# Competing bandits: learning under competition*

Yishay Mansour [†] Aleksandrs Slivkins [‡] Zhiwei Steven Wu [§]

September 2017

**Abstract**

Most modern systems strive to learn from interactions with users, and many engage in *exploration*: making potentially suboptimal choices for the sake of acquiring new information. We initiate a study of the interplay between *exploration and competition*—how such systems balance the exploration for learning and the competition for users. Here the users play three distinct roles: they are customers that generate revenue, they are sources of data for learning, and they are self-interested agents which choose among the competing systems.

In our model, we consider competition between two multi-armed bandit algorithms faced with the same bandit instance. Users arrive one by one and choose among the two algorithms, so that each algorithm makes progress if and only if it is chosen. We ask whether and to what extent competition incentivizes the adoption of better bandit algorithms. We investigate this issue for several models of user response, as we vary the degree of rationality and competitiveness in the model. Our findings are closely related to the "competition vs. innovation" relationship, a well-studied theme in economics.

## 1 Introduction

Learning from interactions with users is ubiquitous in modern customer-facing systems, from product recommendations to web search to spam detection to content selection to fine-tuning the interface. Many systems purposefully implement *exploration*: making potentially suboptimal choices for the sake of acquiring new information. Randomized controlled trials, a.k.a. A/B testing, are an industry standard, with a number of companies such as *Optimizely* offering tools and platforms to facilitate them. Many companies use more sophisticated exploration methodologies based on *multi-armed bandits*, a well-known theoretical framework for exploration and making decisions under uncertainty.

Systems that engage in exploration typically need to compete against one another; most importantly, they compete for users. This creates an interesting tension between *exploration* and *competition*. In a nutshell, while exploring may be essential for improving the service tomorrow, it may degrade quality and make users leave *today*, in which case there will be no users to learn

*This research has been done while Y. Mansour was co-affiliated with Microsoft Research, and while Z.S. Wu was a graduate student at University of Pennsylvania.

[†]Tel Aviv University. Email: mansour@tau.ac.il.

[‡]Microsoft Research, New York, NY, USA. Email: slivkins@microsoft.com.

[§]Microsoft Research, New York, NY, USA. Email: zhiww@microsoft.com.

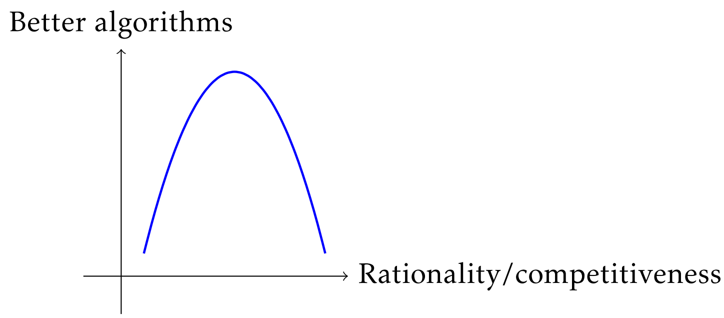


Figure 1: Inverted-U relationship between rationality/competitiveness and algorithms.

from! Thus, users play three distinct roles: they are customers that generate revenue, they generate data for the systems to learn from, and they are self-interested agents which choose among the competing systems.

We initiate a study of the interplay between *exploration* and *competition*. The main high-level question is: **whether and to what extent competition incentivizes adoption of better exploration algorithms**. This translates into a number of more concrete questions. While it is commonly assumed that better learning technology always helps, is this so for our setting? In other words, would a better learning algorithm result in higher utility for a principal? Would it be used in an equilibrium of the "competition game"? Also, does competition lead to better social welfare compared to a monopoly? We investigate these questions for several models, as we vary the capacity of users to make rational decisions (*rationality*) and the severity of competition between the learning systems (*competitiveness*). The two are controlled by the same "knob" in our models; such coupling is not unusual in the literature, *e.g.*, see Gabaix et al. (2016).

On a high level, our contributions can be framed in terms of the "inverted-U relationship" between rationality/competitiveness and the quality of adopted algorithms (see Figure 1).

**Our model.** We define a game in which two firms (*principals*) simultaneously engage in exploration and compete for users (*agents*). These two processes are interlinked, as exploration decisions are experienced by users and informed by their feedback. We need to specify several conceptual pieces: how the principals and agents interact, what is the machine learning problem faced by each principal, and what is the information structure. Each piece can get rather complicated in isolation, let alone jointly, so we strive for simplicity. Thus, the basic model is as follows:

- A new agent arrives in each round $t = 1, 2, \ldots$, and chooses among the two principals. The principal chooses an action (*e.g.*, a list of web search results to show to the agent), the user experiences this action, and reports a reward. All agents have the same "decision rule" for choosing among the principals given the available information.

- Each principal faces a very basic and well-studied version of the multi-armed bandit problem: for each arriving agent, it chooses from a fixed set of actions (a.k.a. *arms*) and receives a reward drawn independently from a fixed distribution specific to this action.

- Principals simultaneously announce their learning algorithms before round 1, and cannot change them afterwards. There is a common Bayesian prior on the rewards (but the realized reward distributions are not observed by the principals or the agents). Agents do not receive any other information. Each principal only observes agents that chose him.

**Technical results.** Our results depend crucially on agents' "decision rule" for choosing among the principals. The simplest and perhaps the most obvious rule is to select the principal which maximizes their expected utility; we refer to it as HardMax. We find that HardMax is not conducive to adopting better algorithms. In fact, each principal's dominant strategy is to do no purposeful exploration whatsoever, and instead always choose an action that maximizes expected reward given the current information; we call this algorithm DynamicGreedy. While this algorithm may potentially try out different actions over time and acquire useful information, it is known to be dramatically bad in many important cases of multi-armed bandits — precisely because it does not explore on purpose, and may therefore fail to discover best/better actions. Further, we show that HardMax is very sensitive to tie-breaking when both principals have exactly the same expected utility according to agents' beliefs. If tie-breaking is probabilistically biased — say, principal 1 is always chosen with probability strictly larger than $\frac{1}{2}$ — then this principal has a simple "winning strategy" no matter what the other principal does.

We relax HardMax to allow each principal to be chosen with some fixed baseline probability. One intuitive interpretation is that there are "random agents" who choose a principal uniformly at random, and each arriving agent is either HardMax or "random" with some fixed probability. We call this model HardMax&Random. We find that better algorithms help in a big way: a sufficiently better algorithm is guaranteed to win all non-random agents after an initial learning phase. While the precise notion of "sufficiently better algorithm" is rather subtle, we note that commonly known "smart" bandit algorithms typically defeat the commonly known "naive" ones, and the latter typically defeat DynamicGreedy. However, there is a substantial caveat: one can defeat any algorithm by interleaving it with DynamicGreedy. This has two undesirable corollaries: a better algorithm may sometimes lose, and a pure Nash equilibrium typically does not exist.

We further relax the decision rule so that the probability of choosing a given principal varies smoothly as a function of the difference between principals' expected rewards; we call it SoftMax. For this model, the "better algorithm wins" result holds under much weaker assumptions on what constitutes a better algorithm. This is the most technical result of the paper. The competition in this setting is necessarily much more relaxed: typically, both principals attract approximately half of the agents as time goes by (but a better algorithm may attract slightly more).

