# ON RISK PREFERENCES OF BANDIT ALGORITHMS

ANDREAS HAUPT AND AROON NARAYANAN

## 1. Main idea and intuition

Learning algorithms are used widely in many economic settings today—their applications range from price setting to auctions. These algorithms learn about the economic environment while interacting within it, making them a useful tool to deploy in settings of incomplete information. As they are used in more and more settings of economic interest, it also becomes important to understand the economic implications of their use. For example, Calvano et al. 2019 show that pricing algorithms can learn to collude without being explicitly told to do so, achieving close to full monopoly output.

We are interested in the behavioral implications of using particular learning algorithms, primarily in terms of inherent risk preferences. To be more precise, suppose we provide the learning algorithm with two options—either pull lever A and get a certain payoff of 0, or pull lever B and get a stochastic payoff of either 1 or $-1$, distributed uniformly. At any point in time, contingent on past observations of payoffs from pulled action (the *bandit setting*), an algorithm specifies a distribution on actions it takes next.

We find that popular algorithms in use are generally not risk neutral. The main intuition for this phenomenon is several algoritms ($\varepsilon$-Greedy, UCB) undersample actions with low rewards. Riskier actions that get a low reward are "trapped" in this estimate of reward, as they undersample, for longer. This leads to, over long time spans of learning, that several algorithms we study, behave consistent with risk aversion.

The tractability of our analysis relies on the fact that many popular bandit algorithms rely on reward estimates for actions, i.e. an expected value for each of the actions. Several used learning algorithms such as $\varepsilon$-Greedy, EXP3, Thompson Sampling and Upper Confidence Band algorithms have such structure, compare Lattimore and Szepesvári 2020). This structure allows us to describe the learning dynamics of algorithms by one-dimensional stochastic processes, which, in the limit of long-time learning, converge to continuous-time stochastic processes, which in turn allows us to use the clean formalism of stochastic calculus.

Our reduction of learning dynamics to stochastic processes allows us to study the risk preferences of any learning algorithm. Understanding these inherent preferences are an important part of understanding how these algorithms function, and also important for deciding which algorithm to choose for different purposes.

## 2. Model

A decision maker repeatedly takes one of $k$ actions, which give her a stochastic payoff sampled identically and independently distributed from distributions $F_i \in \Delta(\mathbb{R}), i \in [k]$.

*Date*: July 2021.

The *strategy* or *algorithm* used by the decision maker can be abstractly represented by the function $\pi\colon ([k]\times[0,1])^* \to \Delta([k])$. Different algorithms imply different $\pi$.

A general algorithm takes the following form. For each $t \in N$, it chooses an action $A_t \sim \pi(A_1, r_1, A_2, r_2, ..., A_{t-1}, r_{t-1})$ and gets a reward $r_t \sim F_{A_t}$.

Some examples of such algorithms are as follows.

**Example** ($\epsilon$-Greedy)**.** *The $\epsilon$-Greedy algorithm chooses the empirically best action with probability $1-\epsilon$, and randomizes between all the actions with probability $\epsilon$. Thus the strategy function can be written as :*

$$\pi_i(A_1, r_1, A_2, r_2, ..., A_{t-1}, r_{t-1}) = \begin{cases} \frac{1-\varepsilon}{|\arg\max_i \sum_{t:A_t=i} r_i|} & \text{if } i \in \arg\max_i \frac{1}{|\{t|A_t=i\}|} \sum_{t:A_t=i} r_i \\ \epsilon & \text{otherwise.} \end{cases}$$

**Example** (EXP3)**.** *The EXP3 algorithm assigns a weight to each action, and then assigns the probability of choosing each action according to a weighted exponential. The weights are updated according to the observed rewards each period. Thus, the strategy function is:*

$$\pi_i(A_1, r_1, A_2, r_2, ..., A_{t-1}, r_{t-1}) = \frac{\exp w_{i,t-1}}{\sum_{j\in[k]} \exp w_{j,t-1}}$$

*where the weights $w$ evolve as:*

$$w_{i,t} = w_{i,t-1} + \mathbb{1}_{A_t=i} \frac{r_t}{\pi_i(A_1, r_1, A_2, r_2, ..., A_{t-1}, r_{t-1})}$$

*with $w_{i,0} \coloneqq 0$ for all $i \in [k]$.*

Several other algorithms, such as Upper Confidence Band (UCB) and Thompson Sampling, can also be captured by this formulation.

