frontiers in Applied Mathematics and Statistics

ORIGINAL RESEARCH
published: 25 July 2018
doi: 10.3389/fams.2018.00027

Performance in Multi-Armed Bandit Tasks in Relation to Ambiguity-Preference Within a Learning Algorithm

Song-Ju Kim[1]* *and Taiki Takahashi*[2]

[1] Graduate School of Media and Governance, Keio University, Fujisawa, Japan, [2] Department of Behavioral Science, Research and Education Center for Brain Sciences, Center for Experimental Research in Social Sciences, Hokkaido University, Sapporo, Japan

Ellsberg paradox in decision theory posits that people will inevitably choose a known probability of winning over an unknown probability of winning even if the known probability is low [1]. One of the prevailing theories that addresses the Ellsberg paradox is known as "ambiguity-aversion." In this study, we investigated the properties of ambiguity-aversion in four distinct types of reinforcement learning algorithms: ucb1-tuned [2], modified ucb1-tuned, softmax [3], and tug-of-war [4, 5]. We took the following scenario as our sample, in which there were two slot machines and each machine dispenses a coin according to a probability that is generated by its own probability density function (PDF). We then investigated the choices of a learning algorithm in such multi-armed bandit tasks. There were different reactions in multi-armed bandit tasks, depending on the ambiguity-preference in the learning algorithms. Notably, we discovered a clear performance enhancement related to ambiguity-preference in a learning algorithm. Although this study does not directly address the issue of ambiguity-aversion theory highlighted in Ellsberg paradox, the differences among different learning algorithms suggest that there is room for further study regarding the Ellsberg paradox and the decision theory.

Keywords: decision making, Ellsberg paradox, ambiguity aversion, reinforcement learning, machine learning, artificial intelligence, natural computing, neuroeconomics

OPEN ACCESS

Edited by:
Hyeng Keun Koo,
Ajou University, South Korea

Reviewed by:
Simon Grima,
University of Malta, Malta
Fabrizio Maturo,
Università degli Studi G. d'Annunzio Chieti e Pescara (UNICH), Italy

***Correspondence:**
Song-Ju Kim
songju@sfc.keio.ac.jp

Specialty section:
This article was submitted to Mathematical Finance, a section of the journal Frontiers in Applied Mathematics and Statistics

Received: 20 April 2018
Accepted: 14 June 2018
Published: 25 July 2018

Citation:
Kim S-J and Takahashi T (2018) Performance in Multi-Armed Bandit Tasks in Relation to Ambiguity-Preference Within a Learning Algorithm. Front. Appl. Math. Stat. 4:27. doi: 10.3389/fams.2018.00027

1. INTRODUCTION

Recently, neuroeconomics has been developing into an increasingly important academic discipline that helps to explain human behavior. Ellsberg paradox is a crucial topic in neuroeconomics, and researchers have employed various theories to approach and to resolve the paradox. The basic concept behind the Ellsberg paradox is that people will always choose a known probability of winning over an unknown probability of winning, even if the known probability is low and the unknown probability could be a near guarantee of winning.

Let us start with an example. Suppose we have an urn that contains 30 red balls and 60 other balls that are either black or yellow. We do not know how many black or yellow balls are there, but we know that the total number of black balls plus the total number of yellow balls equals 60. The balls are well mixed so that each individual ball is as likely to be drawn as any other.

You are now given a choice between two gambles:

[Gamble A] You receive $100 if you draw a red ball,
[Gamble B] You receive $100 if you draw a black ball.

In addition, you are given the choice between these two gambles (about a different draw from the same urn):

[Gamble C] You receive $100 if you draw a red or yellow ball,
[Gamble D] You receive $100 if you draw a black or yellow ball.

Participants are tempted to choose [Gamble A] and [Gamble D]. However, these choices violate the postulates of subjective expected utility [1].