All results extend to a much more general version of the multi-armed bandit problem in which the principal may observe additional feedback before and/or after each decision, as long as the feedback distribution does not change over time. In most results, principal's utility may depend on both the market share and agents' rewards.

**Economic interpretation.** The inverted-U relationship between the severity of competition among firms and the quality of technologies that they adopt is a familiar theme in the economics literature (*e.g.,* Aghion et al., 2005; Vives, 2008).[1] We find it illuminating to frame our contributions in a similar manner, as illustrated in Figure 1.

[1] The literature frames this relationship as one between "competition" and "innovation". In this context, "innovation" refers to adoption of a better technology, at a substantial R&D expense to a given firm. It is not salient whether similar ideas and/or technologies already exist outside the firm. It is worth noting that adoption of exploration algorithms tends to require substantial R&D effort in practice, even if the algorithms themselves are well-known in the research literature; see Agarwal et al. (2016) for an example of such R&D effort.

Our models differ in terms of rationality in agents' decision-making: from fully rational decisions with `HardMax` to relaxed rationality with `HardMax&Random` to an even more relaxed rationality with `SoftMax`. The same distinctions also control the severity of competition between the principals: from cut-throat competition with `HardMax` to a more relaxed competition with `HardMax&Random`, to an even more relaxed competition with `SoftMax`. Indeed, with `HardMax` you lose all customers as soon as you fall behind in performance, with `HardMax&Random` you get some small market share no matter what, and with `SoftMax` you are further guaranteed a market share close to $\frac{1}{2}$ as long as your performance is not much worse than the competition. The uniform choice among principals corresponds to no rationality and no competition.

We identify the inverted-U relationship in the spirit of Figure 1 that is driven by the rationality/competitiveness distinctions outlined above: from `HardMax` to `HardMax&Random` to `SoftMax` to `Uniform`. We also find another, technically different inverted-U relationship which zeroes in on the `HardMax&Random` model. We vary rationality/competitiveness inside this model, and track the marginal utility of switching to a better algorithm.

These inverted-U relationships arise for a fundamentally different reason, compared to the existing literature on "competition vs. innovation." In the literature, better technology always helps in a competitive environment, other things being equal. Thus, the tradeoff is between the costs of improving the technology and the benefits that the improved technology provides in the competition. Meanwhile, we find that a better exploration algorithm may sometimes perform much worse under competition, even in the absence of R&D costs.

**Discussion.** We capture some pertinent features of reality while ignoring some others for the sake of tractability. Most notably, we assume that agents do not receive any information about other agents' rewards after the game starts. In the final analysis, this assumption makes agents' behavior independent of a particular realization of the Bayesian prior, and therefore enables us to summarize each learning algorithm via its Bayesian-expected rewards (as opposed to detailed performance on the particular realizations of the prior). Such summarization is essential for formulating lucid and general analytic results, let alone proving them. It is a major open question whether one can incorporate signals about other agents' rewards and obtain a tractable model.

We also make a standard assumption that agents are myopic: they do not worry about how their actions impact their future utility. In particular, they do not attempt to learn over time, to second-guess or game future agents, or to manipulate principal's learning algorithm. We believe this is a typical case in practice, in part because agent's influence tend to be small compared to the overall system. We model this simply by assuming that each agent only arrives once.

Much of the challenge in this paper, both conceptual and technical, was in setting up the right model and the matching theorems, and not only in proving the theorems. Apart from making the modeling choices described above, it was crucial to interpret the results and intuitions from the literature on multi-armed bandits so as to formulate meaningful assumptions on bandit algorithms and Bayesian priors which are productive in our setting.

**Open questions.** How to incorporate signals about the other agents' rewards? One needs to reason about how exact or coarse these signals are, and how the agents update their beliefs after receiving them. Also, one may need to allow principals' learning algorithms to respond to updates about the other principal's performance. (Or not, since this is not how learning algorithms are usually designed!) A clean, albeit idealized, model would be that (i) each agent learns her exact expected reward from each principal before she needs to choose which principal to go to, but (ii) these updates are invisible to the principals. Even then, one needs to argue about the competition on particular realizations of the Bayesian prior, which appears very challenging.

Another promising extension is to heterogeneous agents. Then the agents' choices are impacted by their idiosyncratic signals/beliefs, instead of being entirely determined by priors and/or signals about the average performance. It would be particularly interesting to investigate the emergence of *specialization*: whether/when an algorithm learns to target specific population segments in order to compete against a more powerful "incumbent".

**Map of the paper.** We survey related work (Section 2), lay out the model and preliminaries (Section 3), and proceed to analyze the three main models, `HardMax`, `HardMax&Random` and `SoftMax` (in Sections 4, 5, and 6, respectively). We discuss economic implications in Section 7. Appendix A provides some pertinent background on multi-armed bandits. Appendix B gives a broad example to support an assumption in our model.

## 2 Related work

Multi-armed bandits (*MAB*) is a particularly elegant and tractable abstraction for tradeoff between *exploration* and *exploitation*: essentially, between acquisition and usage of information. MAB problems have been studied in Economics, Operations Research and Computer Science for many decades; see (Bubeck and Cesa-Bianchi, 2012; Gittins et al., 2011; Slivkins, 2017) for background on regret-minimizing and Bayesian formulations, respectively. A discussion of industrial applications of MAB can be found in Agarwal et al. (2016).

The literature on MAB is vast and multi-threaded. The most related thread concerns regret-minimizing MAB formulations with IID rewards (Lai and Robbins, 1985; Auer et al., 2002a). This thread includes "smart" MAB algorithms that combine exploration and exploitation, such as UCB1 (Auer et al., 2002a) and Successive Elimination (Even-Dar et al., 2006), and "naive" MAB algorithms that separate exploration and exploitation, including explore-first and $\epsilon$-Greedy (*e.g.*, see Slivkins, 2017).

The three-way tradeoff between exploration, exploitation and incentives has been studied in several other settings: incentivizing exploration in a recommendation system (Che and Hörner, 2015; Frazier et al., 2014; Kremer et al., 2014; Mansour et al., 2015; Bimpikis et al., 2017; Bahar et al., 2016; Mansour et al., 2016), dynamic auctions (*e.g.*, Athey and Segal, 2013; Bergemann and Välimäki, 2010; Kakade et al., 2013), pay-per-click ad auctions with unknown click probabilities (*e.g.*, Babaioff et al., 2014; Devanur and Kakade, 2009; Babaioff et al., 2015), coordinating search and matching by self-interested agents (Kleinberg et al., 2016), as well as human computation (*e.g.*, Ho et al., 2016; Ghosh and Hummel, 2013; Singla and Krause, 2013).

Bolton and Harris (1999); Keller et al. (2005); Gummadi et al. (2012) studied models with self-interested agents jointly performing exploration, with no principal to coordinate them.

There is a superficial similarity (in name only) between this paper and the line of work on "dueling bandits" (*e.g.*, Yue et al., 2012; Yue and Joachims, 2009). The latter is not about competing bandit algorithms, but rather about scenarios where in each round two arms are chosen to be presented to a user, and the algorithm only observes which arm has "won the duel".