Since our purpose is to study the risk preferences of algorithms, we consider algorithms facing a choice between a risky and a less risky alternative. At discrete times $t \in \mathbb{N}$, the algorithm needs to make the choice between two actions. For ease of notation, we denote them by $n$ and $r$ instead of numbers. We assume that $F_r$ is a riskier choice than $n$, i.e. the centered $F_r$, $F_r - \mathbb{E}[F_r]$, is a mean-preserving spread of $F_n$.

For the sake of exposition, we will assume that $F_n$ gives reward 0 with certainty, $F_n = \delta_0$. We will relax this in an extension later. The risky action follows an arbitrary distribution $F_r$ that admits second moments. Denote the standard deviation of $F$ by $\sigma$, and its expectation by $\mu$.

In each round, the agent observes the reward from the chosen action (only). We are interested in the average frequency across time with which an agent chooses a risky action, i.e.

$$\mathbb{E}\left[\frac{1}{T}|\{t|A_t = r\}|\right].$$

We say that the algorithm *prefers* action $i$ if

$$\mathbb{E}\left[\frac{1}{T}|\{t|A_t = i\}|\right] > \frac{1}{2}$$

and that it is *indifferent* between $n$ and $r$ if

$$\mathbb{E}\left[\frac{1}{T}|\{t|A_t = r\}|\right] = \frac{1}{2}.$$

This nomenclature is motivated by the standard random choice model with Gaussian noise, where we know that, in a binary-choice setting, a probability of choice of an alternative is greater than $1/2$ if and only if the expected value of the action is preferred.

Finally, we call an algorithm *weakly risk-averse* resp. *weakly risk-affine* if for $\mu = 0$, the agent prefers the riskless resp. risky action.

## 3. Analysis

3.1. **Reduction to a Stochastic Process.** Assume that an agent's strategy is given by $\pi$. We can identify a bandit strategy for 2 actions by the probability of choosing the risky action, yielding a function $\rho$,

$$\rho\colon (\{1,2\} \times \mathbb{R})^* \to [0,1].$$

We can recover the probability of choosing the non-risky action as $\pi_n(A_1, r_1, \ldots, A_t, r_t) = 1 - \rho(A_1, r_1, \ldots, A_t, r_t)$.

Several important learning algorithms ($\varepsilon$-Greedy and EXP3 as we show in this note, as well as UCB and Thompson sampling) can be described as only depending on a one-dimensional quantity, which will take the role of a *reward difference estimate*.

**Definition** (self-reinforcing variance)**.** *Assume that a learning algorithm has a decomposition via a one-dimensional quantity:* $f\colon \mathbb{R} \to [0,1]$, $g\colon (\{1,2\} \times \mathbb{R})^* \to \mathbb{R}^+$

$$\rho(A_1, r_1, A_2, r_2, \ldots, A_t, r_t) = f(g(A_1, r_1, A_2, r_2, \ldots, A_t, r_t)).$$

*Denote* $X_t \coloneqq g(A_1, r_1, A_2, r_2, \ldots, A_t, r_t)$*. If*

- $X_0 = 0$*;*
- $(X_t - \mu t, \sigma(A_1, r_1, \ldots, A_t, r_t))$*, where* $\sigma$ *denotes the* $\sigma$*-algebra operator, is a martingale;*
- $g$ *is point-symmetric around* $(1/2, 1/2)$*, i.e.* $g(x) = 1 - g(-x)$*,* $x \in [0,1]$ *and increasing;*

*we call* $X_t$ *a* reward distance estimate *and the algorithm* adapted to a reward distance estimate*. If, in addition,*

$$\operatorname{Var}[X_{t+1} | X_t = x] \tag{1}$$

*is (strictly) increasing in* $x$*, then we say that the algorithm is (strictly) variance-increasing. If* (1) *is (strictly) decreasing in* $x$*, we say it is (strictly) variance-decreasing.*

We show that $\varepsilon$-Greedy is strictly variance-increasing and EXP3 is strictly variance-decreasing. We will use the notations

$$\hat{\mu}_{i,T} = \sum_{\substack{1 \le t \le T \\ A_t = i}} r_t$$

and

$$\tilde{\mu}_{i,T} = \sum_{\substack{1 \le t \le T \\ A_t = i}} \frac{r_t}{\pi(A_1, r_1, \ldots, A_{t-1}, r_{t-1})}$$

for the empirical action estimated and the importance-sampling weighted estimates, respectively.

