applied sciences

MDPI

Article

Human Randomness in the Rock-Paper-Scissors Game

Takahiro Komai [1], Hiroaki Kurokawa [1,*] and Song-Ju Kim [2,3,*]

[1] School of Engineering, Tokyo University of Technology, 1404-1 Katakura, Hachioji 192-0982, Tokyo, Japan
[2] SOBIN Institute LLC, 3-38-7 Keyakizaka, Kawanishi 666-0145, Hyogo, Japan
[3] Department of Electrical Engineering, Tokyo University of Science, 6-3-1 Niijuku, Katsushika-Ku, Tokyo 125-8585, Japan
* Correspondence: hkuro@stf.teu.ac.jp (H.K.); kim@sobin.org (S.-J.K.)

Abstract: In this study, we investigated the human capacity to generate randomness in decision-making processes using the rock-paper-scissors (RPS) game. The randomness of the time series was evaluated using the time-series data of RPS moves made by 500 subjects who played 50 consecutive RPS games. The indices used for evaluation were the Lempel–Ziv complexity and a determinism index obtained from a recurrence plot, and these indicators represent the complexity and determinism of the time series, respectively. The acquired human RPS time-series data were compared to a pseudorandom RPS sequence generated by the Mersenne Twister and the RPS time series generated by the RPS game's strategy learned using the human RPS time series acquired via genetic programming. The results exhibited clear differences in randomness among the pseudorandom number series, the human-generated series, and the AI-generated series.

Keywords: artificial intelligence; intelligence; human intelligence; game theory; randomness; behavioral economics

check for updates

Citation: Komai, T.; Kurokawa, H.; Kim, S.-J. Human Randomness in the Rock-Paper-Scissors Game. *Appl. Sci.* **2022**, *12*, 12192. https://doi.org/10.3390/app122312192

Academic Editor: Panagiotis G. Asteris

Received: 30 August 2022
Accepted: 23 November 2022
Published: 28 November 2022

1. Introduction

The contributions of artificial intelligence (AI) to the development of industrial and social systems are remarkable, and the corresponding research and technological developments are diverse. A major trigger in the recent expansion of AI is deep learning technology [1–5]. As a result, in recent AI research, both practical AI applications and techniques that consider art and human creativity can be presented [6,7].

A huge number of AI applications have been proposed in recent decades, and the application of AI technology to practical social systems is important and expected to contribute to societal development. In addition, a great deal of knowledge has been accumulated in human brain research, i.e., natural intelligence. For example, previous studies have proposed models that differ from or complement existing AI models based on new knowledge of the human brain, and this trend has been demonstrated to be particularly promising results in recent years [8,9]. However, the proven limitations of existing AI algorithms emphasize the need for further research in this direction [10]. Thus, broad exploration of new functional expression mechanisms is essential for sustainable development of AI systems, and research into natural intelligence contributes significantly to solving such issues. In particular, we believe that extracting the functional expression mechanism of the human brain should be explored extensively in future AI research.

For example, many Go (a traditional Chinese game) algorithms, e.g., AlphaGo, and human Go strategies are very different, which is demonstrated by the fact that AlphaGo takes steps that humans would never perform, i.e., human intelligence employs a different method that the algorithmic techniques used by AlphaGo. A prominent match between the AlphaGo algorithm and former Go world champion Lee Sedol held in Seoul, Korea in 2016 demonstrated the effectiveness of AI approaches (AlphaGo) [11,12]. Here, AlphaGo won with a record of four wins to one loss; however, it is important to acknowledge that the human player beat AlphaGo in one game. This proves that it is possible for the human brain (at least a former world champion's brain) to develop and employ a winning strategy and overcome the huge difference in computing power. Although it may be difficult to describe this strategy as an algorithm, humans certainly do consider strategies to facilitate decision making. The human ability to generate a strategy is a treasure chest of intelligence; thus, we should attempt to reveal its essential qualities. This is the crucial difference between AI and natural intelligence, and research into understanding the human brain is a source of new computational methods and the emergence of intelligence [13].

Therefore, in this study, to investigate the strategies and decisions generated by the human brain, we focus on the rock-paper-scissors (RPS) game, which requires very simple decision making relative to selecting one of three actions, i.e., using the hand symbolize a rock, paper, or scissors. We analyzed the results of behaviors of human strategies produced in a limited environment and investigated the emergent capabilities and quality of the observed randomness, which is a fundamental component of human behavior. We also investigated the difference between AI and human strategies by investigating whether the characteristics of the human RPS time-series data can be reconstructed using AI technology. This study considered a simplest model to investigate human-generated strategies; however, it also pursues a fundamental question: how much randomness can the human produce? Here, human randomness refers to randomness that is independent of any non-human instrument or physical phenomenon and is generated only by intrinsic human functions. This is an important and interesting concept from a behavioral economics perspective [14,15].

RPS is a simple game that is popular worldwide. In a game of RPS, the players select one of rock, paper, or scissors, which from a trilemma relationship, and display their selection simultaneously. There are various theories regarding the origin of RPS, and many believe that the game is derived from Ishi-ken or Jaku-ken, which originated from the change of hand play, e.g., Suken or Sukumiken from China [16,17]. Recently, the name RPS game is a common name for the Jan-ken game from Japan.

In the RPS game, the rock, the paper, and the scissors form a three-way relationship; thus, there is no bias between the advantages and disadvantages among players, and winning or losing depends on only luck. In other words, it is natural to assume that there is no way to win (or increase the probability of winning) the first game against a new opponent without prior knowledge.

Thus, we change the assumptions and consider the case where the player can know and exploit the time-series sequence of hand signs the opponent has made previously. If an opponent makes a perfectly random move, no strategy can improve the probability of winning against that opponent by 33% or more because the random sequence is independent of past outputs. Conversely, if a player can make a perfectly random move, this random strategy can minimize the risk of losing regardless of the opponent's strategy; thus the random strategy is considered a Nash equilibrium in the RPS game [18].

It is natural to consider that the result of the hand signs depends on the player's decision-making processes. Thus, the time series of the opponent's hands signs is not completely random, i.e., the time series exhibits characteristics that reflect the feature of the opponent's strategy. Based on this assumption, the probability of winning an RPS game can be increased by extracting the characteristics from the time series of the opponent's hand signs. For example, there is a 100% chance of winning against an opponent who only plays the paper hand sign.

From this perspective, statistical studies of the time series of human hand signs in the RPS game and studies into the relationship with human psychology have been performed [19–21]. In addition, RPS games have recently been regarded as strategic games, and competitions have been held to compete for the strength of the strategy [22,23]. The Japan Jan-ken Association introduces "10 laws of victory" in RPS games on their webpage [24], and the World Rock Paper Scissors Association introduces rules for winning RPS games and the regularity of human RPS strategies [25]. These indicate that winning or losing is primarily determined by player strategy.

It is feasible to model the time series of human moves as an output time series from a Markov chain, which allows the player to formulate an effective strategy and estimate the opponent's strategy [26–29]. Similarly, it is natural idea to employ evolutionary algorithms e.g., genetic algorithms (GA)and genetic programming (GP), to estimate and formulate strategies [30–33]. The evaluation of strategy estimation and formulation depends on the characteristics of the opponent; thus, it is difficult to evaluate which method is superior from the viewpoint of algorithm that can create a strong strategy for the RPS game. However, the validity of the algorithms to generating the RPS strategies has been demonstrated under conditions assumed in each study.

Most previous studies have assumed that the human RPS time series has characteristics based on some sort of decision-making rules. However, to the best of our knowledge, no previous study has evaluated the randomness of the human RPS time series in actual RPS games. Naturally, there is no affirmative reason to deny this premise; however, a quantitative evaluation of the randomness of the RPS time series is useful in terms of discussing the predictability of the time series. In addition, clarifying the nature of this difference from randomness would allow us to gain insights into human behavioral decisions because humans effectively combine behaviors based on rational strategies with spontaneous or interactionally generated randomness, and there may be some meaningful intelligence in this process.