Our setting is closely related to the "dueling algorithms" framework (Immorlica et al., 2011) which studies competition between two principals, each running an algorithm for the same problem. However, this work considers algorithms for offline / full input scenarios, whereas we focus on online machine learning and the explore-exploit-incentives tradeoff therein. Also, this work specifically assumes binary payoffs (*i.e.*, win or lose) for the principals.

**Other related work in economics.** The competition vs. innovation relationship and the inverted-U shape thereof have been introduced in a classic book (Schumpeter, 1942), and remained an important theme in the literature ever since (*e.g.*, Aghion et al., 2005; Vives, 2008). Production costs aside, this literature treats innovation as a priori beneficial for the firm. Our setting is very different, as innovation in exploration algorithms may potentially hurt the firm.

A line of work on *platform competition*, starting with Rysman (2009), concerns competition between firms (*platforms*) that improve as they attract more users (*network effect*); see Weyl and White (2014) for a recent survey. This literature is not concerned with *innovation*, and typically models network effects exogenously, whereas in our model network effects are endogenous: they are created by MAB algorithms, an essential part of the model.

Relaxed versions of rationality similar to ours are found in several notable lines of work. For example, "random agents" (a.k.a. noise traders) can side-step the "no-trade theorem" (Milgrom and Stokey, 1982), a famous impossibility result in financial economics. The `SoftMax` model is closely related to the literature on *product differentiation*, starting from Hotelling (1929), see Perloff and Salop (1985) for a notable later paper.

There is a large literature on non-existence of equilibria due to small deviations (which is related to the corresponding result for `HardMax&Random`), starting with Rothschild and Stiglitz (1976) in the context of health insurance markets. Notable recent papers (Veiga and Weyl, 2016; Azevedo and Gottlieb, 2017) emphasize the distinction between `HardMax` and versions of `SoftMax`.

# 3 Our model and preliminaries

**Principals and agents.** There are two principals and $T$ agents. The game proceeds in rounds (we will sometimes refer to them as *global rounds*). In each round $t \in [T]$, the following interaction takes place. A new agent arrives and chooses one of the two principals. The principal chooses a recommendation: an action $a_t \in A$, where $A$ is a fixed set of actions (same for both principals and all rounds). The agent follows this recommendation, receives a reward $r_t \in [0,1]$, and reports it back to the principal.

The rewards are i.i.d. with a common prior. More formally, for each action $a \in A$ there is a parametric family $\psi_a(\cdot)$ of reward distributions, parameterized by the mean reward $\mu_a$. (The paradigmatic case is 0-1 rewards with a given expectation.) The mean reward vector $\mu = (\mu_a : a \in A)$ is drawn from prior distribution $\mathcal{P}_{\mathtt{mean}}$ before round 1. Whenever a given action $a \in A$ is chosen, the reward is drawn independently from distribution $\psi_a(\mu_a)$. The prior $\mathcal{P}_{\mathtt{mean}}$ and the distributions $(\psi_a(\cdot) \colon a \in A)$ constitute the (full) Bayesian prior on rewards, denoted $\mathcal{P}$.

Each principal commits to a learning algorithm for making recommendations. This algorithm follows a protocol of *multi-armed bandits* (*MAB*). Namely, the algorithm proceeds in time-steps:[2] each time it is called, it outputs a chosen action $a \in A$ and then inputs the reward for this action. The algorithm is called only in global rounds when the corresponding principal is chosen.

The information structure is as follows. The prior $\mathcal{P}$ is known to everyone. The mean rewards $\mu_a$ are not revealed to anybody. Each agent knows both principals' algorithms, and the global

[2]These time-steps will sometimes be referred to as *local steps/rounds*, so as to distinguish them from "global rounds" defined before. We will omit the local vs. local distinction when clear from the context.

round when (s)he arrives, *but not* the rewards of the previous agents. Each principal is completely unaware of the rounds when the other is chosen.

**Some terminology.** The two principals are called "Principal 1" and "Principal 2". The algorithm of principal $i \in \{1,2\}$ is called "algorithm $i$" and denoted $\texttt{alg}_i$. The agent in global round $t$ is called "agent $t$"; the chosen principal is denoted $i_t$.

Throughout, $\mathbb{E}[\cdot]$ denotes expectation over all applicable randomness.

**Bayesian-expected rewards.** Consider the performance of a given algorithm $\texttt{alg}_i$, $i \in \{1,2\}$, when it is run in isolation (*i.e.*, without competition, just as a bandit algorithm). Let $\texttt{rew}_i(n)$ denote its Bayesian-expected reward for the $n$-th step.

Now, going back to our game, fix global round $t$ and let $n_i(t)$ denote the number of global rounds before $t$ in which this principal is chosen. Then:

$$\mathbb{E}[r_t \mid \text{principal } i \text{ is chosen in round } t \text{ and } n_i(t) = n] = \texttt{rew}_i(n+1) \quad (\forall n \in \mathbb{N}).$$

**Agents' response.** Each agent $t$ chooses principal $i_t$ as as follows: it chooses a distribution over the principals, and then draws independently from this distribution. Let $p_t$ be the probability of choosing principal 1 according to this distribution. Below we specify $p_t$; we need to be careful so as to avoid a circular definition.

Let $\mathcal{I}_t$ be the information available to agent $t$ before the round. Assume $\mathcal{I}_t$ suffices to form posteriors for quantities $n_i(t)$, $i \in \{1,2\}$, denote them by $\mathcal{N}_{i,t}$. Note that the Bayesian expected reward of each principal $i$ is a function only of the number rounds he was chosen by the agents, so the posterior mean reward for each principal $i$ can be written as

$$\mathsf{PMR}_i(t) := \mathbb{E}[r_t \mid \mathcal{I}_t \text{ and } i_t = i] = \mathbb{E}[\texttt{rew}_i(n_i(t)+1) \mid \mathcal{I}_t] = \mathop{\mathbb{E}}_{n \sim \mathcal{N}_{i,t}}[\texttt{rew}_i(n+1)].$$

This quantity represents the posterior mean reward for principal $i$ at round $t$, according to information $\mathcal{I}_t$; hence the notation $\mathsf{PMR}$. In general, probability $p_t$ is defined by the posterior mean rewards $\mathsf{PMR}_i(t)$ for both principals. We assume a somewhat more specific shape:

$$p_t = f_{\text{resp}}(\,\mathsf{PMR}_1(t) - \mathsf{PMR}_2(t)\,). \tag{1}$$

Here $f_{\text{resp}} : [-1,1] \to [0,1]$ is the *response function*, which is the same for all agents. We assume that the response function is known to all agents.

To make the model well-defined, it remains to argue that information $\mathcal{I}_t$ is indeed sufficient to form posteriors on $n_1(t)$ and $n_2(t)$. This can be easily seen using induction on $t$.

Since all agents arrive with identical information (other than knowing which global round they arrive in), it follows that all agents have identical posteriors for $n_{i,t}$ (for a given principal $i$ and a given global round $t$). This posterior is denoted $\mathcal{N}_{i,t}$.

