A Questionable Method of Winning Using Constant Betting and Doubling in a Virtual Racecourse

Takashi Ohyama

nirarebakun@hotmail.com (Use “About Doubling Paper” as the subject line)
INDEPENDENT (NOT RELATED TO MY COMPANY’S BUSINESS)

LIVES IN TOKYO

Published online: 12 August 2025

Abstract

In this paper, I propose improvements to a dubious method of winning in horse racing using the strategies of constant betting and doubling (the Martingale method of progressive betting) such that rewards can be obtained with lower risk. Since it is said that applying doubling generally leads to bankruptcy, I introduce an modified method such that the risk is reduced. That is, we use doubling as a last resort when the uncollected amount reaches a predetermined value. Before using the doubling method, we start by placing a place ticket bet on the first-ranked horse with the lowest odds whose hit rate is the highest. After a win, we skip all subsequent races to avoid risking a loss. While the uncollected amount remains under a predetermined value, we continue placing win ticket bets on the first-ranked horse with odds sufficient to clear the uncollected amount. After the uncollected amount reaches the predetermined value, we use doubling for the first-ranked horse and constant betting for the second-ranked horse. By this method, we can keep the risk within a tolerable level. To analyze this strategy, I experimented by entering competitions in a virtual racecourse. While the stochastic data such as hit rate are different from actual real-world data, this model shows an upward trend with small valleys (the risk). While it is still necessary to consider what would happen in actual operation, I believe that this study can become a foothold to develop a winning method in the future.

Keywords: Doubling. Martingale. Horse racing. Constant betting. Virtual racecourse.

1 Introduction

In this paper, I consider the use of doubling to be a last resort as a method of winning. Doubling is well known as a questionable winning method. Turner (1998) compares doubling to constant betting by using computer simulations. Many websites and videos, such as Umameshi.com (2021), introduce its application to horse racing.

In Ohyama (2025), I proposed some methods to apply doubling to horse racing. In the paper, the input amount (wager) is set to the uncollected amount divided by a predetermined value. Furthermore, some additional strategies are proposed: (1) Start betting on the first-ranked horse with low odds whose hit rates are highest. (2) Skip all subsequent races after clearing the uncollected amount (the temporary accumulated loss). (3) Increase the coefficient when the uncollected amount becomes large. By these strategies, we can reduce the risk to some extent. However, a simulation consisting of 10,000 races finds that there is a risk that the amount of a single bet could rise to as much as 100,000 yen.

This paper is a continuation of the work presented in Ohyama (2025). First, I prepare the simulation data (data generated through simulated races in a virtual racecourse), including both win and place tickets. In Japan, place tickets are a type of bet where rewards are obtained when the betting horse comes in first, second, or third place (or first or second place when the number of horses is less than 8). While the returns (winnings) are lower, the hit rate (chance of winning) is higher. Considering an investment where making a profit is a priority, the high hit rate is a big attraction, even if it is a small amount.

Further, I consider to how to use constant betting to our advantage. Given a certain degree of values for hit rate and recovery rate, there is no risk of incurring a large loss with constant betting. Therefore, until and unless a certain amount of loss occurs, it is more effective to continue using constant betting (compared to the riskier doubling method) and wait for profits to emerge. Additionally, in horse racing, we can purchase multiple tickets. While using the risky doubling method with one ticket type, we can simultaneously employ constant betting with another ticket type. This may help to reduce the risk.

By making effective use of the advantages and trade-offs of doubling and constant betting, I consider whether there is an optimal winning method with less risk.

In Section 2, I explain the process of constructing the simulation data used in subsequent sections. Apart from the data, Section 2 is not related to the methods discussed in those sections and can be skipped if there is no interest in the data creation process.

Section 3 is the core of this paper. In Section 3.1, I define constants and variables, and I formulate the standard model. In Section 3.2, I show an example. In Section 3.3, I show the results of 100,000 races using the parameters and models of section 3. Looking ahead to actual operation and practical use, I confirm whether there are any bad cases or outcomes in the 100,000 simulated races.

Finally, in Section 4, I conclude this paper.

I believe successful methods of winning do exist. I hope that this study may offer support to develop them.

2 Constructing horse racing data

Before the main discussion, I will explain how to construct the racing data and the results, as in the previous paper. For verifying some of the models, it is important to use actual data, needless to say. However, I did not obtain permission to use the racing data that originates from organizations or tipsters. Further, even if I had access to that data, it is difficult to gather a sufficient amount in order to evaluate the results confidently.

For this paper, I constructed the racing data by simulating 110,000 races. The data from the first 10,000 races was used to design the models, and the data from the next 100,000 races was used to confirm the amount of risk that these models contain.

There are many factors in actual races, such as race course conditions and the weather. There are also racing types such as handicap/non-handicap racing and different racing classes. However, in this paper, we will construct data for just one type of race.

The variables described here are only used in this section. This section is unrelated to extended methods of doubling, and you can skip this section if you are only interested in those methods.

2.1 The racing model

I constructed the racing data by making simple assumptions. Watching actual races, we can see that odds and the horse conditions change from moment to moment. As the deadline approaches, the number of votes (representing weighted probabilities or or betting calculations) increases. The popularity, or evaluation of the horses, can also change depending on footage from the paddock and warm-up, as well as any final comments from tipsters. I decided to analyze horse strength and voting behavior by considering the following three stages:

1. Before one's own vote

The number of horses for each race varies from 6 to 16, and this number will be decided by a roulette wheel with predetermined arc lengths. Uniform random variables will decide how many horses run in each race. The strength of each horse is decided by an exponential distribution. The odds for each horse are decided by two votes based on voting behaviors. One vote is dependent on strength, and the other is independent of strength.

2. After one's own vote

The strength (condition) of each horse may change. I add uniform random variables to represent the strength of each horse. This will be reflected in the vote.

3. After the deadline and during the race

The strength of each horse may change again. Further, each horse number has an advantage/disadvantage. The winner is decided by a roulette wheel where the arc length for each horse is determined by its strength and horse number.

In this section, let u, u_1, u_2, u_3, be uniform random variables with values from 0 to 1. To simulate this, I decided to use the RAND function in Microsoft Excel.

The processes are as follows:

PROCESS1: Setting the number of horses, N_H

I set the number of horses for each race by using uniform random variables. Using Table 1, for example, if u is 0 to 0.01, the number of horses is 6, and if u is 0.01 to 0.03 (=0.01+0.02), the number of horses is 7.

Table 1. Occurrence probability for the number of horses

Number of Horses	Occurrence Probability	Number of Horses	Occurrence Probability	Number of Horses	Occurrence Probability
6	0.01	10	0.08	14	0.12
7	0.02	11	0.1	15	0.15
8	0.03	12	0.12	16	0.21
9	0.04	13	0.12		

PROCESS2: Setting the strength of each horse i, s_i

Strength is set by using exponential random variables. The strength is set to:

(1) $$^{(BeforeBet)}s_i = \begin{cases} \text{If } u_1<\alpha_A, & \min(\alpha_B - \varepsilon\times u_2,\ \alpha_C - \alpha_D\times \log u_3) \\ \text{Else} & \min(\alpha_E - \varepsilon\times u_2,\ \alpha_F - \alpha_G\times \log u_3) \end{cases}$$

$(\alpha_A:=0.33,\ \alpha_B:=400,\ \alpha_C:=1,\ \alpha_D:=200,$

$\alpha_E:=400,\ \alpha_F:=1,\ \alpha_G:=50,\ \varepsilon:=0.001)$

The ratio of strength for each horse is set as follows:

(2) $$^{(BeforeBet)}p_i = {}^{(BeforeBet)}s_i \div \Sigma\, {}^{(BeforeBet)}s_j$$

PROCESS3: Setting the rough sum vote related to strength, N_S, $^{(p)}N_S$

The rough sum votes for single win tickets, N_S, and the vote total for place tickets, $^{(p)}N_S$, are set as follows:

(3) $N_S = \alpha_Y - \alpha_Z \times \log u_1$

$^{(p)}N_S = \alpha_Y - \alpha_Z \times \log u_2$

$$(\alpha_Y := 500{,}000,\ \alpha_Z := 1{,}000{,}000)$$

PROCESS4: Setting the rough sum vote not related to strength, N_E, $^{(p)}N_E$

The rough sum votes not related to strength for win tickets, N_E, and those for place, $^{(p)}N_E$, are set as follows:

(4) $N_E = N_S \div \alpha_H \quad (\alpha_H := 160)$

$^{(p)}N_E = N_S \div \alpha_I \quad (\alpha_I := 1{,}000{,}000)$

PROCESS5: Setting the vote-related strength of each horse i, v_i

We can assume that the number of votes is the sum of two numbers.

The first vote follows a binomial distribution with the parameters rough sum total, N_S, and each horse's strength ratio, p_i. This assumption is based on those voters (bettors) whose forecast is formed from an analysis of horse strength.

The second vote is set to $N_E \div N_H$. This assumption can be explained by the fact that some people do not forecast and are motivated to buy tickets based on the amount of cheering, the horse name, the jockey, the horse number, and so on.