In this paper, we present two validations performed to reveal the essence of the emergence of randomness in human decision-making processes. First, we evaluate the randomness of human RPS time series (Section 2), and then we evaluate the randomness of RPS time series generated according to a strategy obtained using an AI technique (Section 3).

In Section 2, we present the results of evaluating the randomness using 500 human RPS time series in an RPS game comprising 50 consecutive games, i.e., the length of a single RPS time series is 50. Here, two indices were used in this evaluation, i.e., Lempel–Ziv complexity, which is used to evaluate the complexity of a series, and determinism (DET), which is obtained from a recurrence plot. DET is used to evaluate the determinism of a time series. The randomness of the time series is evaluated by obtaining these indices from the time series of the RPS game moves collected from the subjects and from highly plausible pseudorandom series and then comparing the relative frequency distribution of these indices.

We found that the RPS time series obtained from the subjects exhibited different characteristics compared to a series of pseudorandom numbers, i.e., they demonstrate characteristics with differing complexity and determinism from that of the pseudorandom series. Note that we evaluated the characteristics of the entire RPS time series set from 500 subjects. Here each RPS time series was generated based on a variety of individual RPS strategies; thus, the results exhibit nonrandom and biased characteristics.

In addition, in Section 3, to investigate whether the RPS time series can be reconstructed using AI technology, we evaluated the randomness of the RPS time series generated by strategies learned using genetic programming. Similar to the evaluation of the human RPS time series, here complexity and determinism were evaluated using Lempel–Ziv complexity and DET, respectively. Here, we discuss whether genetic programming can be used for a strategy that produces an RPS time series with characteristics that are similar to those of the human RPS time series.

2. Evaluation of RPS Time Series

Strategies that determine human behavior are formed by the complicated influences of various factors, either conscious or unconscious, external or internal. Thus, it is difficult to analyze human strategies and evaluate their characteristics directly. In contrast, it is easy and effective to analyze the characteristics of a strategy from the results of human behavior. Here, most of the results of actions are obtained as time-series data, and we can attempt to estimate the strategy by extracting and analyzing the features from the data.

Considering the repetitive RPS game as a mixed strategy game, the Nash equilibrium is a random strategy in which the symbols rock, paper, and scissors are selected with equal probability. In other words, the player's strategy should be random under the assumption that all players are motivated to win. Under this assumption, the human RPS time series should be obtained as random time-series data. Practically, there are differences in random sequence because human individuality is reflected in decision making. Such differences between human RPS decisions and random sequences are significant characteristics when evaluating the human ability to generate random events or sequences. Thus, in this study, to evaluate the randomness of the human RPS time series, we evaluated both complexity and determinism.

In this study, Lempel–Ziv complexity is used to evaluate complexity, and a recurrence plot is employed to evaluate determinism. The target RPS time series comprises data acquired from 500 people who played 50 consecutive RPS games. The RPS time-series data were obtained from participants attending open days at the Tokyo University of Technology from 2017–2019. Here, each subject played against an RPS algorithm using GA and GP [31,32], and the human RPS time-series data were obtained by the player's own operations using the keyboard. Note that all subjects played the RPS game after understanding the purpose of the experiments, and the goal of the player was to win the RPS game.

2.1. Lempel–Ziv Complexity

Lempel–Ziv complexity was used to evaluate the complexity of the RPS time series [34,35]. The algorithm used to obtain the Lempel–Ziv complexity of the RPS time series is described in Appendix A. Here, we show the cumulative relative frequency distribution of the Lempel–Ziv complexity from a set of RPS time-series data of 500 subjects to evaluate the randomness of the human RPS time series. Similarly, 500 series of pseudorandom numbers were prepared, and the difference between the human series and random series was evaluated according to the characteristics of a cumulative relative frequency distribution of the Lempel–Ziv complexity. Note that the cumulative relative frequency distribution represents the probability distribution of Lempel–Ziv complexity for the set of target series; thus, we evaluated the properties of the set rather than the local properties of each individual series. The set of series is characterized by the strategy used to generate the set, and we evaluated the average human strategy and pseudorandom strategy. Here, we assumed that the distribution of the cumulative relative frequency of Lempel–Ziv complexity in a set with a sufficient number of elements obtained from the same RPS strategy is reproducible.

In this evaluation, we prepared a set of RPS series acquired from 500 human participants, a set of 500 pseudorandom series, and a set of 10^8 pseudorandom series. Here, both the pseudorandom series were generated using the Mersenne Twister [36]; 500 pseudorandom series are used for comparison with human RPS series, and 10^8 pseudorandom series are used to confirm statistical properties. The length of each RPS series was 50 and the number of elements was three, i.e., rock, paper, and scissors. We prepared two datasets for the random strategy to verify whether the cumulative relative frequency distribution of the statistics obtained from the same strategy exhibits the same distribution with different sample sizes.

Figure 1 shows the cumulative frequency distribution of the Lempel–Ziv complexity obtained from these sets of RPS series. Here, the red line is the cumulative frequency distribution obtained from the set of 500 human RPS time series, the black line with circle mark is from the set of 500 pseudorandom series, and the black line with square mark is from the set of 10^8 pseudorandom series. As can be seen, both random sets exhibit similar distributions; thus, we can evaluate the average strategy of humans by analyzing a set of human hands with only 500 series.

By comparing the cumulative frequency distribution of the Lempel–Ziv complexity obtained from the set of human RPS time series and the set of pseudorandom series, we found that their characteristics clearly differ. This indicates that the randomness of the human RPS time series is lower than that of the pseudorandom series. To further observe the characteristics of each set of RPS time series, we show the distribution of relative frequency of the same data in Figure 1 on a different scale in Figure 2. Here, the horizontal axis denotes the square of the distance between a Lempel–Ziv complexity and the Lempel–Ziv complexity giving the peak value of the relative frequency distribution, and the vertical axis denotes the relative frequency of the Lempel–Ziv complexity. Note that we assumed the average of the distribution was estimated by the Lempel–Ziv complexity giving the peak value.

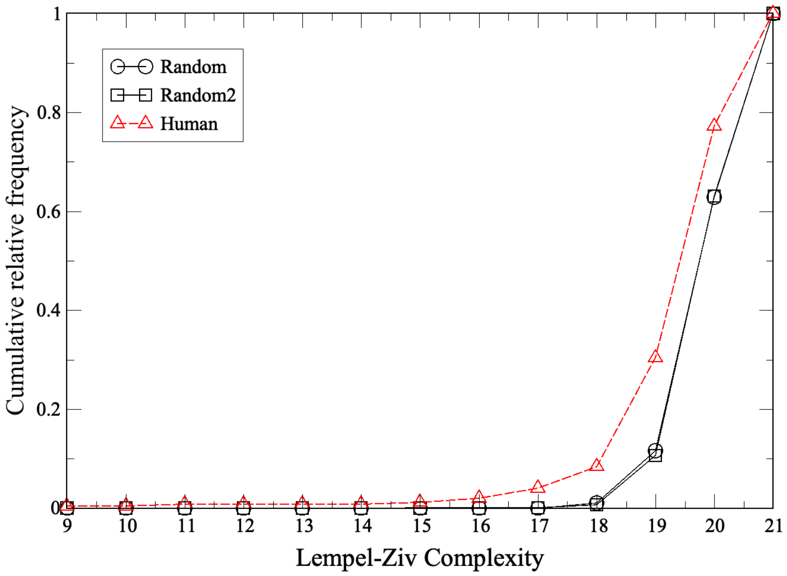

Figure 1. Cumulative frequency distribution of Lempel–Ziv complexity obtained from the set of the human RPS time series and pseudorandom series. “Random” and “Random2” are the results from the set of 500 and 10^8 pseudorandom series generated by the Mersenne Twister, respectively. As shown in Figure 2, the results from the pseudorandom series follow a normal distribution.