**Response functions.** We use the response function $f_{\text{resp}}$ to characterize the amount of rationality and competitiveness in our model. We assume that $f_{\text{resp}}$ is monotonically non-decreasing, is larger than $\frac{1}{2}$ on the interval $(0,1]$, and smaller than $\frac{1}{2}$ on the interval $[-1,0)$. Beyond that, we consider three specific models, listed in the order of decreasing rationality and competitiveness (see Figure 2):

- $\mathsf{HardMax}$: $f_{\text{resp}}$ equals 0 on the interval $[-1,0)$ and 1 on the interval $(0,1]$. In other words, the agents will deterministically choose the principal with the higher posterior mean reward.

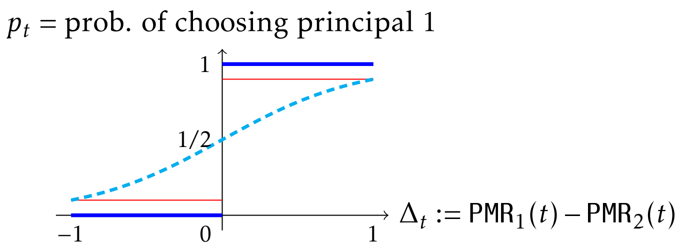


Figure 2: The three models for agents' response function: HardMax is thick blue, HardMax&Random is slim red, and SoftMax is the dashed curve.

- HardMax&Random: $f_{\mathrm{resp}}$ equals $\epsilon_0$ on the interval $[-1,0)$ and $1-\epsilon_0$ on the interval $(0,1]$, where $\epsilon_0 \in (0, \frac{1}{2})$ are some positive constants. In words, each agent is a HardMax agent with probability $1-2\epsilon_0$, and with the remaining probability she makes a random choice.
- SoftMax: $f_{\mathrm{resp}}(\cdot)$ lies in the interval $[\epsilon_0, 1-\epsilon_0]$, $\epsilon_0 > 0$, and is "smooth" around 0 (in the sense defined precisely in Section 6).

We say that $f_{\mathrm{resp}}$ is *symmetric* if $f_{\mathrm{resp}}(-x) + f_{\mathrm{resp}}(x) = 1$ for any $x \in [0,1]$. This implies *fair tie-breaking*: $f_{\mathrm{resp}}(0) = \frac{1}{2}$.

**MAB algorithms.** We characterize the inherent quality of an MAB algorithm in terms of its *Bayesian Instantaneous Regret* (henceforth, BIR), a standard notion from machine learning:

$$\mathtt{BIR}(n) := \mathop{\mathbb{E}}_{\mu \sim \mathcal{P}_{\mathrm{mean}}}\left[\max_{a \in A} \mu_a\right] - \mathtt{rew}(n), \tag{2}$$

where $\mathtt{rew}(n)$ is the Bayesian-expected reward of the algorithm for the $n$-th step, when the algorithm is run in isolation. We are primarily interested in how BIR scales with $n$; we treat $K$, the number of arms, as a constant unless specified otherwise.

We will emphasize several specific algorithms or classes thereof:

- "smart" MAB algorithms that combine exploration and exploitation, such as UCB1 Auer et al. (2002a) and Successive Elimination Even-Dar et al. (2006). These algorithms achieve $\mathtt{BIR}(n) \le \tilde{O}(n^{-1/2})$ for all priors and all (or all but a very few) steps $n$. This bound is known to be tight for any fixed $n$. [3]

- "naive" MAB algorithms that separate exploration and exploitation, such as Explore-then-Exploit and $\epsilon$-Greedy. These algorithms have dedicated rounds in which they explore by choosing an action uniformly at random. When these rounds are known in advance, the algorithm suffers constant BIR in such rounds. When the "exploration rounds" are instead randomly chosen by the algorithm, one can usually guarantee an inverse-polynomial upper bound BIR, but not as good as the one above: namely, $\mathtt{BIR}(n) \le \tilde{O}(n^{-1/3})$. This is the best possible upper bound on BIR for the two algorithms mentioned above.

- DynamicGreedy: at each step, recommends the best action according to the current posterior: an action $a$ with the highest posterior expected reward $\mathbb{E}[\mu_a \mid \mathcal{I}]$, where $\mathcal{I}$ is the information

[3] This follows from the lower-bound analysis in Auer et al. (2002b).

available to the algorithm so far. DynamicGreedy has (at least) a constant BIR for some reasonable priors, *i.e.*, $\text{BIR}(n) > \Omega(1)$.

- StaticGreedy: always recommends the prior best action,*i.e.*, an action $a$ with the highest prior mean reward $\mathbb{E}_{\mu\sim\mathcal{P}_{\text{mean}}}[\mu_a]$. This algorithm typically has constant BIR.

We focus on MAB algorithms such that $\text{BIR}(n)$ is non-increasing; we call such algorithms *monotone*. While some reasonable MAB algorithms may occasionally violate monotonicity, they can usually be easily modified so that monotonicity violations either vanish altogether, or only occur at very specific rounds (so that agents are extremely unlikely to exploit them in practice).

More background and examples can be found in Appendix A. In particular, we prove that DynamicGreedy is monotone.

**Competition game between principals.** Some of our results explicitly study the game between the two principals. We model it as a simultaneous-move game: before the first agent arrives, each principal commits to an MAB algorithm. Thus, choosing a pure strategy in this game corresponds to choosing an MAB algorithm (and, implicitly, announcing this algorithm to the agents).

Principal's utility is primarily defined as the market share, *i.e.*, the number of agents that chose this principal. Principals are risk-neutral, in the sense that they optimize their expected utility.

**Assumptions on the prior.** We make some technical assumptions for the sake of simplicity. First, each action $a$ has a positive probability of being the best action according to the prior:

$$\forall a \in A: \Pr_{\mu\sim\mathcal{P}_{\text{mean}}}[\mu_a > \mu_{a'}\ \forall a' \in A] > 0. \tag{3}$$

Second, posterior mean rewards of actions are pairwise distinct almost surely. That is, the history $h$ at any step of an MAB algorithm[4] satisfies

$$\mathbb{E}[\mu_a \mid h] \neq \mathbb{E}[\mu_{a'} \mid h] \quad \forall a, a' \in A, \tag{4}$$

except at a set of histories of probability 0. In particular, prior mean rewards of actions are pairwise distinct: $\mathbb{E}[\mu_a] \neq \mathbb{E}[\mu'_a]$ for any $a, a' \in A$.

We provide two examples for which property (4) is 'generic', in the sense that it can be enforced almost surely by a small random perturbation of the prior. Both examples focus on 0-1 rewards and priors $\mathcal{P}_{\text{mean}}$ that are independent across arms. The first example assumes Beta priors on the mean rewards, and is very easy.[5] The second example assumes that mean rewards have a finite support, see Appendix B for details.

**Some more notation.** Without loss of generality, we label actions as $A = [K]$ and sort them according to their prior mean rewards, so that $\mathbb{E}[\mu_1] > \mathbb{E}[\mu_2] > \ldots > \mathbb{E}[\mu_K]$.

Fix principal $i \in \{1, 2\}$ and (local) step $n$. The arm chosen by algorithm $\texttt{alg}_i$ at this step is denoted $a_{i,n}$, and the corresponding BIR is denoted $\text{BIR}_i(n)$. History of $\texttt{alg}_i$ up to this step is denoted $H_{i,n}$.