**Example** ($\varepsilon$-Greedy)**.** *Define $X_t = \hat{\mu}_{r,t} - \hat{\mu}_{n,t} = \hat{\mu}_{r,t}$. $\varepsilon$-Greedy can be expressed with the function $g$*

$$g(x) = \frac{\varepsilon}{2} + (1-\varepsilon)\left(\mathbb{1}_{\mathbb{R}_{>0}}(x) + \frac{1}{2}\mathbb{1}_{\{0\}}(x)\right). \tag{2}$$

*$g$ is point-symmetric and clearly $X_0 = 0$. Also, $X_t - \mu$ is a martingale by well-known properties of statistical estimators. We can calculate the variance by hand. It is given by*

$$\mathrm{Var}[X_{t+1}|X_t = x] = \left(\frac{\varepsilon}{2} + (1-\varepsilon)\left(\mathbb{1}_{\mathbb{R}_{>0}}(x) + \frac{1}{2}\mathbb{1}_{\{0\}}(x)\right)\right)\sigma^2,$$

**Example** (EXP3)**.** *Set $X_t = \tilde{\mu}_{r,t} - \tilde{\mu}_{m,t} = \tilde{\mu}_{r,t}$. Also observe that $X_t = w_{r,t} - w_{n,t} = w_{r,t}$. We can then express*

$$g(x) = \frac{e^x}{1+e^x}.$$

*The function $g$, the sigmoid function, is well known to be point symmetric.*

*Clearly, $X_0 = 0$. Also, $X_t - \mu t$ is a martingale, which follows from the property that $A_t \sim \pi(A_1, r_1, \ldots, A_{t-1}, r_{t-1})$, or known properties of importance-sampling estimators. We can express the conditional variance as*

$$\mathrm{Var}[X_{t+1}|X_t = x] = \left(\frac{e^x}{1+e^x}\right)\frac{\sigma}{\left(\frac{e^x}{1+e^x}\right)^2} = \frac{\sigma^2}{\frac{e^x}{1+e^x}},$$

*which is decreasing in $x$.*

As we do not show in this note, UCB is variance-increasing and Thompson sampling is, depending on the likelihood model, weakly variance increasing, or strictly so.

We continue our analysis merely for the case of Markovian rules, i.e. rules in which $X_{t+1}$ is measurable with respect to $X_t$, $a_t$ and $r_t$. Examples of such rules are $\varepsilon$-Greedy and UCB. The following statement likely holds also more broadly. As we leave out some of the details, we list it as a conjecture.

**Conjecture 1.** *All Markovian strictly variance-increasing algorithms are weakly risk-averse. All Markovian strictly variance-decreasing algorithms are risk-affine.*

We continue our analysis for the, for our purposes, most tractable algorithm, $\varepsilon$-Greedy. This allows us to introduce mathematical techniques also used in the proof above.

3.2. **Further Analysis of $\varepsilon$-Greedy.** We will perform this analysis for $\varepsilon$-Greedy. The decision-maker prefers the risky action if and only if

$$\mathbb{E}\left[\lim_{T\to\infty}\frac{1}{T}\sum_{i=1}^{T} f(X_t)\right] > \frac{1}{2}.$$

**Conjecture 2.** *For $\varepsilon$-Greedy,*

$$\mathbb{E}\left[\lim_{T\to\infty}\frac{1}{T}\sum_{i=1}^{T} f(X_t)\right] = \int_0^1 f(X_t)\mathrm{d}t = \frac{\varepsilon}{2} + (1-\varepsilon)\int_0^1 \mathbb{1}_{\mathbb{R}_+}(Y_t)\mathrm{d}t$$

*where $Y_t$ is a strong solution to the stochastic differential equation*

$$\mathrm{d}Y_t = \mu\mathrm{d}t + g(Y_t)\mathrm{d}W_t, \tag{3}$$

*where $(W_t)_{t\geq 0}$ is Brownian motion.*

The theorem uses a generalization of Donsker's theorem, which we omit in this document.