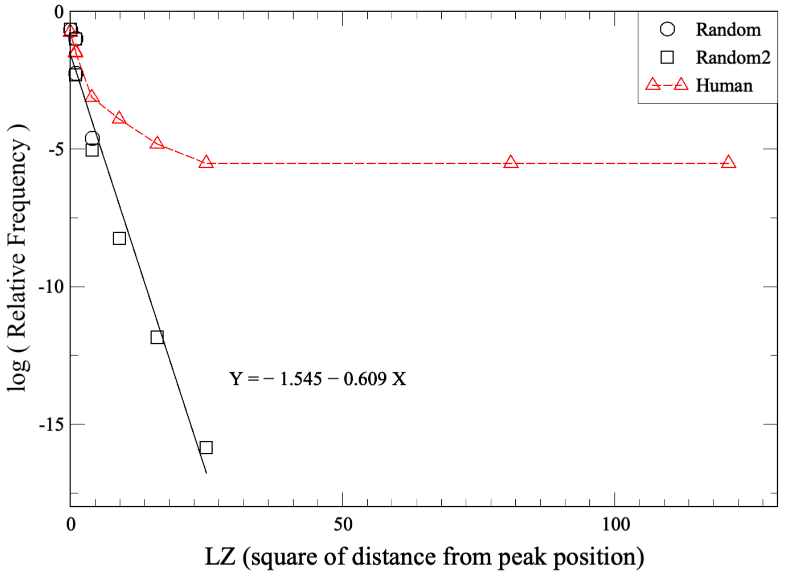

Figure 2. Distribution of relative frequency at different scale. The horizontal axis is the square of distance from the peak value. The results from the pseudorandom series (circle and square black marks) follow a normal distribution. The data are the same as those in Figure 1.

Here, the normal distribution is calculated as follows:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right). \tag{1}$$

In Equation (1), by replacing $x - \mu$ with X and taking the logarithm on both sides, we obtain the following linear equation for X^2:

$$\log f(x) = \log \frac{1}{\sqrt{2\pi\sigma^2}} + \left(-\frac{1}{2\sigma^2}\right) X^2. \tag{2}$$

Thus, in Figure 2, the straight line represents the normal distribution. Here, the slope is denoted by b; thus, the variance is calculated as follows:

$$\sigma^2 = -\frac{1}{2b}. \tag{3}$$

As shown in Figure 2, the relative frequency from the random series is distributed approximately on a straight line and that from the human RPS time series is distributed on curves that deviate considerably from the straight line. This indicates that the human RPS series has fat tail characteristics. In other words, the distribution includes a relatively large number of Lempel–Ziv complexity values away from the peak, which occurs infrequently in the random distribution. These results demonstrate a clear difference between human RPS time series and pseudorandom series and they suggest that the human strategy involves some rule-based decision-making processes that are influenced by player's individuality.

2.2. Recurrence Plot

We also investigated the determinism of the human RPS time series. Here, determinism was evaluated using a recurrence plot [37–41], which is a well-known method to analyze time-series data in nonlinear dynamical systems. The characteristics of various time-series data are visualized using the recurrence plot, which is effective for the analysis of both stationarity and determinism. Note that a time series generated based on rules that do not depend on probability is highly deterministic; thus we assumed that the recurrence plot can be used to evaluate the randomness of strategies that determine human behavior.

The determinism of time-series data can be evaluated using the ratio of the points forming diagonal lines in recurrence plot, which is often denoted by DET. The DET takes a value of [0, 1) based on the determinism of the time-series data, and it is possible to evaluate the deterministic property of chaotic time series in nonlinear dynamical systems. Thus, here, we use DET to compare the deterministic property of the set of human RPS time series to that of the pseudorandom series. Further details about the DET applied to the RPS time series are described in Appendix B.

Here, the evaluation target was the same sets of 500 series and 10^8 series of RPS time series used in the previous evaluation, and the characteristics of each set were evaluated according to the cumulative relative frequency distribution. Parameter l_{min}, i.e., the minimum length of the diagonal line in the DET calculation, was set to two. The results are shown in Figure 3, where the bin size is 0.1. A time series with a high DET value has a high deterministic property; thus, as the cumulative relative frequency distribution becomes increasingly weighted to the right, the set of series is considered to be the more deterministic (and vice versa). Randomness was assessed as a feature of the strategy used for generating the RPS time-series data.

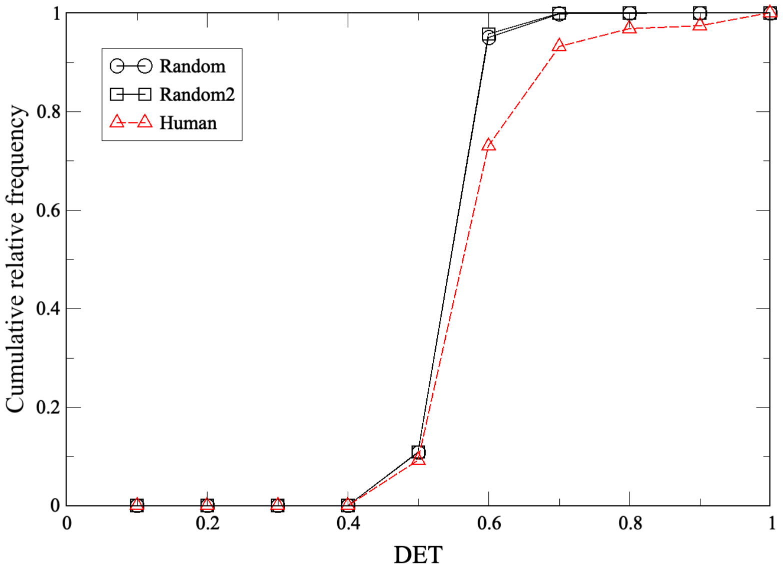

Figure 3. Cumulative relative frequency distribution of DET from recurrence plot obtained from the set of human RPS time series and pseudorandom series. Here, "Random" and "Random2" show the results from the sets of 500 and 10^8 pseudorandom series generated using the Mersenne Twister, respectively. As shown in Figure 4, the results from the pseudorandom series follow a normal distribution.

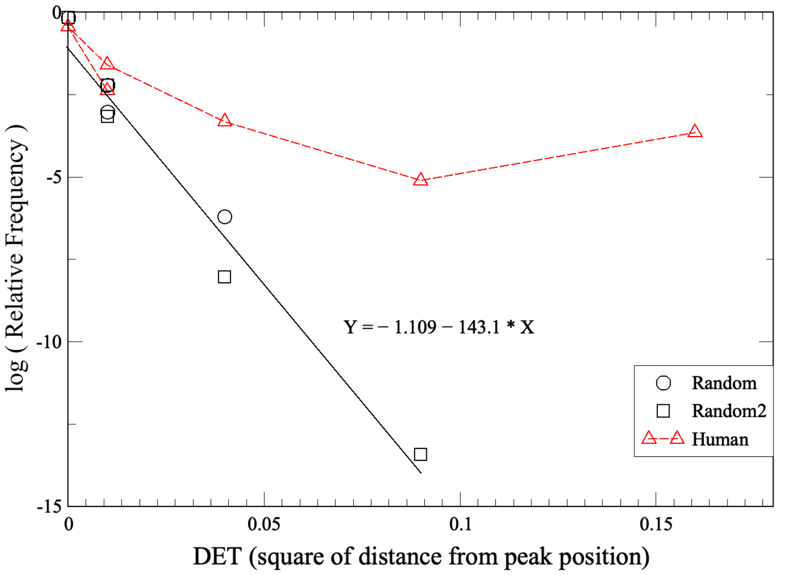

Figure 4. Distribution of relative frequency at different scale. The horizontal axis is the square of distance from the peak value. The results from the pseudorandom series (circle and square black marks) follow a normal distribution. The data are the same as those of Figure 3.

In Figure 3, the red line shows the cumulative relative frequency distribution of DET obtained from the set of human RPS time series, the black line with circle mark shows that obtained from the set of 500 pseudorandom sequences, and the black line with square mark shows that obtained from the set of 10^8 pseudorandom sequences. As can be seen, the distributions obtained from the two different pseudorandom sets are similar. Thus, as observed in the evaluation of randomness using Lempel–Ziv complexity, it is also reasonable to evaluate determinism using the set of 500 series.