Moreover, I use an approximation of normal distribution using uniform random variables and an approximation of binomial distribution using normal distribution (see Box and Muller, 1958). That is, we can compute the number of votes as follows:

(5) $$v_i^{(S)} = \{(-2 \times \log u_1)^{0.5}\} \times \cos(2 \times \pi \times u_2)\ \{N_S \times p_i \times (1 - {}^{(\mathrm{BeforeBet})}p_i)\}^{0.5} + N_S \times {}^{(\mathrm{BeforeBet})}p_i$$

$$v_i^{(E)} = \{(-2 \times \log u_1)^{0.5}\} \times \sin(2 \times \pi \times u_2)\ \{N_E \div N_H \times (1 - (1 \div N_H))\}^{0.5} + N_E \times (1 \div N_H)$$

As a side note, $\Sigma v_i^{(S)} \neq N_S$, $\Sigma v_i^{(R)} \neq N_R$.
The total votes for horse i, v_i, is expressed as:

(6) $v_i = v_i^{(S)} + v_i^{(E)}$

Similarly, for place tickets, ${}^{(p)}v_i^{(S)}$, ${}^{(p)}v_i^{(S)}$, ${}^{(p)}v_i$ can be computed.

Furthermore, we can also set the rank of each horse, r_i and ${}^{(p)}r_i$. That is, if v_i is the largest value, then r_i=1, and if p_i is the smallest value, then $r_i = N_H$ for the race.

PROCESS6: Setting the odds for each horse, o_i, ${}^{(p)s}o_i$, ${}^{(p)l}o_i$
The refund (payout) rate for win and place tickets is 80% in Japan. Odds for win tickets, o_i, are set as follows:

(7) $o_i = \lfloor 8 \times v_i \div \Sigma v_j \rfloor \div 10$
where $\lfloor x \rfloor$ represents the round-down value of x.

For place tickets, returns will depend on who else finishes in the top three. Let $v_{(k)}$ be the k-th largest votes. If the votes of the other two top-three finishers are the top two votes, the return reaches its smallest value. For this case, the total votes for the winners, wv_i, is computed as follows:

If N_H<8 then
If ${}^{(p)}r_i$=1 then
$wv_i = v_i + v_{(2)}$
Else
$wv_i = v_i + v_{(1)}$
(8) Else
If ${}^{(p)}r_i$>2 then

$wv_i = v_i + v_{(1)} + v_{(2)}$

Elseif $^{(p)}r_i = 1$ then

$wv_i = v_i + v_{(2)} + v_{(3)}$

Else

$wv_i = v_i + v_{(1)} + v_{(3)}$

The total votes for the losers, lv_i, is computed as follows:

(9) $lv_i = \Sigma\, ^{(p)}v_j - wv_i$

The smallest odds, $^{(p)s}o_i$, is computed as follows:

If $N_H < 8$ then

$^{(p)s}o_i = \lfloor 0.08 \times (^{(p)}v_i \times 100 + lv_i \times 100 \div 2) \div {}^{(p)}v_i \rfloor \div 10$

(10) Else

$^{(p)s}o_i = \lfloor 0.08 \times (^{(p)}v_i \times 100 + lv_i \times 100 \div 3) \div {}^{(p)}v_i \rfloor \div 10$

If the votes of the other two top-three finishers are the bottom two votes of the top three, the return reaches its largest value. For this case, the total votes for the winners, wv_i, is computed as follows:

If $N_H < 8$ then

If $^{(p)}r_i = N_H$ then

$wv_i = v_i + v_{(NH\text{-}1)}$

Else

$wv_i = v_i + v_{(NH)}$

(11) Else

If $^{(p)}r_i < N_H$-1 then

$wv_i = v_i + v_{(NH\text{-}1)} + v_{(NH)}$

Elseif $^{(p)}r_i = N_H$ then

$wv_i = v_i + v_{(NH\text{-}1)} + v_{(NH\text{-}2)}$

Else

$wv_i = v_i + v_{(NH)} + v_{(NH\text{-}2)}$

Using (9) and (10), we can compute the largest odds, $^{(p)l}o_i$.

PROCESS7: Setting votes after betting

In this paper, I assume that after the vote, there will be another vote of the same size as the one before our own vote. That is, we run PROCESS3 to PROCESS6 again. Initially, I assume the strength values change as follows:

(12) $^{(AfterBet)}s_i = {}^{(BeforeBet)}s_i + (-0.5+u) \times \alpha_H \; (\alpha_H = 1)$

As in (2), the ratio of the strength for each horse is set as follows:

(13) $^{(AfterBet)}p_i = {}^{(AfterBet)}s_i \div \Sigma^{(AfterBet)}s_j$

For votes after our own bet, the vote total for single win tickets, $^{(last)}N_S$, and the vote total for place tickets, $^{(last)\,(p)}N_S$, are set as follows:

(14) $^{(last)}N_S = \alpha_Y - \alpha_Z \times \log u_1$

$^{(last)\,(p)}N_S = \alpha_Y - \alpha_Z \times \log u_2$

(α_Y:= 500,000, α_Z:=1,000,000)

Similarly, the rough sum votes not related to strength for win tickets, ${}^{(last)}N_E$, and those for place, ${}^{(last)\,(p)}N_E$, for votes following our own bet are set as follows:

(15) $\quad {}^{(last)}N_E = N_S \div \alpha_I \quad (\alpha_I := 160)$

$\quad {}^{(last)\,(p)}N_E = N_S \div \alpha_J \quad (\alpha_J := 1{,}000{,}000)$

We can compute the number of votes after our own vote as follows:

(16) $\quad {}^{(last)}v_i^{(S)} = \{(-2\times\log u_1)^{0.5}\}\times\cos(2\times\pi\times u_2)\ \{N_S\times p_i\times(1-{}^{(AfterBet)}p_i)\}^{0.5}$

$\quad + N_S\times {}^{(AfterBet)}p_i$

$\quad {}^{(last)}v_i^{(E)} = \{(-2\times\log u_1)^{0.5}\}\times\sin(2\times\pi\times u_2)\ \{N_E\div N_H\times(1-(1\div N_H))\}^{0.5}$

$\quad + N_E\times(1\div N_H)$

The total votes for horse i, ${}^{(last)}v_i$, is expressed as:

(17) $\quad {}^{(last)}v_i = {}^{(last)}v_i^{(S)} + {}^{(last)}v_i^{(E)} + v_i$

For place ticket i, ${}^{(last)(p)}v_i$ is computed similarly.

PROCESS8: Deciding on the winning horse and setting the return

Looking at actual racing data, we can see that there is an advantage/disadvantage by horse number for each racecourse. Further, cases of good luck and bad luck, unrelated to strength or horse number, can occur. Accordingly, we will set an advantage/disadvantage for each horse number, w_i, using the horse number advantage coefficient, c_i, and a uniform random variable, u, as follows (see Table 2 for coefficient values):

(18) $\quad {}^{(AfterDeadline)}s_i = {}^{(AfterBet)}s_i + (-0.5+u)\times\alpha_K \quad (\alpha_K=1)$

(19) $w_i = {}^{(\text{AfterDeadline})}s_i + \alpha_L \times c_i + \alpha_M \times u$ ($\alpha_L = 1$, $\alpha_M = 0$)

Table 2. Advantage/disadvantage coefficient for each horse number

i	c_i	i	c_i	i	c_i	i	c_i
1	0.93	5	0.91	9	1.07	13	0.96
2	1.1	6	1.07	10	0.94	14	0.94
3	0.95	7	1	11	1	15	1.08
4	1.08	8	0.92	12	1.1	16	1.1

Further, we can compute the ratio of w_i, q_i, as follows:

(20) $q_i = w_i \div \Sigma w_j$

We decide the winning horse number by using a roulette-style random selection method. That is, if $u < q_1$, then horse number 1 is the winner. If $q_1 < u < q_1 + q_2$, then horse number 2 is the winner. For place tickets, I configure (19) with different parameters as follows:

(21) If i's finishing order has already been decided then $w_i = 0$
Else $w_i = {}^{(\text{AfterDeadline})}s_i + \alpha_N \times c_i + \alpha_O \times u$ ($\alpha_N = 6$, $\alpha_O = -1$)

We can compute the ratio of w_i, q_i, as follows:

(22) $q_i = w_i \div \Sigma w_j$

PROCESS9: Refunds calculation

For single win tickets, we can compute the refunds, R, by a simple adjustment of (7) as follows:

(23) $R=\lfloor 8\times v_i\div\Sigma v_j\rfloor \times 10$ (i is winner number)

As in (8) to (10), we can compute the return for place tickets. First, the winner total vote, wv, is computed as follows:

If $N_H<8$ then

$wv= v_{(1)}+v_{(2)}$

(24) Else

$wv= v_{(1)}+v_{(2)} +v_{(3)}$

Total votes for losers, lv is computed as follows:

(25) $lv=\Sigma\, {}^{(p)}v_j - wv$

The return, ${}^{(p)}R_i$ is computed as follows:

If $N_H<8$ then

${}^{(p)}R_i=\max(100, \lfloor 0.08\times({}^{(p)}v_i\times 100+lv_i\times 100\div 2)\div{}^{(p)}v_i\rfloor\times 10)$

(26) Else

${}^{(p)}R_i=\max(100, \lfloor 0.08\times({}^{(p)}v_i\times 100+lv_i\times 100\div 3)\div{}^{(p)}v_i\rfloor\times 10)$

This equation is not intended to accurately reproduce a real-world refund formula, and the extent of any correlation is unknown. This equation merely represents the assumptions described in this paper.

