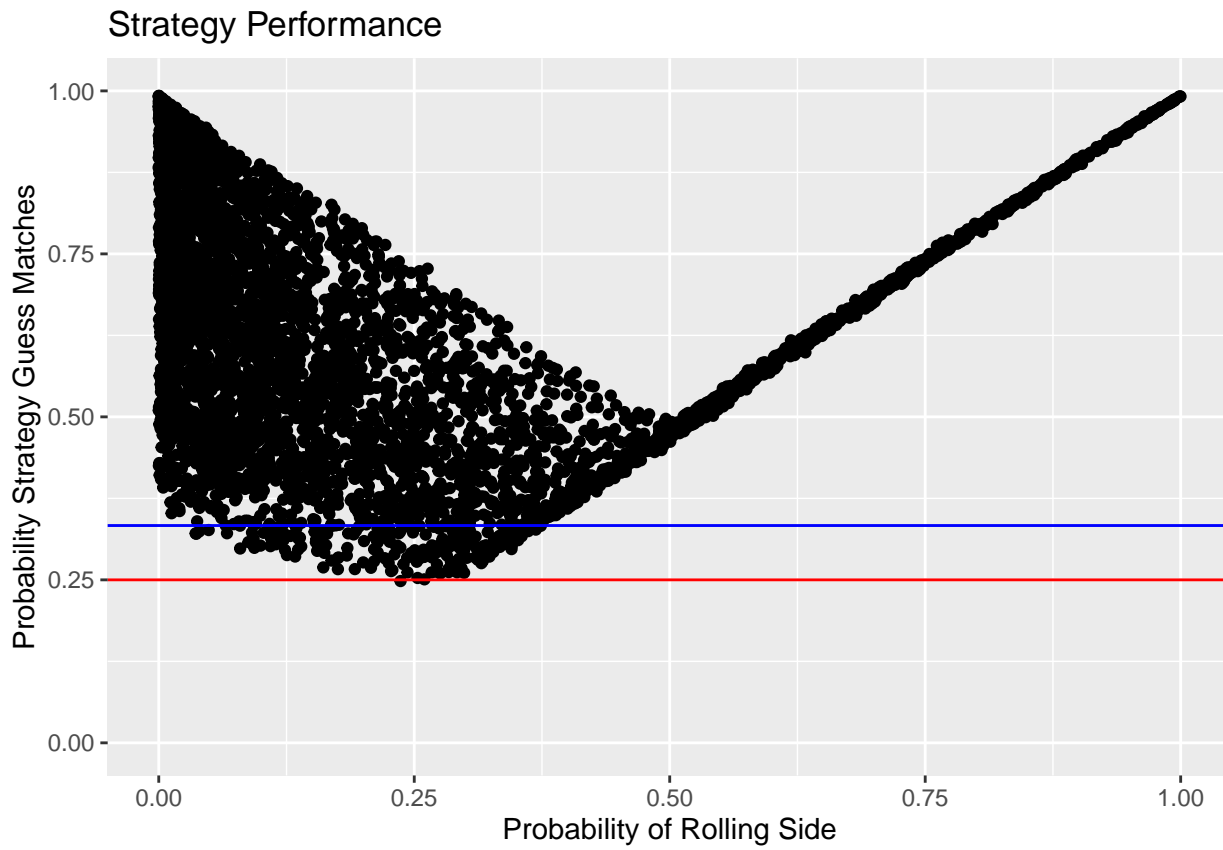
Strategy 2

```
grapher(N = 4, strat = strategy_2)
```



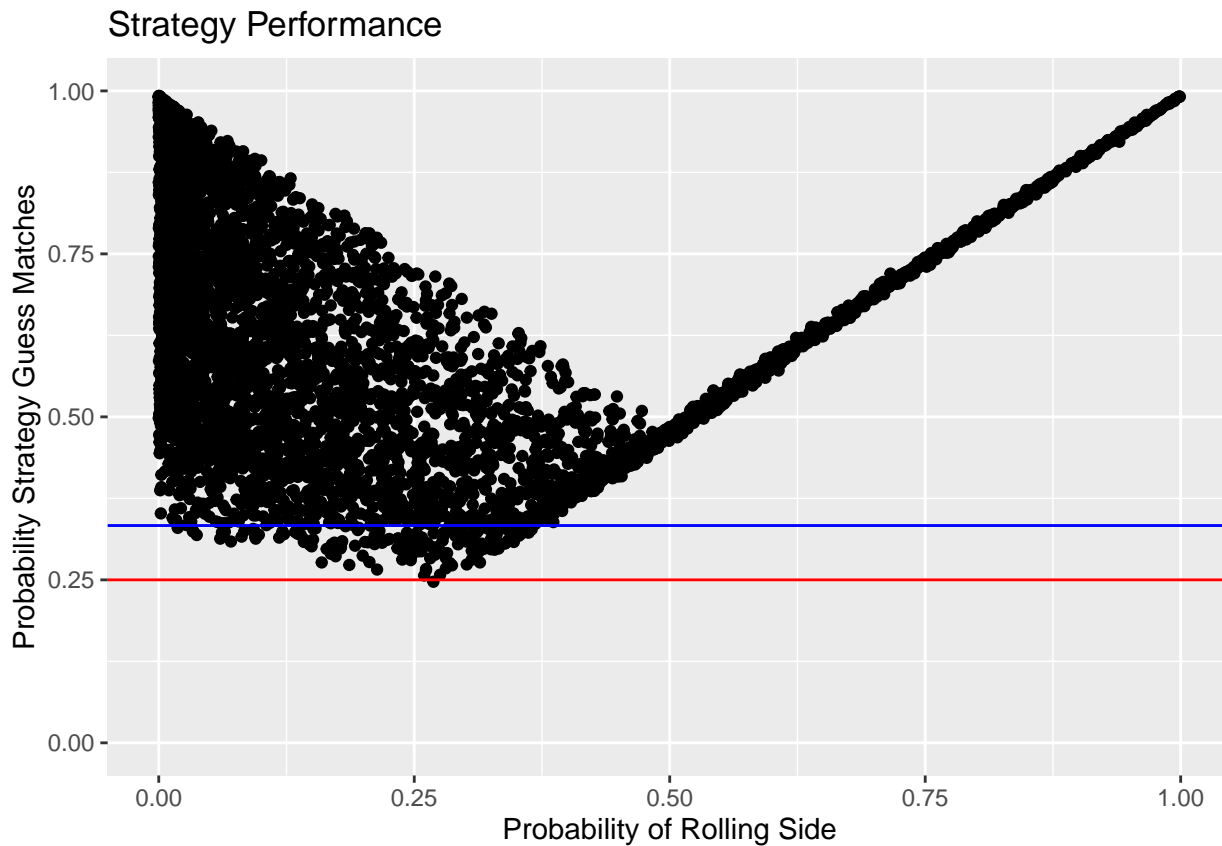
Strategy 3

```
grapher(N = 4, strat = strategy_3)
```



Strategy 4

```
grapher(N = 4, strat = strategy_4)
```



This is actually analogous to the $n = 3$ case. This is because there are several other sides to distribute probability to if the probability of rolling a side is low. Again if the probability of rolling a side is very low and the leftover probability is spread evenly to the other sides then the probability the good strategies guess correctly is about $\frac{1}{n-1}$ (represented by the blue line). This pattern of the graphs actually continues for all $n \in \mathbb{N}$. Notice how in each case near the middle every strategy approaches the red line which is the line to represent randomly guessing, and in the good strategies on the left side of the graph most of the points are above the blue line which is the line to represent randomly guessing from the remaining $n - 1$ sides.