Similar to the discussion regarding complexity, here, by comparing the cumulative relative frequency distribution of DET obtained from the set of human RPS time series and the set of pseudorandom series, we found that their characteristics are different. In addition, the relative frequency distribution was also evaluated, and the results are shown in Figure 4. As can be seen, the DET from the human RPS time series shows a distribution that differs from the normal distribution obtained from the random series. Note that the horizontal axis indicates the distance from the peak. The distribution observed from a human RPS series is asymmetric with respect to the peak and thus shows different characteristics on the left and right sides of the peak. The shorter of the two red lines shown in Figure 4 is the characteristic obtained from the distribution to the right of the peak. In the case of the LZ in Figure 2, there was too little data on the right side from the peak position and the difference in characteristics between the left and right sides of the peak was not very noticeable. The difference in randomness is also evident in terms of determinism, which suggests the same conclusion as that obtained in terms of Lempel–Ziv complexity in the previous section.

In summary, we investigated the differences between the sets of RPS time series generated by human strategies and the set of pseudorandom RPS time series using Lempel–Ziv complexity, which evaluates the complexity of the series, and the DET derived from the recurrence plots. The results clearly demonstrate a difference between the sets of pseudorandom series and the human RPS time series, i.e., the human RPS time series are less complex and more deterministic than the pseudorandom series. These results suggest that the average human RPS strategy includes regularities that emerge due to characteristics of human decision making. Although this finding is very natural, it should be verified in a quantitative evaluation.

3. Strategy Inference from RPS Time Series Using Genetic Programming

Is it possible for an AI system or algorithm to beat a human opponent in the RPS game? This question has attracted the interest of many researchers and engineers, and competitions for strong RPS algorithms have been held. Here, there is an essential difference between the goal of always winning and having a high win rate in a sufficient number of RPS game trials, and it is common to aim for the latter. As discussed in the previous section, the characteristics of the human RPS time series differ from those of the pseudorandom series, i.e., the human RPS time series is generated from some strategy other than a random strategy. Thus, improving of the winning rate in repetitive RPS games can be achieved by estimating the opponent's strategy using their RPS time series and predicting the next move based on the estimate strategy. In other words, this is equivalent to performing reverse engineering to infer the procedure used to generate the time series from the observable time series. Based on this idea, we can used the Markov model as a predictive model, which has been used in many studies [26–29]. In addition, strategy estimation metaheuristics has been also considered effective in many conventional studies [30–33].

In this study, we inferred the RPS game strategy using the human RPS time-series data via genetic programming, which is a type of evolutionary computation. In this experiment, we inferred the RPS strategy using 500 RPS time series from each of 500 subjects; thus, we obtained a set of strategic models for 500 the individuals who provided the data. We evaluated the complexity and determinism of the set of RPS time series generated from the set of strategies inferred using GP. The results were then compared to the sets of human RPS time series and random series discussed in the previous section. The results demonstrate that GP's strategic inference ability in RPS games in terms of the randomness of the training data. This could be a valuable finding relative to determining whether it is possible to reconstruct human RPS time series using an AI technique.

GP is an extension of the well-known GA. In the GA, a numerical sequence (frequently a binary sequence) is defined as an individual representing a candidate solution, and the objective function is minimized or maximized through repetitive generational changes in a population of individuals. In each generational change, genetic operators, e.g., crossover and mutation, are applied to individuals to search for the optimal solution. GP extends the structure of individuals to a tree structure, which allows the algorithm or program to be taken as a solution. Here, the optimal solution is searched by essentially the same algorithm as the GA. This means that GP can infer an unknown system from the observable time series and we apply this technique to the inference of RPS strategies.

In the following, we describe the procedure used to infer a human RPS strategy using GP. First, the data used for inference comprised the 500 human RPS time-series data and opponent RPS time-series data acquired from a computer opponent. Here, the length of one RPS time-series data was 50. Note that the data used to infer a single strategy comprised a pair of the RPS time-series data.

The individual for the RPS strategy was represented using a tree structure using the functions shown in Table 1 and an arbitrary integer in the range [0:99]. Here, we used previously proposed functions [30] that have been shown to be effective in previous studies [30,31]. Functions 1–7 perform arithmetic operations on arguments, and functions 8–12 reflect the game history and give some randomness to the output series. Here, function involving probabilistic elements were not considered, and all strategies described by the individual were implemented as deterministic systems.

Table 1. Functions used to infer RPS strategies.

No.	Function	Note
1	add(x, y)	$Return\ x + y$
2	sub(x, y)	$Return\ x - y$
3	multiple(x, y)	$Return\ x \times y$
4	divide(x, y)	$Return\ x/y\ if\ y == 0\ return\ x$
5	mod(x, y)	$Returnx\%y\ if\ y == 0\ return\ x$
6	plus1(x)	$Returnx + 1$
7	plus2(x)	$Returnx + 2$
8	gp-hand(x)	$Return\ the\ hand\ sign\ program\ made\ prior\ to\ the\ x\ games\ (if\ t - x \leq 0\ Return\ x)$
9	opp-hand(x)	$Return\ the\ hand\ sign\ opponent\ made\ prior\ to\ the\ x\ games\ (if\ t - x \leq 0\ Return\ x)$
10	If-r(x, y_1, y_2)	$if\ x\%3 = 0\ return\ y_1\ else\ return\ y_2$
11	If-s(x, y_1, y_2)	$if\ x\%3 = 1\ return\ y_1\ else\ return\ y_2$
12	If-p(x, y_1, y_2)	$if\ x\%3 = 2\ return\ y_1\ else\ return\ y_2$

The individual is calculated from the leaf node toward the root node, and the remainder obtained by dividing the output of the root node by three shows the RPS hand sign. The strategy described by an individual generates an RPS series by giving the opponent's RPS series. Thus, the length of the generated RPS series is the same as that of the opponent's RPS series. Stochastic elements are not included in the strategy; thus, the RPS series generated by an individual for a specific opponent's RPS series is uniquely determined.

An individual is evaluated using the Fitness calculated by Equations (4) and (5). Here, an individual that generates an RPS time series with a high similarity to a human RPS time series will have a high fitness. Here, similarity was evaluated in terms of the number of equal elements in the time series:

$$Fitness = \sum_{i=0}^{L} D_i\ , \tag{4}$$

$$D_i = \begin{cases} 1 & if\ Human\ hand\ sign_i = Program\ hand\ sign_i \\ 0 & otherwise \end{cases}\ , \tag{5}$$

where L is the length of the RPS time series, $Human\ hand\ sign_i$ is the i-th hand sign of the human RPS time series, and $Program\ hand\ sign_i$ is the i-th hand sign of the RPS time series generated by an individual. Here, the $Program\ hand\ sign_i$ sequences are generated with reference to the RPS time series data shown by the opponent's computer during data collection.

The flow of RPS strategy acquisition via GP is illustrated in Figure 5. In this study, the hyperparameters were set as follows: crossover rate: 0.95, mutation rate: 0.05, and maximum number of generations: 10^6.