PROCESS10: Delete abnormal data

Finally, the following cases involving abnormal data are deleted because they rarely occur in actual races:

・Single win odds of the most popular horse are 1.0 or under.

・Single win odds of the most popular horse are over 6.0.

・Single win odds of the second most popular horse are 2 or under.

・Single win odds are over 700.

・Single win returns are 100 yen or less.

2.2 The results of 110,000 races

To begin, I constructed data for 10,000 races using the processes described in the previous section.

Tables 3 and 4 show the odds by rank and by horse number, respectively.

Table 3. Results of odds before betting by rank

Rank	avg_odds	Rank	avg_odds	Rank	avg_odds	Rank	avg_odds
1	2.77	5	12.60	9	29.69	13	86.61
2	4.74	6	16.83	10	39.21	14	109.21
3	6.90	7	22.37	11	53.37	15	137.94
4	9.46	8	29.69	12	67.77	16	184.28

Table 4. Results of odds before betting by horse number

Number	avg_odds	Number	avg_odds	Number	avg_odds	Number	avg_odds
1	41.32	5	41.78	9	42.37	13	47.21
2	42.47	6	40.80	10	43.04	14	48.98
3	41.76	7	42.70	11	42.87	15	49.08
4	42.04	8	42.50	12	45.16	16	49.68

Table 5 shows horse odds for the first 10 races. When compared with actual data, we can see that there are some examples where the relationship between single win odds and place odds seems different. While place odds for the high-ranked horses tend to be large, those for the lower ranks tend to be small.

Table 5. Results of odds for the first 10 races

Rank	1	2	3	4	5	6	7	8	9	10
1	6_4.7 1.7～2.0	7_4.2 1.4～1.9	1_3.5 1.3～1.6	4_2.8 1.2～1.4	3_1.7 1.0～1.1	4_3.1 1.1～1.5	1_1.9 1.0～1.1	7_3.3 1.2～1.6	4_3.6 1.3～1.7	1_3.0 1.2～1.5
2	11_5.5 1.8～2.3	14_4.3 1.4～1.9	8_4.1 1.4～1.8	14_6.8 1.9～2.7	9_3.1 1.0～1.5	13_3.3 1.2～1.6	2_4.1 1.1～1.8	1_3.4 1.2～1.6	12_5.2 1.6～2.2	10_4.1 1.3～1.8
3	5_7.7 2.3～3.1	11_5.7 1.7～2.4	4_5.4 1.6～2.3	6_7.5 2.0～3.0	2_7.1 1.2～2.8	10_5.5 1.4～2.3	6_6.0 1.3～2.4	5_5.8 1.5～2.4	1_5.2 1.6～2.2	2_5.1 1.4～2.1
4	16_9.1 2.6～3.5	13_6.4 1.8～2.6	2_7.8 2.0～3.0	11_7.7 2.0～3.0	10_11.9 1.6～4.4	6_9.0 2.0～3.5	7_7.7 1.5～3.0	9_5.9 1.5～2.4	11_7.4 2.0～3.0	6_6.7 1.7～2.6
5	3_9.7 2.7～3.7	6_10.5 2.7～4.0	9_9.2 2.3～3.5	16_8.5 2.2～3.3	1_23.4 2.7～8.1	9_9.4 2.1～3.6	8_11.3 2.0～4.1	6_16.0 3.3～5.7	10_9.9 2.5～3.8	7_10.3 2.4～3.8
6	4_9.9 2.8～3.8	12_10.9 2.8～4.1	10_9.8 2.4～3.7	7_13.6 3.2～5.0	5_36.9 3.9～12.4	5_20.2 3.9～7.2	3_14.1 2.4～5.0	12_16.2 3.3～5.7	8_13.7 3.4～5.1	11_11.6 2.6～4.2
7	7_10.4 2.9～3.9	8_11.3 2.8～4.2	6_10.0 2.4～3.7	9_14.0 3.3～5.2	7_45.6 4.7～15.3	2_21.5 4.1～7.6	5_29.4 4.4～9.6	3_18.4 3.7～6.5	16_14.3 3.5～5.3	4_20.5 4.2～7.0
8	1_10.6 2.9～4.0	5_14.6 3.5～5.3	5_14.1 3.2～5.1	8_17.2 3.9～6.2	4_60.9 6.2～20.2	8_25.0 4.7～8.8	4_61.0 8.6～18.9	8_21.9 4.4～7.6	9_17.9 4.2～6.4	3_31.5 6.2～10.6
9	2_11.6 3.1～4.3	10_20.5 4.7～7.2	3_42.4 8.6～13.4	2_24.8 5.4～8.7	8_64.1 6.4～20.9	15_29.6 5.5～10.3		2_24.2 4.7～8.2	3_19.1 4.5～6.9	9_34.5 6.8～11.6
10	8_21.8 5.5～7.7	3_46.7 10.1～15.8	7_61.4 12.3～19.3	3_27.5 5.9～9.6	6_251.3 23.5～79.6	1_34.6 6.3～11.9		10_44.6 8.3～14.8	2_25.3 5.8～8.9	5_41.5 8.0～13.7
11	13_38.9 9.4～13.3	4_50.6 11.0～17.1		15_30.8 6.7～10.7		11_64.0 11.2～21.5		11_53.5 9.8～17.5	6_28.1 6.3～9.7	8_41.5 8.0～13.7
12	12_47.3 11.3～16.1	2_123.6 25.5～40.1		10_48.9 10.2～16.7		3_127.2 21.2～41.2		4_60.8 11.1～19.8	13_67.9 14.7～23.2	
13	9_53.1 12.6～18.0	9_147.2 31.1～48.9		12_59.6 12.5～20.4		12_173.2 28.9～56.5			5_112.7 23.7～37.4	
14	10_55.0 13.1～18.7	1_270.7 57.1～90.0		5_72.5 14.7～24.1		14_284.4 46.3～90.4			14_240.7 50.4～79.9	
15	15_113.4 25.7～36.6			13_170.6 34.8～57.1		7_377.5 61.0～119.3			7_500.8 98.5～156.4	
16	14_322.1 71.7～102.1			1_419.2 78.5～128.7					15_649.2 129.6～205.7	

(HorseNumber)_(SingleWinOdds)
(SmallerPlaceOdds)～(LargerPlaceOdds)

Table 6 shows the results of returns by rank. It can be seen that the recovery rates of the high ranking horses are close to the refund rate of 80%. This phenomenon can also be observed in actual racing data.

Table 7 shows the results of returns by horse number. Compared with Table 2, the advantage coefficient affects some numbers' recovery rates, while others are not affected.

Table 6. Results of returns by rank

Rank	Single Win			Place		
	Hit Rate	Recovery Rate	Avg Return	Hit Rate	Recovery Rate	Avg Return
1	30.8	76.7	249.0	67.8	82.1	121.0
2	18.0	79.5	441.4	50.2	76.5	152.3
3	12.5	80.1	643.6	38.0	72.8	191.5
4	9.5	82.4	866.2	30.8	74.6	242.2
5	7.4	83.7	1,133.8	24.0	72.2	301.0
6	5.2	76.5	1,473.9	18.7	70.7	378.0
7	4.6	82.6	1,782.4	16.2	74.8	460.3
8	3.7	83.0	2,266.9	13.7	79.5	579.8
9	2.9	82.3	2,809.8	11.5	84.9	738.0
10	2.3	81.5	3,536.3	9.2	87.9	953.6
11	1.7	85.1	5,110.6	7.9	98.9	1,249.4
12	1.3	71.0	5,658.0	6.2	98.0	1,582.9
13	1.1	80.9	7,130.3	5.8	114.3	1,981.6
14	0.8	73.9	9,378.4	5.0	129.8	2,577.0
15	0.5	68.4	13,053.2	4.0	145.0	3,601.3
16	0.6	130.8	23,014.2	3.1	175.7	5,623.9

Table 8 shows the results of the most popular horse whose odds are 2 or less. It can be seen that hit rates are high while recovery rates are relatively low.

I tried to ensure that the averages and standard deviations of odds and returns were made as close as possible to actual races. Equations (1) through (7) may be a bit complicated. While observing the results, I made adjustments to these equations and to the values of parameters. However, while some indices may be close in value, others are not. It may be impossible to confidently decide the first to third place winners through the roulette method alone. The race can sometimes be decided by just a nose. Sometimes the race situation favors certain horses regardless of their strength. Although there are different types and conditions in actual races, I am only handling one situation here. It gets further complicated when considering real-world processes and the behavior of voters. However, I am more interested in the development of a successful method of winning, so let’s move ahead to that discussion.