Write $\text{PMR}(a \mid E) = \mathbb{E}[\mu_a \mid E]$ for posterior mean reward of action $a$ given event $E$.

[4]The *history* of an MAB algorithm at a given step comprises the chosen actions and the observed rewards in all previous steps in the execution of this algorithm.

[5]Suppose the rewards are Bernouli r.v. and the mean reward $\mu_a$ for each arm $a$ is drawn from some Beta distribution $\text{Beta}(\alpha_a, \beta_a)$. Given any history that contains $h_a$ number of heads and $t_a$ number of tails from arm $a$, the posterior mean reward is $\frac{\alpha_a+h_a}{\alpha_a+h_a+\beta_a+t_a}$. Note that $h_a$ and $t_a$ take integer values. Therefore, perturbing the parameters $\alpha_a$ and $\beta_a$ independently with any continuous noise will induce a prior with property (4) with probability 1.

### 3.1 Generalizations

Our results can be extended compared to the basic model described above.

First, unless specified otherwise, our results allow a more general notion of principal's utility that can depend on both the market share and agents' rewards. Namely, principal $i$ collects $U_i(r_t)$ units of utility in each global round $t$ when she is chosen (and 0 otherwise), where $U_i(\cdot)$ is some fixed non-decreasing function with $U_i(0) > 0$. In a formula,

$$U_i := \textstyle\sum_{t=1}^{T} \mathbf{1}_{\{i_t=i\}} \cdot U_i(r_r). \tag{5}$$

Second, our results carry over, with little or no modification of the proofs, to much more general versions of MAB, as long as it satisfies the i.i.d. property. In each round, an algorithm can see a *context* before choosing an action (as in *contextual bandits*) and/or additional feedback other than the reward after the reward is chosen (as in, *e.g.*, *semi-bandits*), as long as the contexts are drawn from a fixed distribution, and the (reward, feedback) pair is drawn from a fixed distribution that depends only on the context and the chosen action. The Bayesian prior $\mathcal{P}$ needs to be a more complicated object, to make sure that `PMR` and `BIR` are well-defined. Mean rewards may also have a known structure, such as Lipschitzness, convexity, or linearity; such structure can be incorporated via $\mathcal{P}$. All these extensions have been studied extensively in the literature on MAB, and account for a substantial segment thereof; see Bubeck and Cesa-Bianchi (2012) for background and details.

### 3.2 Chernoff Bounds

We use an elementary concentration inequality known as *Chernoff Bounds*, in a formulation from Mitzenmacher and Upfal (2005).

**Theorem 3.1** (Chernoff Bounds)**.** *Consider $n$ i.i.d. random variables $X_1 \ldots X_n$ with values in $[0,1]$. Let $X = \frac{1}{n}\sum_{i=1}^{n} X_i$ be their average, and let $\nu = \mathbb{E}[X]$. Then:*

$$\min\left(\, \Pr[X-\nu > \delta\nu], \quad \Pr[\nu - X > \delta\nu] \,\right) < e^{-\nu n \delta^2/3} \quad \textit{for any } \delta \in (0,1).$$

## 4 Full rationality (HardMax)

In this section, we will consider the version in which the agents are fully rational, in the sense that their response function is `HardMax`. We show that principals are not incentivized to *explore*—*i.e.*, to deviate from `DynamicGreedy`. The core technical result is that if one principal adopts `DynamicGreedy`, then the other principal loses all agents as soon as he deviates.

To make this more precise, let us say that two MAB algorithms *deviate* at (local) step $n$ if there is an action $a \in A$ and a set of step-$n$ histories of positive probability such that any history $h$ in this set is feasible for both algorithms, and under this history the two algorithms choose action $a$ with different probability.

**Theorem 4.1.** *Assume* `HardMax` *response function with fair tie-breaking. Assume that* $\texttt{alg}_1$ *is* `DynamicGreedy`, *and* $\texttt{alg}_2$ *deviates from* `DynamicGreedy` *starting from some (local) step $n_0 < T$. Then all agents in global rounds $t \geq n_0$ select principal* 1.

**Corollary 4.2.** *The competition game between principals has a unique Nash equilibirium: both principals choose* `DynamicGreedy`.

*Remark* 4.3. This corollary holds under a more general model which allows time-discounting: namely, the utility of each principal $i$ in each global round $t$ is $U_{i,t}(r_t)$ if this principal is chosen, and 0 otherwise, where $U_{i,t}(\cdot)$ is an arbitrary non-decreasing function with $U_{i,t}(0) > 0$.

### 4.1 Proof of Theorem 4.1

The proof starts with two auxiliary lemmas: that deviating from `DynamicGreedy` implies a strictly smaller Bayesian-expected reward, and that `HardMax` implies a "sudden-death" property: if one agent chooses principal 1 with certainty, then so do all subsequent agents do. We re-use both lemmas in later sections, so we state them in sufficient generality.

**Lemma 4.4.** *Assume that* $\texttt{alg}_1$ *is* `DynamicGreedy`, *and* $\texttt{alg}_2$ *deviates from* `DynamicGreedy` *starting from some (local) step* $n_0 < T$. *Then* $\texttt{rew}_1(n_0) > \texttt{rew}_2(n_0)$. *This holds for any response function* $f_{\texttt{resp}}$.

Lemma 4.4 does not rely on any particular shape of the response function because it only considers the performance of each algorithm without competition.

*Proof of Lemma 4.4.* Since the two algorithms coincide on the first $n_0-1$ steps, it follows by symmetry that histories $H_{1,n_0}$ and $H_{2,n_0}$ have the same distribution. We use a *coupling argument*: w.l.o.g., we assume the two histories coincide, $H_{1,n_0} = H_{2,n_0} = H$.

At local step $n_0$, `DynamicGreedy` chooses an action $a_{1,n_0} = a_{1,n_0}(H)$ which maximizes the posterior mean reward given history $H$: for any realized history $h \in \texttt{support}(H)$ and any action $a \in A$

$$\mathsf{PMR}(a_{1,n_0} \mid H = h) \geq \mathsf{PMR}(a \mid H = h). \tag{6}$$

By assumption (4), it follows that

$$\mathsf{PMR}(a_{1,n_0} \mid H = h) > \mathsf{PMR}(a \mid H = h) \quad \text{for any } h \in \texttt{support}(H) \text{ and } a \neq a_{1,n_0}(h). \tag{7}$$

Since the two algorithms deviate at step $n_0$, there is a set $S \subset \texttt{support}(H)$ of step-$n_0$ histories such that $\Pr[S] > 0$ and any history $h \in S$ satisfies $\Pr[a_{2,n_0} \neq a_{1,n_0} \mid H = h] > 0$. Combining this with (7), we deduce that

$$\mathsf{PMR}(a_{1,n_0} \mid H = h) > \mathbb{E}\left[\mu_{a_{2,n_0}} \mid H = h\right] \quad \text{for each history } h \in S. \tag{8}$$

Using (6) and (8) and integrating over realized histories $h$, we obtain $\texttt{rew}_1(n_0) > \texttt{rew}_2(n_0)$. □

**Lemma 4.5.** *Consider* `HardMax` *response function with* $f_{\texttt{resp}}(0) \geq \frac{1}{2}$. *Suppose* $\texttt{alg}_1$ *is monotone, and* $\mathsf{PMR}_1(t_0) > \mathsf{PMR}_2(t_0)$ *for some global round* $t_0$. *Then* $\mathsf{PMR}_1(t) > \mathsf{PMR}_2(t)$ *for all subsequent rounds* $t$.