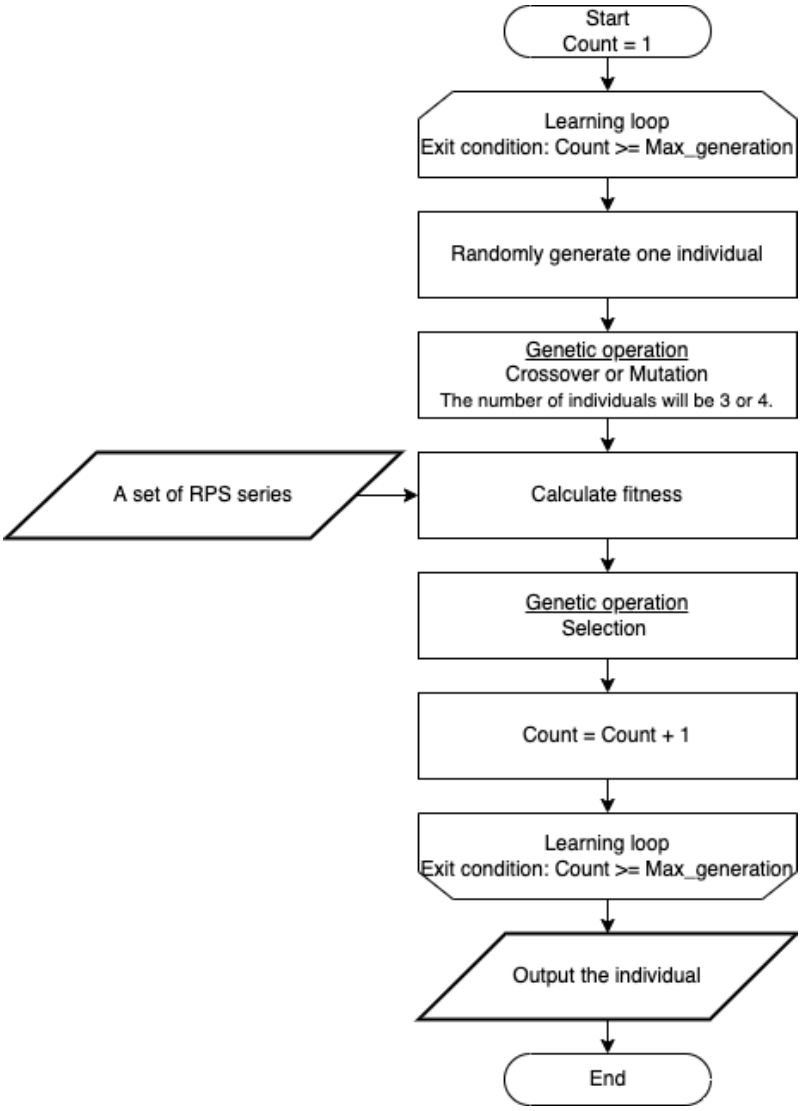

Figure 5. Flow of RPS strategy acquisition via GP.

We obtained the set of 500 RPS time-series from the set of 500 strategies inferred using a pair of RPS time series data for 500 players and their opponents. Here, the obtained 500 RPS data were evaluated according to Lempel–Ziv complexity and DET, and the results are shown in Figures 6 and 7, respectively. In addition, the relative frequency distribution was evaluated in the manner described in Section 2. The relative frequency distribution results are shown in Figures 8 and 9, respectively. In each figure, the analysis results of the RPS time series from the GP inferred strategy are added to Figures 1–4. These results demonstrate that the RPS time series obtained from the strategies estimated by GP are even less complex and more deterministic than the human RPS time series.

From the results shown in Figures 8 and 9, we observe that the relative frequency distributions obtained from the GP and human RPS time series both exhibit fat tail distributions compared to the normal distribution obtained from a pseudorandom sequence. Here, the relative frequency distribution characteristics of the human RPS series are similar to the normal distribution in the high-frequency region (i.e., near the peak of the distribution) but are larger than the normal distribution in the low-frequency region. For the RPS time series acquired by GP, even in the high-frequency region, the distribution is not similar to that of the random case (i.e., a normal distribution).

These are reasonable results considering that the strategy inferred by GP comprises a finite combination of functions (Table 1) and does not include stochastic elements. However, although the strategy comprises simple functions, some individuals have generated series with randomness that was comparable to that of the human RPS series; thus, the result does not negate the validity of strategy inference via GP. Note that the performance of GP in the optimal solution search is strongly dependent on the functions used for each individual. Thus, we only present the results obtained under certain conditions. Thus, the results suggest that GP can be used effectively to infer a strategy based on human RPS time-series data. In future, it would be interesting to consider improving the performance of GP.

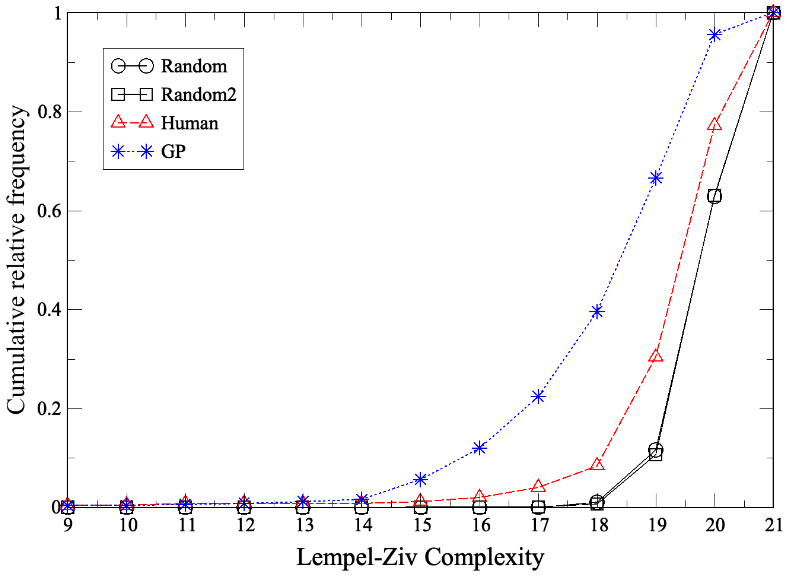

Figure 6. Cumulative relative frequency distribution of Lempel–Ziv complexity obtained from the set of RPS time series. The RPS time series were generated by a pseudorandom set, human, and the strategy inferred by GP. Here, “Random” and “Random2” show the results from the sets of 500 and 10^8 pseudorandom series generated using the Mersenne Twister, respectively.

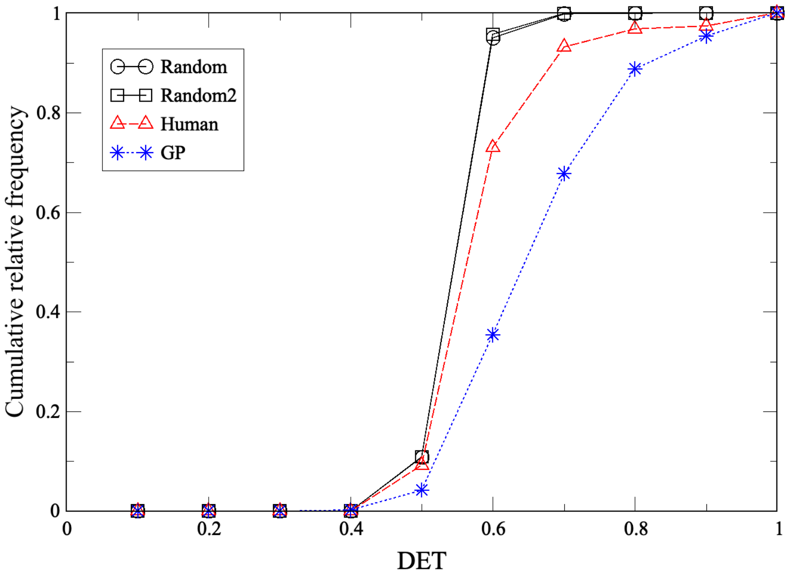

Figure 7. Cumulative relative frequency distribution of DET from recurrence plot obtained from the set of RPS time series. The RPS time series were generated by pseudorandom set, human, and the strategy inferred using GP. Here, “Random” and “Random2” show the results from the sets of 500 and 10^8 pseudorandom series generated using the Mersenne Twister, respectively.