Table 7. Results of returns by horse number

Number	Single Win			Place		
	Hit Rate	Recovery Rate	Avg Return	Hit Rate	Recovery Rate	Avg Return
1	8.3	79.9	962.0	23.7	84.3	355.4
2	7.9	86.7	1,093.3	23.6	89.9	381.1
3	8.1	75.3	930.1	23.5	83.0	353.0
4	7.8	70.5	904.9	23.3	84.2	362.0
5	7.9	82.4	1,045.3	23.8	88.8	372.7
6	8.5	79.6	935.3	24.2	86.4	356.6
7	7.8	84.3	1,086.6	23.8	90.3	379.8
8	7.9	78.7	992.0	23.4	83.5	357.7
9	7.7	72.3	940.2	23.1	82.4	357.4
10	7.5	83.1	1,110.2	22.1	85.3	386.5
11	7.4	76.6	1,037.5	22.0	83.8	380.1
12	7.1	89.5	1,265.4	21.7	95.5	440.5
13	7.4	101.8	1,383.1	21.5	97.9	455.3
14	6.6	91.4	1,381.6	19.6	90.1	459.4
15	5.8	70.1	1,204.5	18.9	86.2	455.9
16	5.3	68.9	1,299.9	18.7	93.0	498.3

In Section 3, I will use data0; that is, the data gathered from 10,000 races. To ensure thorough validation of the models, I gathered additional racing data from 100,000 (10,000×10) more races. These data are referred to as data1 to data10. The results are shown in Tables 9 and 10. When compared with Tables 6 and 7, it can be seen that hit rates and recovery rates are relatively stable.

Table 8. Results when the most popular horse's odds are low

Rank	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2.0
1	65.4	50.0	61.3	49.7	51.3	50.0	43.1	41.6	42.7	41.8
	71.9	59.9	79.5	69.6	76.8	80.0	73.3	75.1	81.1	83.5
2	15.4	9.7	11.7	19.8	17.1	17.6	19.6	19.4	20.2	19.1
	117.9	54.7	76.9	114.2	83.4	90.0	94.0	83.0	95.6	77.1
3	7.7	11.1	4.5	9.0	6.3	9.4	9.7	9.4	10.5	10.1
	87.3	126.5	42.4	88.6	55.0	80.7	69.5	70.2	78.6	74.4
4	3.8	6.9	4.5	4.8	6.3	6.8	6.2	9.4	9.1	8.2
	56.2	126.8	65.0	56.7	84.8	82.1	68.3	102.3	97.5	80.7
5	3.8	9.7	6.3	4.8	7.5	7.2	4.4	6.7	5.4	5.4
	97.1	173.2	83.0	68.5	138.9	125.8	67.4	103.3	67.5	65.4
6	0.0	2.8	4.5	4.8	2.1	2.2	4.1	4.7	2.6	4.1
	0.0	53.3	113.9	115.1	43.1	39.5	87.1	86.2	46.7	98.3
7	1.9	5.6	1.8	1.8	1.7	1.1	3.8	3.8	4.0	3.9
	125.6	168.5	41.3	36.5	37.3	23.8	106.7	79.8	95.2	94.7
8	1.9	2.8	1.8	4.8	1.7	2.2	2.9	1.8	1.7	2.1
	79.0	147.2	122.3	143.1	52.0	70.4	84.3	62.9	50.6	52.4
9	0.0	1.4	0.9	0.0	2.5	1.8	2.9	1.2	1.1	1.3
	0.0	147.1	22.3	0.0	155.1	113.5	116.2	58.6	49.0	48.5
10	0.0	0.0	1.8	0.6	2.5	0.4	0.9	0.3	0.3	2.8
	0.0	0.0	77.9	31.0	175.3	19.2	65.0	11.4	8.6	169.5
11	0.0	0.0	0.9	0.0	1.3	0.7	0.6	0.9	0.3	0.5
	0.0	0.0	81.3	0.0	215.5	55.8	36.1	55.8	63.4	51.9
12	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.6	0.9	0.5
	0.0	0.0	0.0	0.0	0.0	0.0	57.1	24.8	102.2	41.3
13	0.0	0.0	0.0	0.0	0.0	0.4	0.6	0.0	0.3	0.0
	0.0	0.0	0.0	0.0	0.0	43.0	49.5	0.0	26.1	0.0
14	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.3	0.6	0.3
	0.0	0.0	0.0	0.0	0.0	0.0	11.3	25.0	72.3	14.1
15	0.0	0.0	0.0	0.0	0.0	0.4	0.0	0.0	0.3	0.0
	0.0	0.0	0.0	0.0	0.0	45.1	0.0	0.0	31.1	0.0
16	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Race Cnt	52	72	111	167	240	278	341	341	351	388

(HitRate)
(RecoveryRate)

Table 9. Results of 100,000 races by rank

Rank	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
data0	30.8	18.0	12.5	9.5	7.4	5.2	4.6	3.7	2.9	2.3	1.7	1.3	1.1	0.8	0.5	0.6
	76.7	79.5	80.1	82.4	83.7	76.5	82.6	83.0	82.3	81.5	85.1	71.0	80.9	73.9	68.4	130.8
data1	30.9	18.0	12.7	9.6	7.5	5.1	4.1	3.6	2.8	2.1	1.8	1.4	1.2	0.9	0.8	0.6
	77.0	78.5	80.5	83.5	83.7	73.8	76.2	84.2	80.0	74.6	85.8	78.9	110.9	85.5	81.9	93.8
data2	31.3	18.2	12.8	9.0	7.0	5.6	4.3	3.7	2.8	2.0	1.8	1.3	1.3	0.9	0.5	0.2
	78.5	78.1	82.3	78.3	77.8	81.6	76.6	82.9	84.6	76.5	77.5	77.7	105.5	90.1	48.6	37.8
data3	30.1	18.3	12.6	9.3	7.1	5.4	4.7	3.7	2.7	2.6	1.9	1.3	1.3	0.8	0.8	0.5
	74.6	78.7	80.4	80.8	81.5	77.6	87.5	86.0	83.3	85.3	89.0	83.9	93.1	99.4	92.9	110.3
data4	31.4	18.7	12.4	9.2	6.7	5.5	3.8	3.6	2.8	2.3	1.9	1.5	1.1	1.0	0.7	0.5
	78.0	81.4	78.9	79.3	76.5	77.2	70.4	78.6	89.3	76.6	96.3	84.2	76.5	84.2	118.2	94.3
data5	31.7	18.9	12.0	9.1	6.9	5.4	4.3	3.1	2.9	2.2	1.9	1.3	1.2	0.9	0.6	0.2
	79.2	82.5	78.2	77.9	77.8	77.6	78.8	71.0	83.2	76.8	96.8	76.3	83.3	77.4	54.6	28.6
data6	31.3	18.1	12.4	9.5	7.2	5.4	4.1	3.5	2.6	2.4	1.6	1.3	1.2	1.2	0.8	0.5
	78.0	78.9	80.4	83.3	81.5	78.0	74.8	78.9	74.4	91.2	76.3	87.0	106.7	116.1	89.2	92.0
data7	32.0	17.8	12.2	9.1	7.1	5.9	4.1	3.4	2.8	2.4	1.6	1.4	0.9	0.8	0.6	0.6
	79.6	77.6	78.9	77.7	82.1	84.1	74.8	78.0	82.4	86.1	75.0	86.0	66.4	86.8	65.0	91.9
data8	30.7	18.1	12.5	9.8	7.4	5.0	4.7	3.5	2.9	2.3	1.8	1.3	1.2	0.7	0.5	0.5
	76.4	79.4	80.2	85.0	83.3	74.6	84.1	80.4	82.6	82.5	90.4	73.0	87.9	72.8	63.2	120.5
data9	31.5	17.7	12.4	9.3	7.3	5.3	4.2	3.6	2.9	2.1	1.7	1.5	1.4	0.8	0.7	0.6
	78.7	77.3	79.5	81.5	81.2	75.6	75.9	82.2	85.3	76.8	76.6	85.9	118.0	81.9	73.9	96.3
data10	30.7	18.2	13.1	9.2	7.1	5.5	4.3	3.8	2.7	2.2	1.9	1.2	1.2	0.9	0.6	0.3
	77.2	77.7	83.9	79.2	80.0	79.1	78.9	89.4	83.6	79.2	84.2	68.1	86.3	94.4	56.7	71.3