It is well known that ambiguity-aversion property of decision-making is one of the prevailing theories advanced to explain this paradox. On the other hand, reinforcement learning algorithms, such as ucb1-tuned [2], modified ucb1-tuned, softmax [3], and tug-of-war dynamics [4, 5], have been employed in multiple approaches in artificial intelligence (AI) applications. There is tremendous potential for neuroeconomic studies to investigate the properties of decision-making through the use of AI (learning) algorithms. This study is the first attempt to investigate the properties of learning algorithms with regards to the ambiguity-preference point of view.

In this study, we took a multi-armed bandit problem (MAB) as a decision-making problem. We considered two slot machines A and B. Each machine gave rewards with individual probability density function (PDF) whose mean and standard deviations were μ_A (μ_B) and σ_A (σ_B), respectively. The player makes a decision on which machine to play at each trial, trying to maximize the total reward obtained after repeating several trials. The MAB is used to determine the optimal strategy for finding the machine with the highest rewards as accurately and quickly as possible by referring to past experiences. The MAB is related to many application problems in diverse fields, such as communications (cognitive networks [6, 7]), commerce (internet advertising [8]), and entertainment (Monte Carlo tree search techniques in computer game programs [9, 10]).

In this study, we focused on limited MAB cases. Machine A has constant probability 1/3, and machine B has probabilities generated by normal distribution $N(\frac{1}{3} + \Delta\ \mu, \sigma^2)$. Here, we hypothesize that the total rewards from probabilities generated by a PDF is the same as the total rewards directly from the same PDF if we only focus on the average rewards using $1,000$ samples. On the basis of this hypothesis, we consider MABs, where PDFs are $\delta(\frac{1}{3})$ and $N(\frac{1}{3} + \Delta\ \mu, \sigma^2)$. Here, $\delta(x)$ is a delta function. In this study, "ambiguity" is expressed by σ although "ambiguity" becomes "risk" if our hypothesis does not hold.

2. LEARNING ALGORITHMS

2.1. Ambiguity-Neutral: SOFTMAX Algorithm

SOFTMAX algorithm is a well-known algorithm for solving MABs [3]. In this algorithm, the selecting probability of A or B, $P'_A(t)$ or $P'_B(t)$, is given by the following Boltzmann distributions:

$$P'_A(t) = \frac{\exp[\beta \cdot Q_A(t)]}{\exp[\beta \cdot Q_A(t)] + \exp[\beta \cdot Q_B(t)]}, \tag{1}$$

$$P'_B(t) = \frac{\exp[\beta \cdot Q_B(t)]}{\exp[\beta \cdot Q_A(t)] + \exp[\beta \cdot Q_B(t)]}, \tag{2}$$

where $Q_k(t)$ ($k \in \{A, B\}$) is given by $\frac{\sum_{j=1}^{N_k(t)} R_k(j)}{N_k(t)}$. Here, $N_k(t)$ is the number of selections of machine k until time t and $R_k(j)$ is the reward from machine k at time j. β is a time-dependent form in our study, which is as follows:

$$\beta(t) = \tau \cdot t. \tag{3}$$

$\beta = 0$ corresponds to a random selection and $\beta \rightarrow \infty$ corresponds to a greedy action. The SOFTMAX algorithm is "ambiguity-neutral" because "ambiguity" σ is not used in the algorithm.

2.2. Ambiguity-Neutral: Tug-Of-War Dynamics

In the tug-of-war (TOW) dynamics, a machine that has larger X_k ($k \in \{A, B\}$) is played in each time [4, 5]. Displacement X_A $(= -X_B)$ is determined by the following equations:

$$X_A(t+1) = Q_A(t) - Q_B(t) + \xi(t), \tag{4}$$

$$Q_k(t) = \sum_{j=1}^{N_k(t)} (R_k(j) - K). \tag{5}$$

Here, $Q_k(t)$ is an "estimate" of information of past experiences accumulated from the initial time 1 to the current time t, $N_k(t)$ is the number of selections of machine k until time t, $R_k(j)$ is the reward from machine k at time j, $\xi(t)$ is an arbitrary fluctuation to which the body is subjected, and K is a parameter. Consequently, the TOW evolves according to a simple rule: in addition to the fluctuation, if machine k is played at each time t, $R_k - K$ is added to $X_k(t)$. The TOW is also "ambiguity-neutral" because "ambiguity" σ is not used in the algorithm.