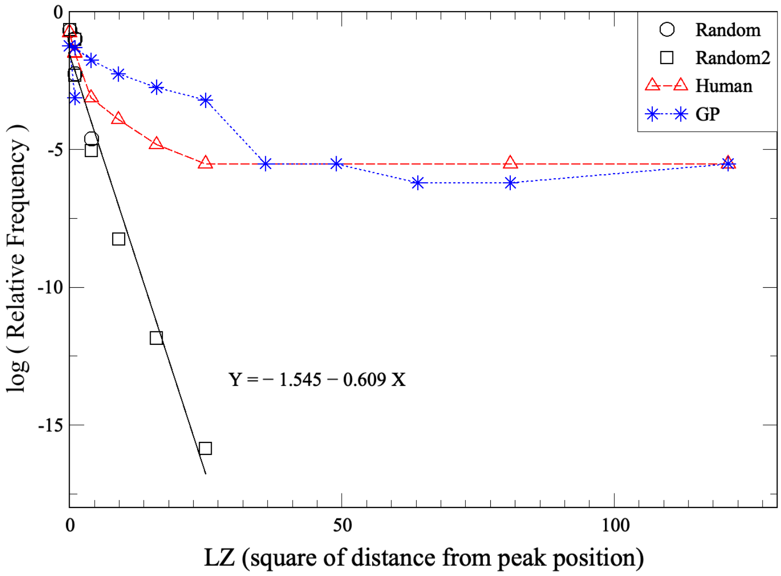

Figure 8. Relative frequency distribution of Lempel–Ziv complexity obtained from the set of RPS time series. The data are the same as those in Figure 6.

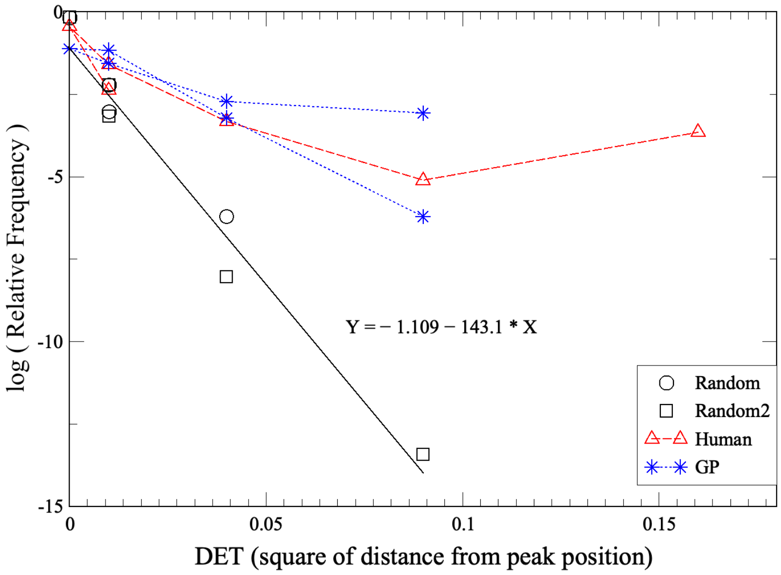

Figure 9. Relative frequency distribution of DET from recurrence plot obtained from the set of RPS time series. The data are the same as those in Figure 7.

4. Discussion and Conclusions

In this study, we evaluated the randomness of the time series of human hand signs in the RPS game. Here, we compared the complexity and deterministic characteristics of the RPS time-series data collected from 500 subjects with an equal number of pseudorandom RPS series sets. We also evaluated the complexity and determinism of the output time series from an RPS strategy inferred via genetic programming using the human RPS time-series data. Here, Lempel–Ziv complexity was used as an index to complexity, and DET obtained from a recurrence plot was used as an index to evaluate determinism.

From observation of RPS time series data alone, it was difficult to correctly classify them into pseudorandom series, human RPS time series, and time series generated from the GP's trained strategy models. However, by evaluating the complexity and determinism of each set of 500 RPS series, we found that each set (i.e., pseudorandom series, human RPS series, and GP RPS series) exhibited clearly different statistical distribution. In other words, the results quantitatively corroborate the above intuitively recognized properties.

The subjects were informed of the purpose of the experiment when the RPS time-series data were collected, and the goal was to win the RPS game. Thus, this result evaluates the randomness of the RPS series generated by the human system, including the RPS strategy to win the RPS game, and it does not universally evaluate the ability of the human system to generate random series. Furthermore, although the 500 RPS time series data collected in this study were provided by 500 different individuals, different results would be expected if a set of 500 time-series data generated by one person was used. Thus, many meaningful assumptions can be made about the collection of human RPS series, and the results in this study are only considered from one aspect. Thus, in future, we plan to clarify the characteristics of human time series generation under various conditions.

Human-generated randomness can have essential significance in terms of human behavior because randomness contributes to eliminating bias in fair decision making and accurate estimates of probability events. This randomness is also strongly related to the diversity of human behavior and exploratory behavior; however, humans may possess unconscious knowledge about the limits of human-generated randomness. This is evident in the fact that people have used objects and natural phenomena to apply randomness in gambling since ancient times [42]. In terms of search methods, for example, some results demonstrate that methods that consider chaos, which is a deterministic phenomenon, give more efficient search results than the method with randomness [43,44]. Certain features of near randomness but not belonging to randomness allow for the creation of special functions, such as efficient search. We believe that this study provides possible findings that could be the basis for a series of future studies evolving from this perspective.

In this paper, we considered AI-based strategy inference using only genetic programming. However, the strategies inferred in this study were defined by a procedure that does not include a stochastic component; thus, we assume that different results could be obtained by including stochastic elements in the strategy component. In addition, in terms of the evaluation function, the GP used in this study only considered the similarity of series as the evaluation function. However, we consider that evaluation via homology, which considers deviations in patterns, is more suitable for strategy evaluation. Although improving the performance of strategy inference methods was beyond the scope of this study, it is an interesting subject that has been studied extensively to date, and we would like to explore this area in the future based on the results presented in this paper.

Author Contributions: Conceptualization, T.K., H.K. and S.-J.K.; methodology, T.K., H.K. and S.-J.K.; software, T.K.; validation, T.K., H.K. and S.-J.K.; formal analysis, T.K., H.K. and S.-J.K.; investigation, T.K., H.K. and S.-J.K.; resources, T.K. and H.K.; data curation, T.K.; writing—original draft preparation, T.K.; writing—review and editing, T.K., H.K. and S.-J.K.; visualization, T.K.; supervision, S.-J.K.; project administration, H.K. and S.-J.K.; funding acquisition, H.K. and S.-J.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the research grant SP003 from SOBIN Institute LLC.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used in this study is available at https://github.com/kuro-lab/RPSdata (accessed on 22 November 2022).

Acknowledgments: This study was performed under the Cooperative Research Project (No. R02/A31) of the Research Institute of Electrical Communication, Tohoku University.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

RPS	Rock-Paper-Scissors
DET	DETerminism
GA	Genetic Algorithm
GP	Genetic Programming

Appendix A. Lempel–Ziv Complexity

Lempel–Ziv complexity is an index obtained based on methods such as LZ77, which is a data compression algorithm. In data compression algorithms like LZ77, the sequence is encoded by recording the first-look sequence pattern in a dictionary in order from the beginning of the entire sequence data. Here, the partial sequence recorded in the dictionary is referred to as a word, and the number of words recorded in the dictionary is defined as the complexity of the series. The pseudocode to calculate Lempel–Ziv complexity is given in Algorithm A1.

Algorithm A1 pseudocode to calculate Lempel–Ziv complexity

```
▷ $S$ is target sequence.
▷ $N$ is the number of elements in the $S$.
▷ $l$ is length of subsequence.
▷ $sub_l$ is a subsequence of length $l$ from the initial element of $S$.
▷ $D$ is a set of subsequence.
$l \leftarrow 1$
while $N <= 0$ OR $N < l$ do
    if $sub_l$ is the first-look sequence pattern then
        Add $sub_l$ to $D$                    ▷ Record the "word" in the dictionary.
        Remove $sub_l$ from $S$
        Update the value of N
        $l \leftarrow 1$
    else
        Increment $l$
    end if
end while
return The number of elements in $D$
```

For example, in the case of series [0,1,0,1,1,0,1,1,1,0,1,1,0,1,1,0,1], the number of words registered in the dictionary is seven, i.e., the Lempel–Ziv complexity is calculated as having a value seven. In this study, Lempel–Ziv complexity was calculated for the RPS time series; thus, the above procedure was applied to a sequence with three elements. For sequences with a length of 50 and three elements, Lempel–Ziv complexity (C_{LZ}) takes a value in range $9 \leq C_{LZ} \leq 21$. It can be seen that the larger the calculated Lempel–Ziv complexity, the more different patterns are in the series, and the smaller the Lempel–Ziv complexity, the more similar patterns are included.