Table 10. Results of 100,000 races by horse number

HorseN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
data0	8.3	7.9	8.1	7.8	7.9	8.5	7.8	7.9	7.7	7.5	7.4	7.1	7.4	6.6	5.8	5.3
	79.9	86.7	75.3	70.5	82.4	79.6	84.3	78.7	72.3	83.1	76.6	89.5	101.8	91.4	70.1	68.9
data1	8.2	8.1	7.8	7.7	8.0	7.6	7.9	8.4	8.2	7.3	7.2	7.2	7.2	6.8	7.0	5.6
	81.5	83.3	77.2	76.5	82.2	78.8	83.3	78.6	87.2	71.0	86.5	85.0	87.5	79.4	99.3	84.6
data2	8.2	8.0	8.4	8.1	8.3	8.0	8.2	8.0	7.8	7.5	7.0	6.7	6.6	6.4	6.3	5.5
	85.1	79.4	73.9	90.5	91.0	79.2	83.2	81.2	75.6	76.6	79.8	78.2	64.0	64.4	74.7	68.8
data3	8.4	7.7	7.6	8.7	8.1	8.2	7.9	7.9	7.8	7.1	7.5	6.9	7.3	6.5	6.0	5.3
	88.5	70.3	71.2	99.1	82.0	93.0	73.7	85.4	87.9	89.1	87.9	84.4	95.2	75.0	83.9	64.4
data4	7.8	8.1	8.1	7.8	8.7	8.3	7.9	7.5	8.0	7.4	7.0	7.3	6.3	6.8	7.2	6.7
	78.6	85.5	77.1	72.0	76.9	83.7	88.2	75.2	82.5	78.9	85.2	79.8	80.4	95.2	100.1	87.3
data5	8.0	8.4	7.8	8.5	7.9	8.1	7.6	7.7	8.0	7.7	7.3	6.6	6.9	6.5	7.2	6.1
	86.3	77.2	79.3	81.7	81.4	78.6	81.9	70.8	83.5	73.6	68.7	77.2	73.6	66.9	90.0	72.3
data6	8.2	8.4	8.0	8.3	8.2	8.1	7.7	7.5	7.8	7.4	7.4	7.7	6.5	6.0	6.0	5.8
	83.1	86.3	80.8	86.9	86.8	80.8	78.9	74.7	83.2	86.7	77.3	96.5	83.5	76.5	81.5	89.0
data7	8.3	7.7	7.9	8.0	7.2	8.5	8.2	8.3	8.2	7.4	8.1	6.7	6.3	6.6	6.0	6.5
	83.3	75.2	75.4	72.5	73.6	81.7	82.9	81.1	77.5	82.1	85.4	79.8	81.6	89.1	85.4	63.7
data8	8.2	8.0	8.0	7.8	7.7	8.3	8.1	8.0	7.6	7.4	7.3	7.1	7.4	6.8	6.0	5.5
	77.9	89.3	76.1	72.1	81.9	77.9	87.6	78.9	72.1	80.2	76.9	93.6	95.9	94.2	76.8	70.6
data9	8.1	7.8	7.9	8.1	8.2	7.8	7.9	8.1	8.1	7.4	7.3	6.9	7.2	6.8	6.6	5.3
	74.5	84.4	77.0	86.8	92.6	78.1	78.5	77.3	86.6	74.9	91.3	77.6	84.6	74.4	91.9	66.8
data10	8.4	8.2	8.4	8.2	8.5	8.0	7.8	8.1	7.8	7.1	6.8	6.7	6.7	6.5	6.5	5.9
	89.5	76.4	79.5	86.5	90.6	86.6	81.0	83.5	79.7	76.3	74.8	73.5	69.5	60.1	73.2	80.4

3. A method of combining constant betting and doubling

3.1. Steps of the model

We define the following constants and variables:

Definition of Constants

We define the following predetermined constants:

v: Number of races per 1 period. We can freely define 1 period to 1 day, 2 days, 1 week, 1 month etc. We restart betting the first race of the next period, that is, restart betting when mod(n,v)=1 where mod(y, x) means remainder when y divided by x.

sw_const: Starting odds of the first ranked horse for single win ticket.

p_const: Starting odds of the first ranked horse for place ticket. We start betting when the first rank's odds are this value or under. (During higher odds, we skip all races.) If the odds are sw_const or under, we bet single win. If the odds are larger than sw_const, we bet place.

a_{init}: Initial betting amount. In this paper, a_{init}=100.

swr_const: Odds of the first ranked horse for single win ticket when uncollected amount remains.

pr_const: Odds of the first ranked horse for place ticket when uncollected amount remains.

sws_const: Odds of the first ranked overwhelmingly strong horse for single win ticket when uncollected amount remains. Low odds first ranked horses have higher hit rates. We bet single win tickets regardless of uncollected amount.

swt_const: Lower limit of odds of the first ranked horse for single win ticket when uncollected amount remains.

swu_const: Upper limit of odds of the first ranked horse for single win ticket when uncollected amount remains.

sws2_const: Odds of the second ranked horse for single win ticket.

u^{dsta}: value of uncollected amout that we start doubling.

ζ_j : Upper limit of uncollected amount that increase coefficient, b is moved to upper value. (j=1, 2, …,5)

ξ_j: The candidates of increase parameter, b. (j=1, 2, …, 6) b is changed by uncollected amount.

Definition of Variables

Variables are defined as follows. These value changes as races progress.

n: number of races.

race_no: race number of the day (period).

σ_{vn}: Total input amount of the day.

$o_{(i)}$: Single win odds of the i-th ranked horse.

$h^1{}_n$: Horse number we bet at the n-th race.

$h^2{}_n$: Another horse number we bet at the n-th race. When uncollected amount is large, we bet another horse.

$h_{(i)}$: Horse number of the i-th ranked horse.

bet_type: We buy single win ticket when this value is 1, place ticket when this value is 2, respectively.

u_n : (after-race) uncollected amount. When losing, this value is before-race uncollected amount. When obtaining smaller than before-race uncollected value, the differences are remained. When obtaining larger than before-race uncollected value, reset to 0.

$a^1{}_n$: input amount of $h^1{}_n$.

$a^2{}_n$: input amount of $h^2{}_n$.

a_n : sum of $a^1{}_n$ and $a^2{}_n$.

t_n : (before-race) uncollected amount.

r_n: refunds of the n-th race for 100 yen betting. When losing, this value is 0.

s_n: refunds of n-th race for input amount a_n . When a_n=100, s_n=r_n .

b : increase coefficient. Input amount is computed as round up of ratio of uncollected amount and this value. This value is computed by u_n, ζ_j, ξ_j.

Definition of the value used to evaluate the result

bet_cnt: betting count.

bet^1_cnt: betting count of the first ranked horse.

bet^{1d}_cnt: betting count of the first ranked horse whose betting amount is over 100 yen.

bet^2_cnt: betting count of the second ranked horse.

P: Profit. (Σs_n- Σa_n)

hc: hit_cnt

hr: hit rate (hc÷bet_cnt)

rr: recovery rate ($\Sigma s_n \div \Sigma a_n$)

ur: update rate ((number of times P's record was broken)÷bet_cnt)

For this model, we perform the following steps iteratively:

STEPS of the standard model:

STEP1 Compute a_n and buy horse ticket a_n

Set race_no=if mod(n, ν) =0 then ν-1 else mod(n, ν)-1 end

(27) $\sigma_{\nu n}$ =0

For i=race_no to 1 step -1

$\sigma_{\nu n}$ =$\sigma_{\nu n}$+a_i

Next

If $u_{(n-1)}$=0 then

If $u_{(n-1)}$=0 and $\sigma_{\nu n}$=0 and $o_{(1)}$<sw_const

then $a^1{}_n$= a_{init}, $h^1{}_n$=$h_{(1)}$, $a^2{}_n$=0, $h^2{}_n$=-2, bet_type=1