**Corollary.** *We get as a result that the algorithm is random risk-averse if and only if*

$$\mathbb{E}\left[\int_0^1 \mathbb{1}_{\mathbb{R}_{>0}}(Y_t)\mathrm{d}W_t\right] > \frac{1}{2}$$

This is closely related to existing *arcsine laws* in probability. We state the result for the (unrealistic) case of $\varepsilon = 1$. While risk-neutrality in this case is trivial, this gives us techniques to move forward, using generalized laws Watanabe, K. Yano, and Y. Yano 2005:

**Lemma** (Arcsine Law)**.** $\int_0^1 \mathbb{1}_{\mathbb{R}_{>0}}(Y_t)\mathrm{d}W_t \sim G$*, where* $G$ *is the arcsine distribution with density* $\mathbb{1}_{[0,1]}\frac{2}{\pi}\arcsin(\sqrt{x})$*.*

In particular, as $\int_0^1 \frac{2}{\pi}\arcsin(\sqrt{x})x\mathrm{d}x = \frac{1}{2}$, a purely exploring agent is risk-neutral. We expose the arcsine law as a technique to analytically solve for quantities of relevance, which can complement simulations.

## 4. Simulations

Here we present some results on how the $\epsilon$-Greedy chooses between the actions. Figure 1 presents the exact unconditional probability of the algorithm choosing the risky action as the time progresses, up to 10 periods. What is striking is that the action becomes *less* likely over time to choose the risky action.

For Figure 2, we run $\epsilon$-Greedy 1000 times, with each time it running for 10000 periods. For each run, we keep track of the fraction of times that it chose the risky action. This gives us a distribution of the probability of the algorithm choosing the risky action. Note that the density leans heavily towards 0, which means that the algorithm is quite unlikely to choose the risky action. This can be seen as evidence for the risk aversion of $\varepsilon$-Greedy.

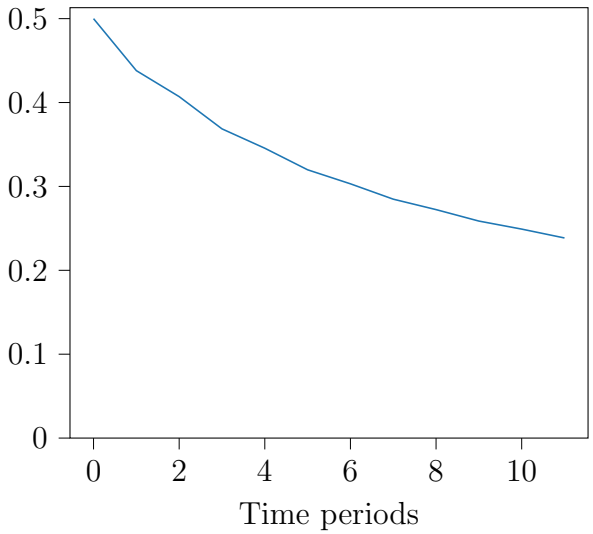


Figure 1. The exact probability over time of the risky action being chosen for $\epsilon$-Greedy with $\epsilon = 0.05$

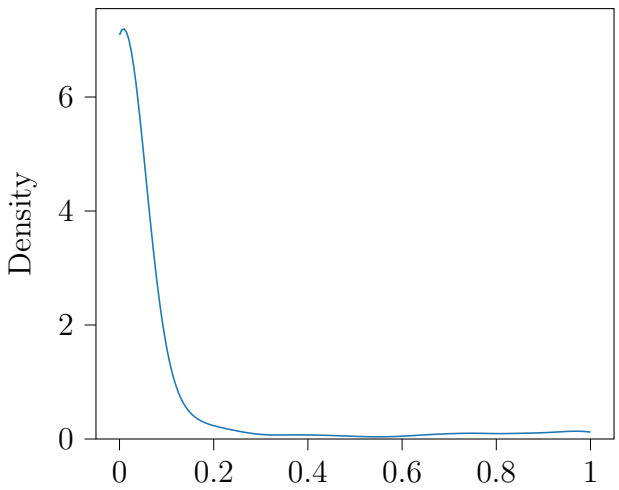


FIGURE 2. The simulated distribution of the probability of the risky action being chosen for $\epsilon$-Greedy with $\epsilon = 0.05$ and no bias in expected reward

## 5. Certainty Equivalents

Our analysis thus far allows us to find a safe payment $\mu$ the algorithm prefers for a given variance $\sigma^2$. This corresponds to the concept of the certainty equivalent.