Appendix B. DET from Recurrence Plot and Application to RPS Time Series

Generally, to obtain a recurrence plot from time-series data, the first step is to convert the time-series data to a reconstructed state space vector. Assuming that the time-series data is represented as $x_t(t = 1, \ldots .n)$, the dimension of the reconstructed state space is m, and the delay time is τ. Thus, the reconstructed state space vector X_t is obtained as follows:

$$X_t = \{x_t, x_{t+\tau}, \ldots, x_{t+(m+1)\tau}\}. \tag{A1}$$

The next step is to calculate the Euclidean distance $D_{i,j}$ between the two points of this reconstructed state space vector X_t, where $D_{i,j}$ is defined as follows:

$$D_{i,j} = \|X_i - X_j\|. \tag{A2}$$

Eventually, the recurrence plot is obtained using this $D_{i,j}$. The recurrence plot is an $N \times N$ pixels image, where each pixel takes a value of 0 or 1, i.e., white or black. Now, with $i = 1, \ldots, N$ on the horizontal axis and $j = 1, \ldots, N$ on the vertical axis, the pixel value $RP_{i,j}$ at the coordinates (i, j) of the recurrence plot is calculated as follows:

$$RP_{i,j} = \begin{cases} 1, & if\ D_{i,j} \le \theta \\ 0, & otherwise \end{cases}. \tag{A3}$$

Here, parameter θ defines the proximity of X_i and X_j. Note that parameter θ is set empirically according to the target problem.

Examples of recurrence plots can be found in many previous studies [37–41]. Generally, highly regular patterns appear for time series derived from periodic functions, and irregular patterns appear for highly random time series. In the case of seemingly irregular time series generated from deterministic systems, e.g., a chaotic time series, a partially regular pattern will appear in the recurrence plot. These properties of recurrence plots are quantified by DET and used as a measure of determinism.

DET is calculated as the ratio of the number of points forming a diagonal line to the total number of points in the recurrence plot. Given that the 45-degree diagonal line appears when the distance between two points is continuously close, the higher the deterministic property of the time series, the higher the DET value. Here, $P(l)$, i.e., the number of the diagonal lines of length $l(l \ge 1)$ in the recurrence plot of length N on a side, is expressed as follows:

$$P(l) = \begin{cases} \sum_{j\ge2}^{N-1}\left(1 - RP_{i+l,j+l}\right)\prod_{k=0}^{l-1} RP_{i+k,j+k} & if\ i = 1 \\ \sum_{i\ge2}^{N-1-l}\sum_{j\ge i+1}^{N-l}\left(1 - RP_{i-1,j-1}\right)\left(1 - RP_{i+l,j+l}\right)\prod_{k=0}^{l-1} RP_{i+k,j+k} & if\ i \ge 2 \end{cases}. \tag{A4}$$

Note that the recurrence plot is line symmetrical with the diagonal line; thus, the number of diagonal lines is calculated in the range $i < j$ in Equation (A4) to simplify the calculation.

Using $P(l)$, DET can be calculated as follows:

$$DET = \frac{\sum_{l\le l_{min}}^{N-1} lP(l)}{\sum_{l\le1}^{N-1} lP(l)}. \tag{A5}$$

The diagonal line is always drawn as a continuous line; thus, it is excluded from the calculation by setting the upper limit of the sum of Equation (A5) to $N - 1$. In addition, parameter l_{min} represents the minimum length of the diagonal line and is determined empirically.

We generate a recurrence plot from the RPS time series to evaluate the deterministic property of the human RPS time series. Here, the target time series is a type of point process data; thus, we assumed that the dimension of the reconstructed state space, m, is one. Thus, the recurrence plot is generated using the values of the RPS time series without modification. In other words, X_t in Equation (A1) is the t-th hand sign (i.e., rock, scissors, paper = 0, 1, 2) in the RPS time series, and the length of a side in the recurrence plot is equal to the length of the RPS time series. Then, let $X_i = X_j$ be the condition for $RP_{i,j} = 1$ as shown in Equation (A6). This is equivalent to setting parameter $\theta = 0$ in Equation (A3).

$$RP_{i,j} = \begin{cases} 1, & if\ X_i = X_j \\ 0, & otherwise \end{cases}. \tag{A6}$$

For example, the recurrence plot obtained from the RPS series [1,0,2,1,0,2,1,0,1,2,1,0,2,0,1,2,1,0,0,1,0,0,2,1,1,2,0,2,1,0,2,1,0,2,1,1,0,2,1,1,1,0,1,2,1,0,1,2,1,2] is shown in Figure A1. Since several short sequences such as [1,0,2] or [0,2,1] appear repeatedly in this series, a line with an angle of 45 degrees appears in the recurrence plot, and it is evaluated as a series with relatively high determinism. In this example, the DET is calculated as 0.652709. As demonstrated by this example, the recurrence plot can visualize the characteristics of the RPS time series.

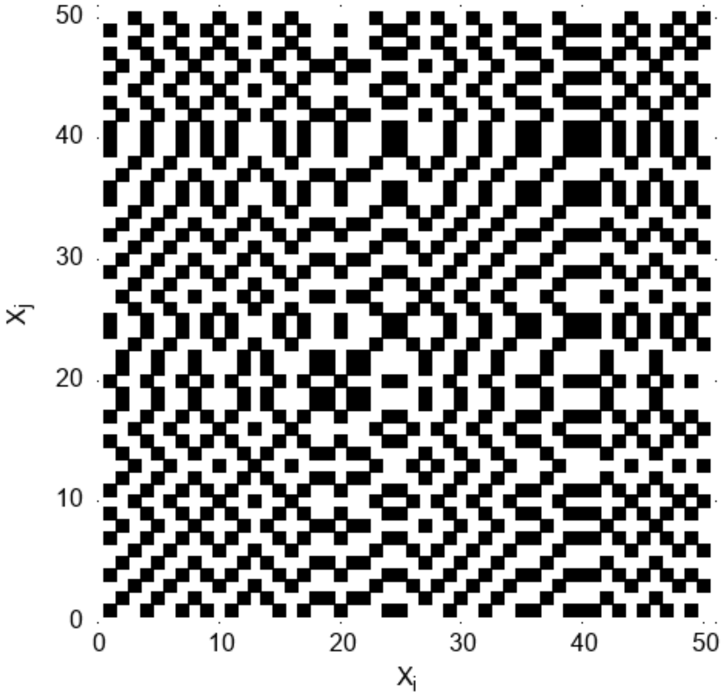

Figure A1. Example recurrence plot for RPS time series [1,0,2,1,0,2,1,0,1,2,1,0,2,0,1,2,1,0,0,1,0,0,2,1,1,2,0,2,1,0,2,1,0,2,1,1,0,2,1,1,1,0,1,2,1,0,1,2,1,2].