Elseif $u_{(n-1)}=0$ and $\sigma_{vn}=0$ and $o_{(1)}<$p_const

then $a^1_n= a_{init}$, $h^1_n=h_{(1)}$, $a^2_n=0$, $h^2_n=-2$, bet_type=2

Else $a^1_n=0$, $h^1_n=-1$, $a^2_n=0$, $h^2_n=-2$, bet_type=0

Elseif $u_{(n-1)}\leqq 50$ then

If $o_{(1)}<$swr_const

then $a^1_n= a_{init}$, $h^1_n=h_{(1)}$, $a^2_n=0$, $h^2_n=-2$, bet_type=1

Elseif $o_{(1)}<$pr_const

then $a^1_n= a_{init}$, $h^1_n=h_{(1)}$, $a^2_n=0$, $h^2_n=-2$, bet_type=2

Else $a^1_n=0$, $h^1_n=-1$, $a^2_n=0$, $h^2_n=-2$, bet_type=0

Elseif $u_{(n-1)}\leqq 100$ then

If $o_{(1)}<$sws_const

then $a^1_n= a_{init}$, $h^1_n=h_{(1)}$, $a^2_n=0$, $h^2_n=-2$, bet_type=1

Elseif $o_{(1)}>$swt_const and $o_{(1)}<$swu_const

then $a^1_n= a_{init}$, $h^1_n=h_{(1)}$, $a^2_n=0$, $h^2_n=-2$, bet_type=1

Else $a^1_n=0$, $h^1_n=-1$, $a^2_n=0$, $h^2_n=-2$, bet_type=0

Elseif $u_{(n-1)}\leqq 200$ then

If $o_{(1)}<$sws_const

then $a^1_n= a_{init}$, $h^1_n=h_{(1)}$, $a^2_n=0$, $h^2_n=-2$, bet_type=1

Elseif $o_{(1)}>$swt_const+1 and $o_{(1)}<$swu_const+1

then $a^1_n= a_{init}$, $h^1_n=h_{(1)}$, $a^2_n=0$, $h^2_n=-2$, bet_type=1

Else $a^1_n=0$, $h^1_n=-1$, $a^2_n=0$, $h^2_n=-2$, bet_type=0

(28) Elseif $u_{(n-1)}\leqq 300$ then

If $o_{(1)}<$sws_const

then $a^1_n= a_{init}$, $h^1_n=h_{(1)}$, $a^2_n=0$, $h^2_n=-2$, bet_type=1

Elseif $o_{(1)}>$swt_const+2

then $a^1_n= a_{init}$, $h^1_n=h_{(1)}$, $a^2_n=0$, $h^2_n=-2$, bet_type=1

Else $a^1_n=0$, $h^1_n=-1$, $a^2_n=0$, $h^2_n=-2$, bet_type=0

Elseif $u_{(n-1)} \leqq u^{dsta}$ then

If $o_{(1)}$<sws_const

then $a^1_n = a_{init}$, $h^1_n = h_{(1)}$, bet_type=1

Elseif $o_{(1)}$>swt_const+2

then $a^1_n = a_{init}$, $h^1_n = h_{(1)}$, bet_type=1

Else a^1_n=0, h^1_n=-1 , bet_type=0

If $o_{(2)}$<sws2_const

then $a^2_n = a_{init}$, $h^2_n = h_{(2)}$

Else a^2_n=0, h^2_n=-2

Else

If $u_{(n-1)} < \zeta_1$ then $b = \xi_1$

Elseif $u_{(n-1)} < \zeta_2$ then $b = \xi_2$

Elseif $u_{(n-1)} < \zeta_3$ then $b = \xi_3$

Elseif $u_{(n-1)} < \zeta_4$ then $b = \xi_4$

Elseif $u_{(n-1)} < \zeta_5$ then $b = \xi_5$

Else $b = \xi_6$

$a^1_n = \lceil (u_{(n-1)} - a_{init}) \div b \rceil \times a_{init}$, $h^1_n = h_{(1)}$, bet_type=1

$a^2_n = a_{init}$, $h^2_n = h_{(2)}$

where $\lceil x \rceil$ represents the round-up value of x.

STEP2 Set $t_n = u_{(n-1)} + a_n$

STEP3 After the n-th race, confirm returns r_n

Compute s_n as returns (if loss s_n=0 else $s_n = r_n \times a_n \div 100$)

STEP4 Compute u_n. If $t_n < s_n$ then u_n=0. If $t_n > s_n$ then $u_n = t_n - s_n$.

STEP5 Set $n = n+1$

STEP6 Go to STEP1

3.2 An example of the model

I will show one example using data0 from Section 2. Table 11 shows the configuration of the parameters.

Table 11. An example of parameters

sw_const	1.3	ζ_1	2,000
p_const	2.0	ξ_1	300
swr_const	1.6	ζ_2	3,000
pr_const	2.5	ξ_2	800
sws_const	1.7	ζ_3	4,000
swt_const	1.9	ξ_3	2,000
swu_const	2.6	ζ_4	6,000
u^{dsta}	500	ξ_4	3,000
sws2_const	4.8	ζ_5	8,000
v	10	ξ_5	4,000
		ξ_6	5,000

When written in descriptive sentences, it looks like this:

(i) We skip all races until the first-ranked horse's odds are 2.0 or less.

(ii) If the odds are 1.3 or lower, then purchase a 100-yen win ticket for the first-ranked horse. If the odds are between 1.4 to 2.0, then purchase a 100-yen place ticket for the first-ranked horse.

(iii) If the horse bet on in (ii) won, then skip all further races for that day. If that horse lost, bet on the first-ranked horse whose odds are between 2.0 to 2.6 such that the loss is eliminated by a win.

(iv) Similarly, if there is an uncollected amount (temporary loss) of 200 yen after the race, then bet on the first-ranked horse whose odds are between 3.0 to 3.6 such that the loss is eliminated by a win.

(v) If there is an uncollected amount of 300 to 500 yen after the race, then purchase 100-yen tickets for the first- and second-ranked horses. (This is done not to eliminate the losses all at once, but in the hope that it will reduce them, even if only a little.)

(vi) If there is an uncollected amount of over 500 yen after the race, then bet using doubling for the first-ranked horse and purchase a 100-yen ticket for the second-ranked horse.

Table 12 shows the results of the first 16 races where bets were placed. (Races in which we don’t place bets are not displayed.) We can see that we start betting with races where the odds are low, so there may be an interval of waiting before betting. After a win, all remaining races are skipped, which reduces the number of races significantly.

Table 12. First 16 betting races

n	5	14	25	34	41	53	79	82	105	117	121	141	163	173	199	201
bet_cnt	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
bet_type	2	2	2	2	2	2	2	2	1	2	1	2	2	2	2	1
a^1_n	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
a^2_n	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
a_n	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
t_n	100	100	100	100	100	100	100	100	100	100	200	100	100	100	100	100
h^1_n	3	5	9	6	6	1	3	8	13	5	10	11	12	3	6	9
h^2_n	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
r_n	100	100	100	100	100	110	100	100	120		250	100	100	100	100	120
s_n	100	100	100	100	100	110	100	100	120		250	100	100	100	100	120
u_n	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0
Σa_n	100	200	300	400	500	600	700	800	900	1,000	1,100	1,200	1,300	1,400	1,500	1,600
Σs_n	100	200	300	400	500	610	710	810	930	930	1,180	1,280	1,380	1,480	1,580	1,700
P	0	0	0	0	0	10	10	10	30	-70	80	80	80	80	80	100
hc	1	2	3	4	5	6	7	8	9	9	10	11	12	13	14	15
hr	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	90.0	90.9	91.7	92.3	92.9	93.3	93.8
rr	100.0	100.0	100.0	100.0	100.0	101.7	101.4	101.3	103.3	93.0	107.3	106.7	106.2	105.7	105.3	106.3

Fig. 1 and Table 13 show the results of 10,000 races. Fig. 1(a) shows the profit change throughout the 10,000 races. We can see a gradual upward trend. There are small, short valleys, but the deep valleys that existed in Ohyama (2025) are gone. Fig. 1(b) shows the first 500 races and Fig. 1(c) shows the first 2,000 races. In Fig. 1(b), the input amount for the first-ranked horse reaches 300 yen at the 496th race. However, this amount is nothing to be afraid of. Similarly, at the 1,155th race, we must spend 600 yen to bet on the first-ranked horse. Still, this amount is not a psychological burden. Fig. 1(d) shows the worst case. At the 2,806th race, we must spend 700 yen for the first-ranked horse, and 100 yen for the second-ranked horse, for a total of 800 yen. This is the biggest crisis point in the 10,000 races, 1,522 of which contain bets (the betting count).

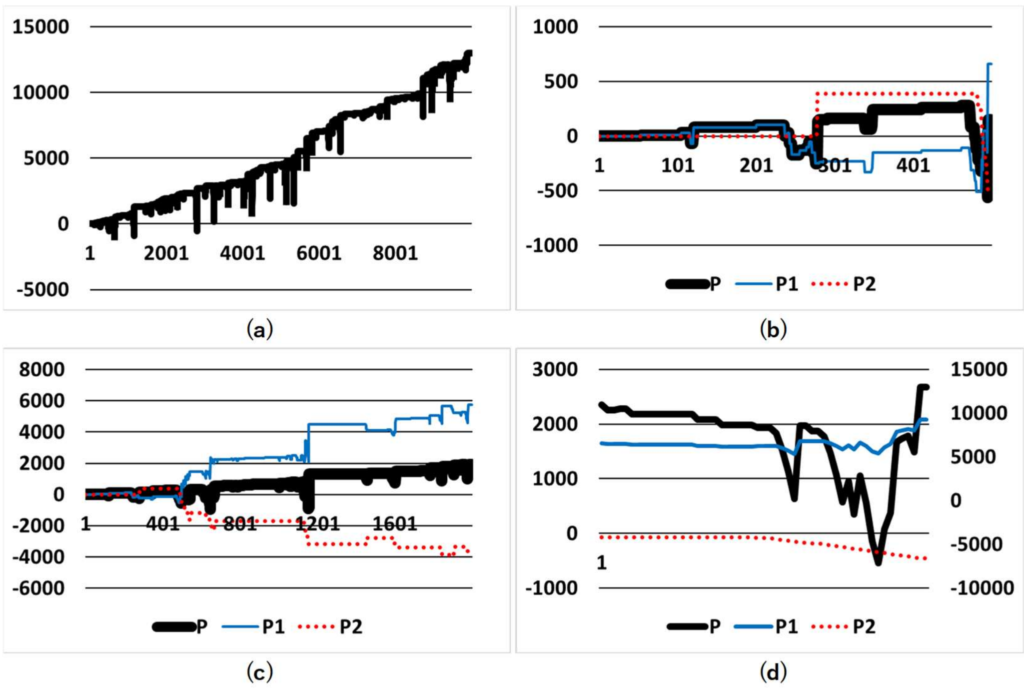

Fig. 1. An example of the combination model. (a) Results of 10,000 races. (b) First 500 races. (c) First 2,000 races. (d) The worst cases (races 2,757 to 2,811).