2.3. Ambiguity-Preference: UCB1-Tuned Algorithm

In the UCB1-tuned algorithm, a machine that has larger "index" is played in each time [2].

Initialization: Play each machine once.

Loop: Play machine j that maximizes following index,

$$\overline{x}_j(t) + \sqrt{\frac{ln(n)}{n_j} min(\frac{1}{4}, V_j(n_j))}, \tag{6}$$

$$V_j(s) = (\frac{1}{s} \sum_{\tau=1}^{s} x_{j,\tau}^2) - \overline{x}_{j,s}^2 + \sqrt{\frac{2ln(t)}{s}}, \tag{7}$$

where $\overline{x}_j(t)$ is the average reward obtained from machine j, n_j is the number of times machine j has been played so far, and

n is the overall number of plays done so far. The UCB1-tuned algorithm has "ambiguity-preference" property because it selects high variance ("ambiguity") machines in the early stage.

2.4. Ambiguity-Aversion: Modified UCB1-Tuned Algorithm

In the modified UCB1-tuned algorithm, a machine that has larger "index" is played in each time. Compared to UCB1-tuned algorithm, the sign of the second term in the index becomes minus.

Initialization: Play each machine once.

Loop: Play machine j that maximizes following index,

$$\bar{x}_j(t) - \sqrt{\frac{ln(n)}{n_j} min(\frac{1}{4}, V_j(n_j))}, \tag{8}$$

$$V_j(s) = (\frac{1}{s}\sum_{\tau=1}^{s} x_{j,\tau}^2) - \bar{x}_{j,s}^2 + \sqrt{\frac{2ln(t)}{s}}, \tag{9}$$

where $\bar{x}_j(t)$ is the average reward obtained from machine j, n_j is the number of times machine j has been played so far, and n is the overall number of plays done so far. The UCB1-tuned algorithm has "ambiguity-aversion" property because it selects low variance ("ambiguity") machines in the early stage.

3. RESULTS

In this study, we focused on the following limited MAB cases. On the basis of the hypothesis, we considered MABs where PDF of machine A is $\delta(\frac{1}{3})$, and PDF of machine B is $N(\frac{1}{3} + \Delta\mu, \sigma^2)$, respectively. "Ambiguity" is expressed by σ.

For positive $\Delta\mu$, we investigate 30 cases where $\Delta\mu = 0.00$, 0.05, 0.10, 0.15, and 0.20, and $\sigma = 0.05$, 0.10, 0.15, 0.20, 0.25, and 0.30, respectively. **Figure 1** shows performance comparison between four learning algorithms for the MABs. The horizontal axis denotes $\Delta\mu$ (6 different σ cases for each either $\Delta\mu$). The vertical axis denotes total rewards (average of 1,000 samples) until time $t = 1,000$ (also see Appendix in Supplementary Material).

For positive $\Delta\mu$ cases, machine B is the correct selection because expected value of machine B is higher than A. This means that ambiguity-preference is needed for correct selections. The UCB1-tuned algorithm (ambiguity-preference) has higher performance than the modified UCB1-tuned algorithm (ambiguity-aversion) in the positive $\Delta\mu$ cases. Performance of the UCB1-tuned algorithms (ambiguity-preference) slightly increases as the ambiguity (σ) of the problems increases, whereas performance of the modified UCB1-tuned algorithms (ambiguity-aversion) largely decreases as ambiguity (σ) of the problems increases.