References

1. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet classification with deep convolutional neural networks. In Proceedings of the NeurIPS 2012, Lake Tahoe, NV, USA, 3 December 2012.
2. Schulz, H.; Behnke, S. Deep Learning. *Künstl. Intell.* **2012**, *26*, 357–363. [CrossRef]
3. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Nets. In Proceedings of the NeurIPS 2014, Montreal, QC, Canada, 9 December 2014.
4. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; The MIT Press: Cambridge, MA, USA, 2016.
5. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
6. Is Artificial Intelligence Set to Become art's Next Medium? Available online: https://www.christies.com/features/a-collaboration-between-two-artists-one-human-one-a-machine-9332-1.aspx (accessed on 5 May 2022).
7. Here's DALL-E: An Algorithm Learned to Draw Anything You Tell It. Available online: https://www.nbcnews.com/tech/innovation/here-s-dall-e-algorithm-learned-draw-anything-you-tell-n1255834 (accessed on 5 May 2022).
8. Hodassman, S.; Vardi, R.; Tugendhaft, Y.; Goldental, A.; Kanter I. Efficient dendritic learning as an alternative to synaptic plasticity hypothesis. *Sci. Rep.* **2022**, *12*, 6571. [CrossRef]
9. Shen, G.; Zhao, D.; Zeng, Y. Backpropagation with biologically plausible spatiotemporal adjustment for training deep spiking neural networks. *Patterns* **2022**, *3*, 100522. [CrossRef] [PubMed]
10. Colbrooka, M.J.; Antunb, V.; Hansena, A. C. The difficulty of computing stable and accurate neural networks: On the barriers of deep learning and Smale's 18th problem. *Proc. Natl. Acad. Sci. USA* **2022**, *119*, e2107151119. [CrossRef]
11. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **2016**, *529*, 484–489. [CrossRef]
12. AlphaGo, The Challenge Match. Available online: https://www.deepmind.com/research/highlighted-research/alphago/the-challenge-match (accessed on 5 May 2022).
13. Kim, S.-J.; Takahashi, T.; Sano, K. A balance for fairness: Fair distribution utilising physics. *Humanit. Soc. Sci. Commun.* **2021**, *8*, 131. [CrossRef]

14. Kim, S.-J.; Takahashi, T. Performance for Multi-armed Bandit Tasks Depending on Ambiguity-Preference of Learning Algorithm. *Front. Appl. Math. Stat.* **2018**, *4*, 27. [CrossRef]
15. Taleb, N.N. *Fooled by Randomness: The Hidden Role of Chance in Life and in the Markets*; Penguin Books: London, UK, 2007.
16. The Official History of Rock Paper Scissors. Available online: https://wrpsa.com/the-official-history-of-rock-paper-scissors (accessed on 5 May 2022).
17. Kako, S. *Densho Asobi Kou 4 Janken Asobi Kou*; Komine Shoten: Tokyo, Japan, 2008. (In Japanese)
18. van den Nouweland, A. Rock-paper-scissors; a new and elegant proof. *Econ. Bull.* **2007**, *3*, 1–6.
19. Batzilis, D.; Jaffe, S.; Levitt, S.; List, J.A.; Picel, J. Behavior in Strategic Settings: Evidence from a Million Rock-Paper-Scissors Games. *Games* **2019**, *10*, 18. [CrossRef]
20. Brockbank, E.; Vul, E. Humans fail to outwit adaptive rock, paper, scissors opponents. *Proc. Annu. Meet. Cogn. Sci. Soc.* **2021**, *43*, 1740–1746.
21. Wang, Z.; Xu, B.; Zhou, H.J. Social cycling and conditional responses in the Rock-Paper-Scissors game. *Sci. Rep.* **2014**, *4*, 5830. [CrossRef] [PubMed]
22. Rock, Paper, Scissors | Kaggle. Available online: https://www.kaggle.com/competitions/rock-paper-scissors (accessed on 5 May 2022).
23. Rock Paper Scissors Programming Competition. Available online: http://www.rpscontest.com (accessed on 5 May 2022).
24. Nippon Janken Kyoukai (Japan Jan-ken Association). Available online: https://japan-rps.jimdofree.com (accessed on 5 May 2022). (In Japanese)
25. World Rock Paper Scissors Association. Available online: https://wrpsa.com (accessed on 5 May 2022).
26. Yu, Q.; Fang, D.; Zhang, X.; Jin, C.; Ren, Q. Stochastic Evolution Dynamic of the Rock-Scissors-Paper Game Based on a Quasi Birth and Death Process. *Sci. Rep.* **2016**, *6*, 28585. [CrossRef]
27. Wang, L.; Huang, W.; Li, Y.; Evans, J.; He, S. Multi-AI competing and winning against humans in iterated Rock-Paper-Scissors game. *Sci. Rep.* **2020**, *10*, 13873. [CrossRef] [PubMed]
28. Hajihashemi, M.; Aghababaei, S.K. Multi-strategy evolutionary games: A Markov chain approach. *PLoS ONE* **2022**, *17*, e0263979. [CrossRef]
29. Aphiratsakun, N.; Blake, X.J.; Tin, K.K.; Ngwe, T. AI-based Rock-Paper-Scissors plug and play system. In Proceedings of the 2020 5th International STEM Education Conference, Hua Hin, Thailand, 4–6 November 2020; pp. 30–33.
30. Yuge, T.; Shirai, H.; Nishino, J.; Odaka, T.; Ogura, H. Evolutional Acquisition of a Strategy Using Genetic Programming. *Mem. Fac. Eng. Univ.* **2001**, *49*, 129–139. (In Japanese)
31. Komai, T.; Kim, S.-J.; Kurokawa, H. Characteristic extraction method of human's strategy in the Rock-Paper-Scissors game. In Proceedings of the 2018 RISP International Workshop on Nonlinear Circuits, Communications and Signal Processing, Honolulu, HI, USA, 4–7 March 2018; pp. 592–595.
32. Komai, T.; Kim, S.-J.; Kousaka, T.; Kurokawa, H. A Human Behavior Strategy Estimation Method Using Homology Search for Rock-Scissors-Paper Game. *J. Signal Process.* **2019**, *23*, 177–180. [CrossRef]
33. Bédard-Couture, R.; Kharma, N.N. Playing Iterated Rock-Paper-Scissors with an Evolutionary Algorithm. In Proceedings of the 11th International Joint Conference on Computational Intelligence, Vienna, Austria, 17–19 September 2019; pp. 205–212.
34. Lempel, A.; Ziv, J. On the Complexity of Finite Sequences. *IEEE Trans. Inf. Theory* **1976**, *22*, 75–81. [CrossRef]
35. Kim, S.-J.; Umeno, K.; Hasegawa, A. On the NIST Statistical Test Suite for Randomness. *Tech. Rep. IEICE* **2003**, *ISEC2003-87*, 21–27.
36. Matsumoto, M.; Nishimura, T. Mersenne twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. *ACM Trans. Model. Comput. Simul.* **1998**, *8*, 3–30. [CrossRef]
37. Eckmann, J.P.; Kamphorst, S.O.; Ruelle, D. Recurrence plots of dynamical systems. *Europhys. Lett.* **1987**, *4*, 973–977. [CrossRef]
38. Zbilut, J.P.; Webber, C.L. Embeddings and delays as derived from quantification of recurrence plots. *Phys. Lett. A* **1992**, *171*, 199–203. [CrossRef]
39. Webber, C.L., Jr.; Zbilut, J.P. Dynamical assessment of physiological systems and states using recurrence plot strategies. *J. Appl. Physiol.* **1994**, *76*, 965–973. [CrossRef] [PubMed]
40. Marwan, N.; Romano, M.C.; Thiel, M.; Kurths, J. Recurrence plots for the analysis of complex systems. *Phys. Rep.* **2007**, *438*, 237–329. [CrossRef]
41. Hirata, Y. Recurrence plots for characterizing random dynamical systems. *Commun. Nonlinear Sci. Numer. Simul.* **2021**, *94*, 105552. [CrossRef]
42. Hacking, I. *The Emergence of Probability*; Cambridge University Press: Cambridge, UK, 2006.
43. Kim, S.-J.; Naruse, M.; Aono, M.; Hori, H.; Akimoto, T. Random Walk with Chaotically Driven Bias. *Sci. Rep.* **2016**, *6*, 38634. [CrossRef]
44. Pedro Castro-Rodrigues, P.; Akam, T.; Snorasson, I.; Camacho, M.; Paixão, V.; Maia, A.; Barahona-Corrêa, J.B.; Dayan, P.; Simpson, H.B.; Costa, R.M.; et al. Explicit knowledge of task structure is a primary determinant of human model-based action. *Nat. Hum. Behav.* **2022**, *6*, 1126–1141. [CrossRef]