The first thing to note is that approximately 8,500 out of 10,000 races are skipped. As Cardano says, the best bet is not to gamble at all. The first time we bet, we bet on a horse with odds of 2 or lower, so our chances of winning are high.

Moreover, by making a place bet, our chances of winning are increased even further. Even if the bet loses, we follow that loss with a bet on the most popular horse whose odds are between 2 and 2.6, which gives us a relatively high chance of winning. Similarly, even if there is a second loss, we follow that with a bet on the most popular horse whose odds are between 3 and 3.6. Although the odds are more than three times as high, the bet-on horse is still the most popular, and its chances of winning are high. If there is a third loss, we will be betting on the most popular and the second most popular horses at the same time. At this last step, doubling will be applied to the most popular horse. Further, when doubling is applied, the coefficient *b* is set in a stepped manner so that input amount does not become large. (See Section 6 of Ohyama 2025)

Table 13. Detail of the worst cases

n/race	2,796	2,797	2,798	2,799	2,800	2,801	2,802	2,803	2,804	2,805	2,806	2,807	2,808	2,809	2,810	2,811
bet_cn	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	446
bet_ty	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-1
bet_an	300	400	600	500	300	400	600	300	400	300	700	200	200	200	300	0
bet_an	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	0
input_	400	500	700	600	400	500	700	400	500	400	800	300	300	300	400	0
bef_un	1,280	1,780	2,480	2,000	2,400	1,790	2,490	2,890	3,390	2,670	2,780	980	920	860	1,260	0
bet_ho	12	2	14	4	5	2	4	4	11	12	9	5	12	2	11	-1
bet_ho	7	1	11	7	7	14	8	10	9	7	12	11	1	4	4	-2
return100_amt/100円賭け			180		370				280	230	300	180	180		530	
return_amt/払戻			1,080		1,110				1,120	690	2,100	360	360		1,590	
aft_un	1,280	1,780	1,400	2,000	1,290	1,790	2,490	2,890	2,270	1,980	680	620	560	860	0	0
accum	57,100	57,600	58,300	58,900	59,300	59,800	60,500	60,900	61,400	61,800	62,600	62,900	63,200	63,500	63,900	63,900
accum	58,170	58,170	59,250	59,250	60,360	60,360	60,360	60,360	61,480	62,170	64,270	64,630	64,990	64,990	66,580	66,580
profit/	1,070	570	950	350	1,060	560	-140	-540	80	370	1,670	1,730	1,790	1,490	2,680	2,680
hit_cnt	269	269	270	270	271	271	271	271	272	273	274	275	276	276	277	277
hit_rat	62.3	62.1	62.2	62.1	62.2	62.0	61.9	61.7	61.8	61.9	62.0	62.1	62.2	62.0	62.1	62.1
recove	101.9	101.0	101.6	100.6	101.8	100.9	99.8	99.1	100.1	100.6	102.7	102.8	102.8	102.3	104.2	104.2

The current parameter settings were decided after hundreds of trial-and-error attempts. I believe that this is near optimal for data0, but I don't know if it is the best configuration in general. I also don't know if these parameters will work well with examples other than data0 until they are actually tested. However, in the worst case scenario with data0, we only needed to bet 800 yen. Even if only small changes were made to the data, it is reasonable to expect that there would not be any major losses. To follow up, I decided to check the results using the simulation data for 100,000 races.

3.3. Results of 100,000 races

Table 14 and Fig. 2 show the results of 100,000 races. Surprisingly, the maximum input amount was found to be 800 yen for all 10 datasets. As Fig. 2 shows, all the examples show a smooth upward trend.

The standard model implemented this strategy: (I) Start betting in races with a high probability of winning. (II) Try to avoid betting in too many races. (III) When the uncollected amount is large, bet on the second most popular horse, in addition to the most popular horse. (IV) Apply doubling to the most popular horse when the uncollected amount increases further. I believe that these efforts are the reason for the good results.

Table 14. Results of 100,000 races

	data0	data1	data2	data3	data4	data5	data6	data7	data8	data9	data10
n	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000
bet_cnt	1,522	2,591	2,291	1,829	1,716	1,538	2,508	1,976	1,631	3,268	2,373
bet^1_cnt	1,438	2,510	2,210	1,718	1,625	1,470	2,423	1,867	1,532	3,154	2,279
bet^{1d}_cnt	241	1,496	1,127	562	447	340	1,321	686	298	2,190	1,144
bet^2_cnt	370	1,617	1,250	720	593	450	1,465	846	458	2,350	1,291
maxa_n	800	800	800	800	800	800	800	800	800	800	800
maxt_n	3,520	11,600	10,660	7,430	7,050	7,060	11,070	9,370	3,520	14,820	11,520
maxu_n	3,220	11,200	10,260	7,130	6,750	6,760	10,670	9,070	3,220	14,420	11,120
hr	62.42	54.57	56.13	58.01	59.79	62.74	54.47	58.15	58.49	51.50	54.83
rr	105.49	102.15	102.63	102.63	104.09	105.03	102.04	102.82	107.41	101.84	102.44
LastP	12,990	13,170	13,370	9,200	12,620	12,530	11,590	10,490	19,830	15,260	12,530
minP	-930	-2,400	-7,540	-5,360	-3,160	-1,200	-6,500	-6,060	-1,900	-14,110	-7,520
maxP	12,990	13,170	13,370	9,200	14,410	12,830	12,190	10,490	20,230	15,260	12,530
ur	11.89	6.64	7.33	10.17	10.02	12.16	6.78	8.86	10.85	4.71	7.8

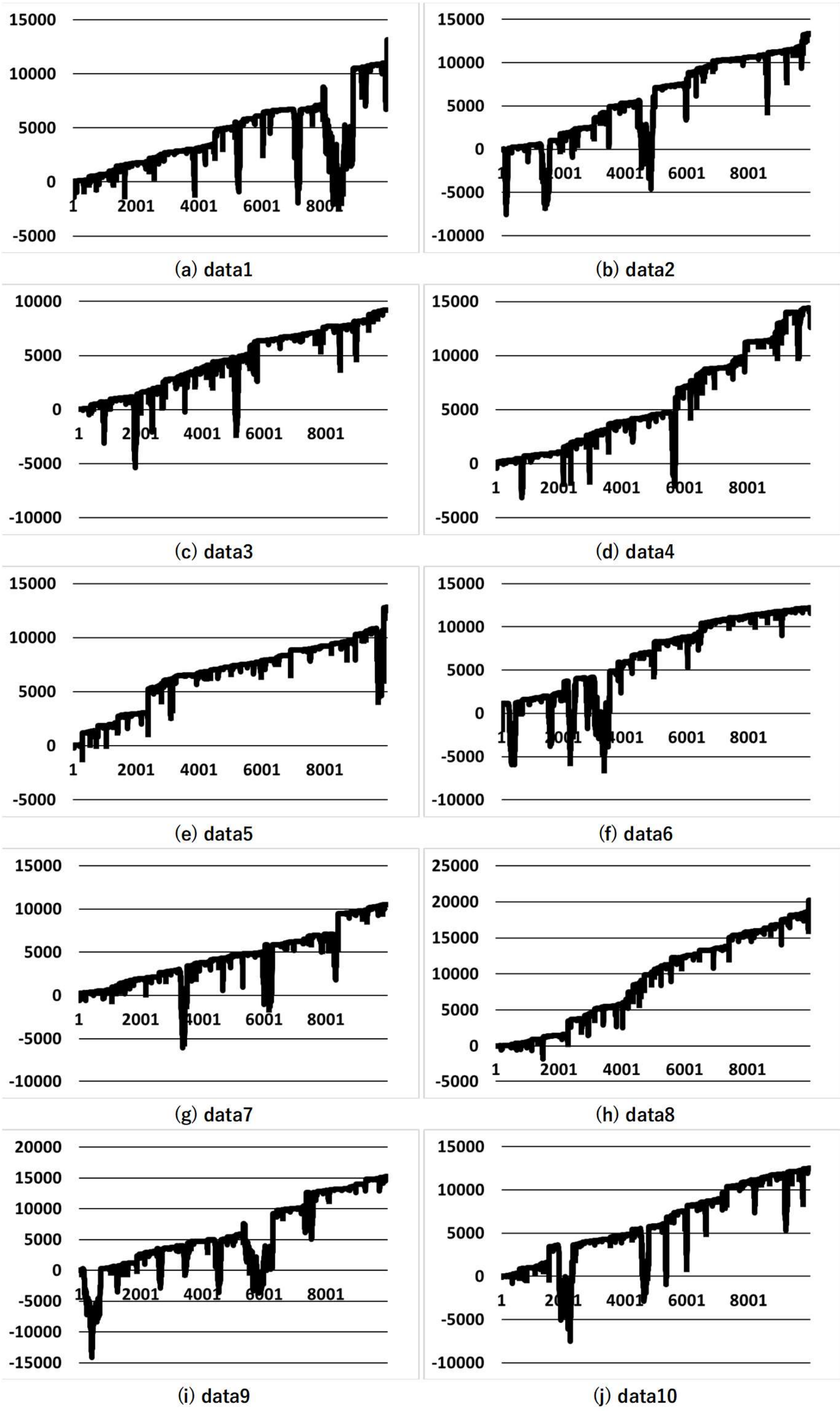

Fig. 2. Results of 100,000 races.