Performances of TOW and SOFTMAX are higher than those of UCB1-tuned and modified UCB1-tuned algorithms because each of the former two algorithms has a parameter that optimized the problems. That is, each of the two algorithms has an advantage over the latter two algorithms that have no parameter. Performances of the former two algorithms (ambiguity-neutral) slightly decrease as ambiguity (σ) of the problems increases. This is because incorrect decisions are slightly increased as estimation for mean value of rewards becomes largely fluctuated.

For negative $\Delta\mu$, we also investigated 30 cases where $\Delta\mu = 0.00$, 0.05, 0.10, 0.15, and 0.20, and $\sigma = 0.05$, 0.10, 0.15, 0.20, 0.25, and 0.30, respectively. **Figure 2** shows the performance comparison between four learning algorithms for the MABs. The horizontal axis denotes $\Delta\mu$ (6 different σ cases for each $\Delta\mu$). The vertical axis denotes total rewards (average of 1,000 samples) until time $t = 1,000$ (also see Appendix in Supplementary Material).

For negative $\Delta\mu$ cases, machine A is correct selection because expected value of machine A is higher than B. This means that ambiguity-aversion is needed for correct selections. The modified UCB1-tuned algorithm (ambiguity-aversion)

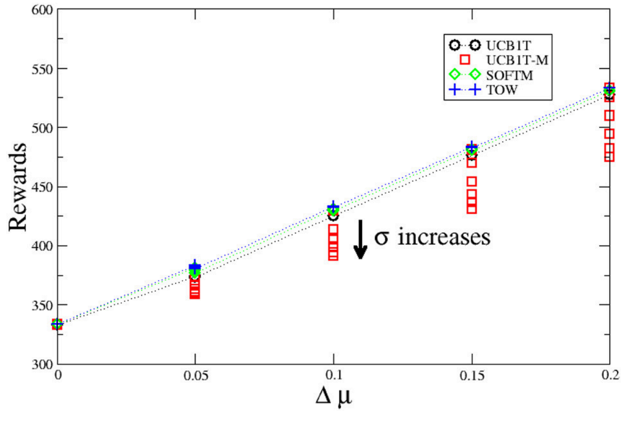

FIGURE 1 | Performance comparison between four learning algorithms for MAB where PDFs are $\delta(\frac{1}{3})$ and $N(\frac{1}{3} + \Delta\mu, \sigma^2)$. $\Delta\mu$ is positive (cases where machine B is correct decision).

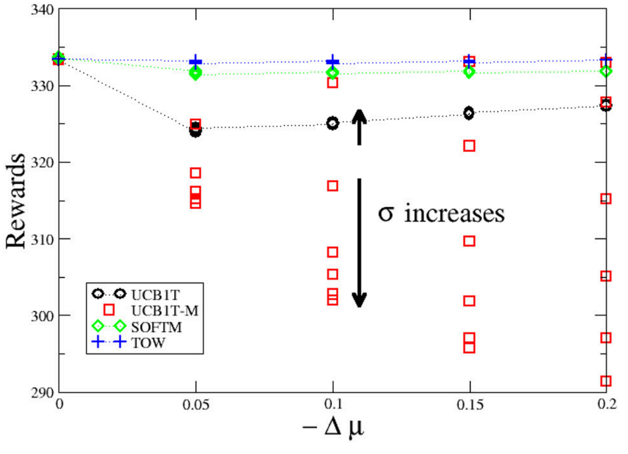

FIGURE 2 | Performance comparison between four learning algorithms for MAB where PDFs are $\delta(\frac{1}{3})$ and $N(\frac{1}{3} + \Delta\mu, \sigma^2)$. $\Delta\mu$ is negative (cases where machine A is correct decision).

has higher performance than the UCB1-tuned algorithm (ambiguity-preference) in the negative $\Delta\mu$ cases only in $\sigma = 0.05$. Performance of the UCB1-tuned algorithms (ambiguity-preference) slightly increases as ambiguity (σ) of the problems increases, whereas performance of the modified UCB1-tuned algorithms (ambiguity-aversion) largely decreases as ambiguity (σ) of the problems increases.