5.1. **Analytical Formulas for Certainty Equivalents.** The standard notion of certainty equivalent captures the strength of risk preferences in rational agents. The essence of the certainty equivalent is capturing indifference between a lottery and a fixed amount of money, which we translate into our setting for algorithms. The certainty equivalent in this setting is the difference in mean reward between the two actions for which the algorithm would choose the risky action half the time on average. So when the risky reward is such that $E[R] = 0$, it is the value of $\mu$ such that the distribution of fraction of times the risky action is chosen has mean one-half.

While still speculative, generalized arcsine laws allow to write

$$\mathbb{E}\left[\int_0^1 \mathbb{1}_{\mathbb{R}_{>0}}(X_t)\mathrm{d}W_t\right] = \frac{1}{2},$$

depending on different values of $\mu$ and allow for finding expressions of $\mu$. As a last part of this note, we show how this can be used to identify a—not necessarily rationalizable—utility function for the agent.

5.2. **Recovering Utility Functions.** Having certainty equivalents allows us to identify utility functions for different algorithms. Note that von Neumann-Morgenstern utility functions are only identified up to affine transformations and hence we can set

$$u(0) = 0 \qquad\qquad u(1) = 1.$$

Assume now that for each level $\sigma^2$ and lottery $l$, we have a certainty equivalent $\mathrm{CE}(\sigma^2, l)$. For any $x$, choosing $l$ such that $\mathit{support}(l) \in \{0, 1\}$ and $E[l] = x$, we can then recover

$u^{-1}(x) = CE(\sigma^2, l)$. Since $u$ is strictly increasing, recovering $u^{-1}$ recovers $u$ as well. Assuming a formula for certainty equivalents allows us to recover a formula for utility functions as well.

This does not mean that algorithms are truly maximizing von Neumann-Morgenstern utility functions, it is possible that they are not consistent. This is an aspect for further study.

## 6. Extensions

While the proofs and techniques outlined in this document need much work, several extensions are possible: Extensions to several actions, actions that are ranked in second-order stochastic dominance, and an analysis for UCB and Thompson sampling-type algorithms.

## References

[1] Emilio Calvano et al. "Artificial Intelligence, Algorithmic Pricing and Collusion". 2019.
[2] Tor Lattimore and Csaba Szepesvári. "Bandit Algorithms". In: *Bandit Algorithms* (2020). DOI: 10.1017/9781108571401.
[3] Shinzo Watanabe, Kouji Yano, and Yuko Yano. "A density formula for the law of time spent on the positive side of one-dimensional diffusion processes". In: *Kyoto Journal of Mathematics* 45.4 (2005), pp. 781–806. ISSN: 0023608X. DOI: 10.1215/kjm/1250281657.

*Proof Idea.* First prove that the average probability of choosing the risky action across time is

$$\int_0^1 g(Y_t)\mathrm{d}t$$

where $Y_t$ is a solution to the stochastic differential equation

$$\mathrm{d}Y_t = \sigma(Y_t)\mathrm{d}W_t.$$

where $\sigma(x) = \mathrm{Var}[X_{t+1}|X_t = x]$. This is a non-trivial extension of both Donsker's theorem, as $X_{t+1}$ conditionally on $X_t$ is not normally distributed and the convergence analysis of the Euler-Maruyama method for the discretization of stochastic differential equations, because it is also not Rademacher distributed.

Next, we can use Ito's formula to observe that $Z_t = g(Y_t)$ is a strong solution to the stochastic differential equation

$$\mathrm{d}Z_t = \frac{1}{2}g''(Y_t)\sigma^2(Y_t)\mathrm{d}t + g'(Y_t)\sigma(Y_t)\mathrm{d}W_t,$$

which is in non-differential notation for $t = 1$ as $Z_0 = 1/2$,

$$Z_t = \frac{1}{2} + \int_0^t g''(Y_t)\mathrm{d}t + \int_0^t g'(Z_t)\mathrm{d}W_t.$$

$\int_0^1 Z_t\mathrm{d}t > \frac{1}{2}$ is equivalent to

$$\int_0^t g''(Y_t)\sigma^2(Y_t)\mathrm{d}t + \int_0^t g'(Z_t)\sigma(Y_t)\mathrm{d}W_t > 0.$$

Note that as $g'(Z_t) > 0$, we get that $\int_0^t g'(Z_t)\sigma(Y_t)\mathrm{d}W_t \geq 0$. What remains is to use that $g$ is point-symmetric, and hence $\int_0^t g''(Y_t)\sigma^2(Y_t)\mathrm{d}t$. This needs more careful analysis of $Y_t$. □