4 Conclusion

In this paper, I proposed a modified version of the doubling method in order to make a profit with low risk in a virtual racecourse. In this model, rather than using predictions, we continue to bet on the first-ranked or the second-ranked horse.

The procedure is relatively simple, as shown in Section 3.1.

Using doubling for the first-ranked horse has the power to reduce losses to zero all at once while the risk stays high. Using constant betting for the second-ranked horse also has the power to reduce large losses while the hit rate stays low. However, the sum of the hit rates of the first- and second-ranked horses is over 50%. Rather than implementing a method that relies solely on doubling for just one horse, hopefully this offers a way to offset losses with low risk.

I designed the model and adjusted the parameters by using data gathered from 10,000 races. I confirmed the amount of risk present in the data through 100,000 additional races (10,000×10), with the goal of looking forward to practical use of the model in future races. In all 10 cases of 10,000 races each, we observed upward trends with a recovery rate of over 102%, despite temporary losses of 10,000 yen. Although the recovery rates are small, I believe that the presence of only small temporary losses is a great achievement. Considering the models of Ohyama (2025), the loss (maximum input amount) observed there was not stable. That is, we couldn’t determine the financial budget needed to prepare for actual operation. By contrast, with the model proposed in this paper, the balance drops by only about 10,000 yen at its deepest point. If the program is bug-free, and if the racing data and assumptions are correct, we can probably operate it safely with a capital between approximately 20,000 and 100,000 yen, but there is no guarantee that this range is valid.

Admittedly, this paper relies on a not-insignificant problem with the data that was used. Differences were found between the simulated data and actual racing data in terms of hit rates, recovery rates, and average returns.

Generating highly accurate data is not easy. It is difficult to provide a logical justification for the model. Many readers will be interested to see examples of practical applications. While the risk has been significantly reduced compared to that of the previous paper, it is still important to study this further. I would also like to try putting this model into practice.

From an investment perspective, the payback rate is important. However, care must be taken when inferring anything about recovery rates. Although the recovery rate in this paper was determined to be 102% for 100,000 races, this does not mean that 1 million yen can be turned into 1.02 million yen. Fig. 1(a) and data0 of Table 14 show examples of results where the recovery rate is 105%. However, the total input was 236,400 yen, and the total return was 249,390 yen. The total profit is less than 15,000 yen. At this stage, I'm not sure what to do to make even as much as 50,000 yen. Can this be achieved by increasing the initial investment amount? I would like to continue researching this question in the future. Further, of personal interest to those of us in Japan, when we make a profit of more than 500,000 yen in one year, we are taxed on that amount. Naturally, I would like to consider how to design a model to earn 500,000 yen in one year.

In Japan, the hit and recovery rates for each rank and horse number are published for each racecourse. By using these values, we can form a mental image of the occurrence of some event. For example, if the data show that the hit rate of the most popular horse is 33%, and that of the second is 20%, then we can imagine the probability of winning with the most popular horse as roughly the same as the probability of rolling a 1 or 2 with a die. Similarly, we can imagine the probability of winning with the first or second most popular horse is larger than the probability of getting heads on a coin toss. The probabilities of dice, coin tosses, and roulette wheels are easy to imagine and have been studied in depth. There are countless models, simulations, and computer games that use these tools. These concepts will also contribute to future research on the topic of horse racing.

When I started studying horse racing in 2021, I was mostly paying attention to doubling. I believed at the time that doubling was the strongest method, although proper configuration of the parameters was difficult. When I tried this strategy out in actual races, I suffered losses of over 200,000 yen, three separate times. Based on data from about 300 races, I realized that it would be dangerous to put this method into practice. I realized the importance of testing with simulation data from over 10,000 races. (Of course, how much data is actually needed is unclear.) I analyzed this topic intensively to write the previous paper, Ohyama (2025). Even though the model was showing an overall upward trend, I found that there were several instances of large losses that occurred during the 10,000 races. In horse racing, it is inevitable that a long losing streak will occur. Similarly, with simple doubling, it is inevitable that large losses will occur.

In this paper, I focused on constant betting as well as doubling. When applying constant betting, even with consecutive losses, the overall loss is small. I believe constant betting is a good method when starting to gamble. Doubling will naturally be considered when losses continue. Further, we can reduce risk by betting on different horses.

Betting on multiple horses can be considered as a type of portfolio, a concept which has been studied deeply in the field of investment. There is also room for further development of a betting portfolio using the method presented in this paper.

I did not submit this paper to a peer-reviewed journal. I would like to make my own methods public for now, without being influenced by extraneous opinions. As the author of this paper, I myself have doubts about using virtual race data.

Regardless of the assumptions and accuracy of the data, I don't think the results are bad. However, I want to avoid being labeled a scammer by announcing this method without verifying its safety. I don't know whether this is really possible, but I intend to attempt it going forward. The results of the implementation will be published on my website at nirarebakun.com. If this turns out to be a success, the story may become famous naturally. Modifications to the model will also likely occur during its implementation. Once I have compiled these findings, I intend to write a peer-reviewed paper or book on this topic.

I think horse racing has two properties that make it stand out. One property is that, in some countries, the hit rates and recovery rates for each racecourse are made public. Alternatively, these values can be figured out through the patient collection of data. Another property is that our individual bets do not lower the odds, assuming that we do not bet very large amounts. This is a characteristic of gambling that perhaps didn't exist in Cardano's time. I am sure that more study will lead to further developments.

This man Cardano, whom I respect as a god, predicted his own death and left behind some famous quotes about gambling. In retrospect, this may not have been beneficial in terms of world peace. Many tragic incidents have occurred as a result of gambling. These cases seem to support his famous saying that gambling is not profitable. To date, mathematical and economic research on this topic has not been carried out in depth. This can be viewed as a bleak history marked by an utter failure to understand gambling, even though much time has passed. By contrast, high probability investment techniques are well studied despite the fact that there are no guarantees that profits will be made. When it comes to horse racing, the courts have proven that surefire ways of making profits in the hundreds of millions do exist.

It's difficult to predict how the world will change after high-probability methods of winning are developed, but I would like to imagine a bright future. It is said that the amount of money needed for retirement in Japan is 20 million yen. Many people are anxious about the future. Personal savings plans are taking priority, and the economy is worsening. Prices are rising, and the gap between rich and poor is widening. Many people think that being employed or working doesn't make their lives easier. I believe that if a successful method of winning can be developed, it will be possible to live a long, wealthy life and be happy with even a small amount of savings. (Still, I certainly don't want to see a world where young people reject work and attempt to live off gambling.) Personal savings will become less of a priority and economic activity will increase. I think it will give us more peace of mind. People cannot choose their parents or the country they are born into. I hope that it will be easier for people born into poverty to become wealthy. If that could happen, perhaps the world would be a little more peaceful.

Indeed, I continue to buy horse racing tickets and bet on the weekends. My current betting method differs from the model I wrote about here. Losing is far more common than winning. There is a Japanese term for horse racing called "Kanpai." This means that if the officials determine that the start was not normal, the start will be restarted. However, this word reminds me of two other meanings. The second meaning is "to toast" and the third is total defeat. There is a song called the Toast Song in Verdi's opera *La Traviata*. I love this upbeat song and often play it on trumpet and sing it. But, whenever I lose at horse racing, a minor version of this opera plays in my head. My hope is that one day this paper will be accepted, and then I will be able to perform the toast as cheerfully as Luciano Pavarotti. "*Libiamo!*"

References

Box G. EP, Muller M. E. (1958). A note on the generation of random normal deviates. *Ann. Math. Stat.*, 29, 610–611.

Ohyama T., (Published online 2025). Some Challenges for the Improvement of Doubling at a Virtual Racecourse.

http://nirarebakun.com/hr/hrpaper.html

Turner N. E., (1998). Doubling vs. Constant Bets as Strategies for Gambling, *J. of Gambling Studies,* 14, 413–429.

Umameshi.com, (Last update 2021).

https://www.umameshi.com/info/002.html [In Japanese]