Performances of TOW and SOFTMAX are higher than those of UCB1-tuned and modified UCB1-tuned algorithms because each of the former two algorithms has a parameter that optimized the problems as well as the positive $\Delta\mu$ cases. Performances of the former two algorithms (ambiguity-neutral) also slightly decrease as the ambiguity (σ) of the problems increases because of the same reason as the positive $\Delta\mu$ cases.

4. CONCLUSION AND DISCUSSION

In both cases (positive $\Delta\mu$ and negative $\Delta\mu$), performance of the UCB1-tuned algorithms (ambiguity-preference) slightly increases as the ambiguity (σ) of the problems increases, whereas performance of the modified UCB1-tuned algorithms (ambiguity-aversion) largely decreases as the ambiguity (σ) of the problems increases. This means that ambiguity-aversion property of learning algorithm has a negative contribution to its performances for MABs, whereas ambiguity-preference has a positive contribution.

From these limited computer simulation results, we conclude that ambiguity-aversion property does not work for efficient decision-making in the learning point of view (repeated decision-making situations). Another point of view will be necessary for justification of ambiguity-aversion property. We suggest that the differences among learning algorithms require further study on the Ellsberg paradox and decision theory.

AUTHOR CONTRIBUTIONS

S-JK and TT designed research. S-JK performed computer simulations. S-JK and TT analyzed the data. S-JK wrote the manuscript. All authors reviewed the manuscript.

FUNDING

This work was supported in part by the Grant-in-Aid for Challenging Exploratory Research (15K13387) from the Japan Society for the Promotion of Science, and by the CREST (JPMJCR17A4) ID-17941861 from Japan Science and Technology Agency.

ACKNOWLEDGMENTS

We would like to thank Prof. Masashi Aono and Dr. Makoto Naruse for fruitful discussions in an early stage of this work.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fams.2018.00027/full#supplementary-material

REFERENCES

1. Ellsberg D. Risk, ambiguity, and the Savage axioms. *Q J Econ.* (1961) **75**:643–69.
2. Auer P, Cesa-Bianchi N, Fischer P. Finite-time analysis of the multiarmed bandit problem. *Mach Learn.* (2002) **47**:235–56. doi: 10.1023/A:1013689704352
3. Sutton R, Barto A. *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press (1998).
4. Kim S-J, Aono M, Nameda E. Efficient decision-making by volume-conserving physical object. *New J. Phys.* (2015) **17**:083023. doi: 10.1088/1367-2630/17/8/083023
5. Kim S-J, Tsuruoka T, Hasegawa T, Aono M, Terabe K, Aono M. Decision maker based on atomic switches. *AIMS Mater. Sci.* (2016) **3**:245–59. doi: 10.3934/matersci.2016.1.245
6. Lai L, Jiang H, Poor HV. Medium access in cognitive radio networks: a competitive multi-armed bandit framework. In: *Proceedings of IEEE 42nd Asilomar Conference on Signals, System and Computers* (California, CA) (2008). p. 98–102.
7. Lai L, Gamal HE, Jiang H, Poor HV. Cognitive medium access: exploration, exploitation, and competition. *IEEE Trans Mobile Comput.* (2011) **10**:239–53. doi: 10.1109/TMC.2010.65
8. Agarwal D, Chen BC, Elango P. Explore/exploit schemes for web content optimization. In: *Proceedings of ICDM2009* (2009). doi: 10.1109/ICDM.2009.52
9. Kocsis L, Szepesvári C. Bandit based Monte-Carlo planning. In: *ECML2006, LNAI 4212*. Berlin: Springer (2006). p. 282–93.
10. Gelly S, Wang Y, Munos R, Teytaud O. Modification of UCT with patterns in Monte-Carlo Go. In: *RR-6062-INRIA* France: Research Report (2006). p. 1–19.

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.