*Proof.* Let us use induction on round $t \geq t_0$, with the base case $t = t_0$. Let $\mathcal{N} = \mathcal{N}_{1,t_0}$ be the agents' posterior distribution for $n_{1,t_0}$, the number of global rounds before $t_0$ in which principal 1 is chosen. By induction, all agents from $t_0$ to $t-1$ chose principal 1, so $\mathsf{PMR}_2(t_0) = \mathsf{PMR}_2(t)$. Therefore,

$$\mathsf{PMR}_1(t) = \mathop{\mathbb{E}}_{n\sim\mathcal{N}}\left[\texttt{rew}_1(n+1+t-t_0)\right] \geq \mathop{\mathbb{E}}_{n\sim\mathcal{N}}\left[\texttt{rew}_1(n+1)\right] = \mathsf{PMR}_1(t_0) > \mathsf{PMR}_2(t_0) = \mathsf{PMR}_2(t),$$

where the first inequality holds because $\texttt{alg}_1$ is monotone, and the second one is the base case. □

*Proof of Theorem 4.1.* Since the two algorithms coincide on the first $n_0 - 1$ steps, it follows by symmetry that $\mathtt{rew}_1(n) = \mathtt{rew}_2(n)$ for any $n < n_0$. By Lemma 4.4, $\mathtt{rew}_1(n_0) > \mathtt{rew}_2(n_0)$.

Recall that $n_i(t)$ is the number of global rounds $s < t$ in which principal $i$ is chosen, and $\mathcal{N}_{i,t}$ is the agents' posterior distribution for this quantity. By symmetry, each agent $t < n_0$ chooses a principal uniformly at random. It follows that $\mathcal{N}_{1,n_0} = \mathcal{N}_{2,n_0}$ (denote both distributions by $\mathcal{N}$ for brevity), and $\mathcal{N}(n_0 - 1) > 0$. Therefore:

$$\begin{aligned} \mathsf{PMR}_1(n_0) &= \mathop{\mathbb{E}}_{n\sim\mathcal{N}}[\mathtt{rew}_1(n+1)] = \sum_{n=0}^{n_0-1} \mathcal{N}(n)\cdot \mathtt{rew}_1(n+1) \\ &> \mathcal{N}(n_0-1)\cdot \mathtt{rew}_2(n_0) + \sum_{n=0}^{n_0-2} \mathcal{N}(n)\cdot \mathtt{rew}_2(n+1) \\ &= \mathop{\mathbb{E}}_{n\sim\mathcal{N}}[\mathtt{rew}_2(n+1)] = \mathsf{PMR}_2(n_0) \end{aligned} \tag{9}$$

So, agent $n_0$ chooses principal 1. By Lemma 4.5 (noting that `DynamicGreedy` is monotone), all subsequent agents choose principal 1, too. □

## 4.2 `HardMax` with biased tie-breaking

The `HardMax` model is very sensitive to the tie-breaking rule. For starters, if ties are broken deterministically in favor of principal 1, then principal 1 can get all agents no matter what the other principal does, simply by using `StaticGreedy`.

**Theorem 4.6.** *Assume* `HardMax` *response function with* $f_{\text{resp}}(0) = 1$ *(ties are always broken in favor of principal* 1*). If* $\mathtt{alg}_1$ *is* `StaticGreedy`, *then all agents choose principal* 1.

*Proof.* Agent 1 chooses principal 1 because of the tie-breaking rule. Since `StaticGreedy` is trivially monotone, all the subsequent agents choose principal 1 by an induction argument similar to the one in the proof of Lemma 4.5. □

A more challenging scenario is when the tie-breaking is biased in favor of principal 1, but not deterministically so: $f_{\text{resp}}(0) > \frac{1}{2}$. Then this principal also has a "winning strategy" no matter what the other principal does. Specifically, principal 1 can get all but the first few agents, under a mild technical assumption that `DynamicGreedy` deviates from `StaticGreedy`. Principal 1 can use `DynamicGreedy`, or any other monotone MAB algorithm that coincides with `DynamicGreedy` in the first few steps.

**Theorem 4.7.** *Assume* `HardMax` *response function with* $f_{\text{resp}}(0) > \frac{1}{2}$ (i.e., *tie-breaking is biased in favor of principal* 1*). Assume the prior* $\mathcal{P}$ *is such that* `DynamicGreedy` *deviates from* `StaticGreedy` *starting from some step* $n_0$. *Suppose that principal* 1 *runs a monotone MAB algorithm that coincides with* `DynamicGreedy` *in the first* $n_0$ *steps. Then all agents* $t \geq n_0$ *choose principal* 1.

*Proof.* The proof re-uses Lemmas 4.4 and 4.5, which do not rely on fair tie-breaking.

Because of the biased tie-breaking, for each global round $t$ we have:

$$\text{if } \mathsf{PMR}_1(t) \geq \mathsf{PMR}_2(t) \text{ then } \Pr[i_t = 1] > \tfrac{1}{2}. \tag{10}$$

Recall that $i_t$ is the principal chosen in global round $t$.

Let $m_0$ be the first step when $\texttt{alg}_2$ deviates from DynamicGreedy, or DynamicGreedy deviates from StaticGreedy, whichever comes sooner. Then $\texttt{alg}_2$, DynamicGreedy and StaticGreedy coincide on the first $m_0-1$ steps. Moreover, $m_0 \leq n_0$ (since DynamicGreedy deviates from StaticGreedy at step $n_0$), so $\texttt{alg}_1$ coincides with DynamicGreedy on the first $m_0$ steps.

So, $\texttt{rew}_1(n) = \texttt{rew}_2(n)$ for each step $n < m_0$, because $\texttt{alg}_1$ and $\texttt{alg}_2$ coincide on the first $m_0 - 1$ steps. Moreover, if $\texttt{alg}_2$ deviates from DynamicGreedy at step $m_0$ then $\texttt{rew}_1(m_0) > \texttt{rew}_2(m_0)$ by Lemma 4.4; else, we trivially have $\texttt{rew}_1(m_0) = \texttt{rew}_2(m_0)$. To summarize:

$$\texttt{rew}_1(n) \geq \texttt{rew}_2(n) \quad \text{for all steps } n \leq m_0. \tag{11}$$

We claim that $\Pr[i_t = 1] > \frac{1}{2}$ for all global rounds $t \leq m_0$. We prove this claim using induction on $t$. The base case $t = 1$ holds by (10) and the fact that in step 1, DynamicGreedy chooses the arm with the highest prior mean reward. For the induction step, we assume that $\Pr[i_t = 1] > \frac{1}{2}$ for all global rounds $t < t_0$, for some $t_0 \leq m_0$. It follows that distribution $\mathcal{N}_{1,t_0}$ stochastically dominates distribution $\mathcal{N}_{2,t_0}$.[6] Observe that

$$\mathsf{PMR}_1(t_0) = \mathop{\mathbb{E}}_{n\sim\mathcal{N}_{1,t_0}}[\texttt{rew}_1(n+1)] \geq \mathop{\mathbb{E}}_{n\sim\mathcal{N}_{2,t_0}}[\texttt{rew}_2(n+1)] = \mathsf{PMR}_2(t_0). \tag{12}$$

So the induction step follows by (10). Claim proved.

Now let us focus on global round $m_0$, and denote $\mathcal{N}_i = \mathcal{N}_{i,m_0}$. By the above claim,

$$\mathcal{N}_1 \text{ stochastically dominates } \mathcal{N}_2, \text{ and moreover } \mathcal{N}_i(m_0-1) > \mathcal{N}_i(m_0-1). \tag{13}$$

By definition of $m_0$, either (i) $\texttt{alg}_2$ deviates from DynamicGreedy starting from local step $m_0$, which implies $\texttt{rew}_1(m_0) > \texttt{rew}_2(m_0)$ by Lemma 4.4, or (ii) DynamicGreedy deviates from StaticGreedy starting from local step $m_0$, which implies $\texttt{rew}_1(m_0) > \texttt{rew}_1(m_0-1)$ by Lemma A.4. In both cases, using (11) and (13), it follows that the inequality in (12) is strict for $t_0 = m_0$.

Therefore, agent $m_0$ chooses principal 1, and by Lemma 4.5 so do all subsequent agents. □

## 5 Relaxed rationality: HardMax & Random

This section is dedicated to the HardMax&Random response model, where each principal is always chosen with some positive baseline probability. The main technical result for this model states that a principal with asymptotically better BIR wins by a large margin: after a "learning phase" of constant duration, all agents choose this principal with maximal possible probability $f_{\text{resp}}(1)$. For example, a principal with $\mathsf{BIR}(n) \leq \tilde{O}(n^{-1/2})$ wins over a principal with $\mathsf{BIR}(n) \geq \Omega(n^{-1/3})$. However, this positive result comes with a significant caveat detailed in Section 5.1.

We formulate and prove a cleaner version of the result, followed by a more general formulation developed in a subsequent Remark 5.2. We need to express a property that $\texttt{alg}_1$ eventually catches up and surpasses $\texttt{alg}_2$, even if initially it receives only a fraction of traffic. For the cleaner version, we assume that both algorithms are well-defined for an infinite time horizon, so that their BIR does not depend on the time horizon $T$ of the game. Then this property can be formalized as:

$$(\forall \epsilon > 0) \qquad \mathsf{BIR}_1(\epsilon n)/\mathsf{BIR}_2(n) \to 0. \tag{14}$$

[6] For random variables $X, Y$ on $\mathbb{R}$, we say that $X$ *stochastically dominates* $Y$ if $\Pr[X \geq x] \geq \Pr[Y \geq x]$ for any $x \in \mathbb{R}$.

In fact, a weaker version of (14) suffices: denoting $\epsilon_0 = f_{\mathtt{resp}}(-1)$, for some constant $n_0$ we have

$$(\forall n \geq n_0) \qquad \mathtt{BIR}_1(\epsilon_0 n/2)/\mathtt{BIR}_2(n) < \tfrac{1}{2}. \tag{15}$$

We also need a very mild technical assumption on the "bad" algorithm:

$$(\forall n \geq n_0) \qquad \mathtt{BIR}_2(n) > 4\, e^{-\epsilon_0 n/12}. \tag{16}$$

**Theorem 5.1.** *Assume* HardMax&Random *response function. Suppose both algorithms are monotone and well-defined for an infinite time horizon, and satisfy* (15) *and* (16)*. Then each agent $t \geq n_0$ chooses principal* 1 *with maximal possible probability $f_{\mathtt{resp}}(1) = 1 - \epsilon_0$.*

*Proof.* Consider global round $t \geq n_0$. Recall that each agent chooses principal 1 with probability at least $f_{\mathtt{resp}}(-1) > 0$.

Then $\mathbb{E}[n_1(t+1)] \geq 2\epsilon_0\, t$. By Chernoff Bounds (Theorem 3.1), we have that $n_1(t+1) \geq \epsilon_0 t$ holds with probability at least $1-q$, where $q = \exp(-\epsilon_0 t/12)$.

We need to prove that $\mathtt{PMR}_1(t) - \mathtt{PMR}_2(t) > 0$. For any $m_1$ and $m_2$, consider the quantity

$$\Delta(m_1, m_2) := \mathtt{BIR}_2(m_2+1) - \mathtt{BIR}_1(m_1+1).$$

Whenever $m_1 \geq \epsilon_0 t/2 - 1$ and $m_2 < t$, it holds that

$$\Delta(m_1, m_2) \geq \Delta(\epsilon_0 t/2, t) \geq \mathtt{BIR}_2(t)/2.$$

The above inequalities follow, resp., from algorithms' monotonicity and (15). Now,

$$\begin{aligned}
\mathtt{PMR}_1(t) - \mathtt{PMR}_2(t) &= \mathop{\mathbb{E}}_{m_1 \sim \mathcal{N}_{1,t},\, m_2 \sim \mathcal{N}_{2,t}} [\Delta(m_1, m_2)] \\
&\geq -q + \mathop{\mathbb{E}}_{m_1 \sim \mathcal{N}_{1,t},\, m_2 \sim \mathcal{N}_{2,t}} [\Delta(m_1, m_2) \mid m_1 \geq \epsilon_0 t/2 - 1] \\
&\geq \mathtt{BIR}_2(t)/2 - q \\
&> \mathtt{BIR}_2(t)/4 > 0 \qquad \text{(by (16))}.
\end{aligned}$$

□

*Remark* 5.2. Many standard MAB algorithms in the literature are parameterized by the time horizon $T$. Regret bounds for such algorithms usually include a polylogarithmic dependence on $T$. In particular, a typical upper bound for BIR has the following form:

$$\mathtt{BIR}(n \mid T) \leq \operatorname{polylog}(T) \cdot n^{-\gamma} \quad \text{for some } \gamma \in (0, \tfrac{1}{2}]. \tag{17}$$

Here we write $\mathtt{BIR}(n \mid T)$ to emphasize the dependence on $T$.

We generalize (15) to handle the dependence on $T$: there exists a number $T_0$ and a function $n_0(T) \in \operatorname{polylog}(T)$ such that

$$(\forall T \geq T_0,\ n \geq n_0(T)) \quad \frac{\mathtt{BIR}_1(\epsilon_0 n/2 \mid T)}{\mathtt{BIR}_2(n \mid T)} < \frac{1}{2}. \tag{18}$$

If this holds, we say that $\mathtt{alg}_1$ *BIR-dominates* $\mathtt{alg}_2$.

We provide a version of Theorem 5.1 in which algorithms are parameterized with time horizon $T$ and condition (15) is replaced with (18); its proof is very similar and is omitted.

To state a game-theoretic corollary of Theorem 5.1, we consider a version of the competition game between the two principals in which they can only choose from a finite set $\mathcal{A}$ of monotone MAB algorithms. One of these algorithms is "better" than all others; we call it the *special* algorithm. Unless specified otherwise, it BIR-dominates all other allowed algorithms. The other algorithms satisfy (16). We call this game the *restricted competition game*.

**Corollary 5.3.** *Assume* `HardMax&Random` *response function. Consider the restricted competition game with special algorithm* `alg`. *Then, for any sufficiently large time horizon $T$, this game has a unique Nash equilibrium: both principals choose* `alg`.

## 5.1 A little greedy goes a long way

Given any monotone MAB algorithm other than `DynamicGreedy`, we design a modified algorithm which learns at a slower rate, yet "wins the game" in the sense of Theorem 5.1. As a corollary, the competition game with unrestricted choice of algorithms typically does not have a Nash equilibrium.

Given an algorithm $\texttt{alg}_1$ that deviates from `DynamicGreedy` starting from step $n_0$ and a "mixing" parameter $p$, we will construct a modified algorithm as follows.

1. The modified algorithm coincides with $\texttt{alg}_1$ (and `DynamicGreedy`) for the first $n_0 - 1$ steps;
2. In each step $n \geq n_0$, $\texttt{alg}_1$ is invoked with probability $1-p$, and with the remaining probability $p$ does the "greedy choice": chooses an action with the largest posterior mean reward given the current information collected by $\texttt{alg}_1$.

For a cleaner comparison between the two algorithms, the modified algorithm does not record rewards received in steps with the "greedy choice". Parameter $p > 0$ is the same for all steps.

**Theorem 5.4.** *Assume symmetric* `HardMax&Random` *response function. Let $\epsilon_0 = f_{\texttt{resp}}(-1)$ be the baseline probability. Suppose $\texttt{alg}_1$ deviates from* `DynamicGreedy` *starting from some step $n_0$. Let $\texttt{alg}_2$ be the modified algorithm, as described above, with mixing parameter $p$ such that $(1-\epsilon_0)(1-p) > \epsilon_0$. Then each agent $t \geq n_0$ chooses principal 2 with maximal possible probability $1 - \epsilon_0$.*

**Corollary 5.5.** *Suppose that both principals can choose any monotone MAB algorithm, and assume the symmetric* `HardMax&Random` *response function. Then for any time horizon $T$, the only possible* pure *Nash equilibrium is one where both principals choose* `DynamicGreedy`. *Moreover, no pure Nash equilibrium exists when some algorithm "dominates"* `DynamicGreedy` *in the sense of* (18) *and the time horizon $T$ is sufficiently large.*

*Remark* 5.6. The modified algorithm performs exploration at a slower rate. Let us argue how this may translate into a larger `BIR` compared to the original algorithm. Let $\texttt{BIR}'_1(n)$ be the `BIR` of the "greedy choice" after after $n-1$ steps of $\texttt{alg}_1$. Then

$$\texttt{BIR}_2(n) = \mathop{\mathbb{E}}_{m \sim (n_0 - 1) + \texttt{Binomial}(n - n_0 + 1, 1-p)} \left[ (1-p) \cdot \texttt{BIR}_1(m) + p \cdot \texttt{BIR}'_1(m) \right]. \tag{19}$$

In this expression, $m$ is the number of times $\texttt{alg}_1$ is invoked in the first $n$ steps of the modified algorithm. Note that $\mathbb{E}[m] = n_0 - 1 + (n - n_0 + 1)(1-p) \geq (1-p)n$.

Suppose $\mathtt{BIR}_1(n) = \beta n^{-\gamma}$ for some constants $\beta, \gamma > 0$. Further, assume $\mathtt{BIR}'_1(n) \geq c\,\mathtt{BIR}_1(n)$, for some $c > 1-\gamma$. Then for all $n \geq n_0$ and small enough $p > 0$ it holds that:

$$\begin{aligned}
\mathtt{BIR}_2(n) &\geq (1-p+pc)\ \mathbb{E}[\ \mathtt{BIR}_1(m)\ ] \\
\mathbb{E}[\ \mathtt{BIR}_1(m)\ ] &\geq \mathtt{BIR}_1(\ \mathbb{E}[m]\ ) && \text{(By Jensen's inequality)} \\
&\geq \mathtt{BIR}_1(\ (1-p)n\ ) && \text{(since } \mathbb{E}[m] \geq n(1-p)\text{)} \\
&\geq \beta \cdot n^{-\gamma} \cdot (1-p)^{-\gamma} && \text{(plugging in } \mathtt{BIR}_1(n) = \beta n^{-\gamma}\text{)} \\
&> \mathtt{BIR}_1(n)\ \ (1-p\gamma)^{-1} && \text{(since } (1-p)^{\gamma} < 1-p\gamma\text{).} \\
\mathtt{BIR}_2(n) &> \alpha \cdot \mathtt{BIR}_1(n), && \text{where} \quad \alpha = \tfrac{1-p+pc}{1-p\gamma} > 1.
\end{aligned}$$

(In the above equations, all expectations are over $m$ distributed as in (19).)

*Proof of Theorem 5.4.* Let $\mathtt{rew}'_1(n)$ denote the Bayesian-expected reward of the "greedy choice" after after $n-1$ steps of $\mathtt{alg}_1$. Note that $\mathtt{rew}_1(\cdot)$ and $\mathtt{rew}'_1(\cdot)$ are non-decreasing: the former because $\mathtt{alg}_1$ is monotone and the latter because the "greedy choice" is only improved with an increasing set of observations. Therefore, the modified algorithm $\mathtt{alg}_2$ is monotone by (19).

By definition of the "greedy choice," $\mathtt{rew}_1(n) \leq \mathtt{rew}'_1(n)$ for all steps $n$. Moreover, by Lemma 4.4, $\mathtt{alg}_1$ has a strictly smaller $\mathtt{rew}(n_0)$ compared to DynamicGreedy; so, $\mathtt{rew}_1(n_0) < \mathtt{rew}_2(n_0)$.

Let $\mathtt{alg}$ denote a copy of $\mathtt{alg}_1$ that is running "inside" the modified algorithm $\mathtt{alg}_2$. Let $m_2(t)$ be the number of global rounds before $t$ in which the agent chooses principal 2 *and* $\mathtt{alg}$ is invoked; in other words, it is the number of agents seen by $\mathtt{alg}$ before global round $t$. Let $\mathcal{M}_{2,t}$ be the agents' posterior distribution for $m_2(t)$.

We claim that in each global round $t \geq n_0$, distribution $\mathcal{M}_{2,t}$ stochastically dominates distribution $\mathcal{N}_{1,t}$, and $\mathtt{PMR}_1(t) < \mathtt{PMR}_2(t)$. We use induction on $t$. The base case $t = n_0$ holds because $\mathcal{M}_{2,t} = \mathcal{N}_{1,t}$ (because the two algorithms coincide on the first $n_0 - 1$ steps), and $\mathtt{PMR}_1(n_0) < \mathtt{PMR}_2(n_0)$ is proved as in (9), using the fact that $\mathtt{rew}_1(n_0) < \mathtt{rew}_2(n_0)$.

The induction step is proved as follows. The induction hypothesis for global round $t-1$ implies that agent $t-1$ is seen by $\mathtt{alg}$ with probability $(1-\epsilon_0)(1-p)$, which is strictly larger than $\epsilon_0$, the probability with which this agent is seen by $\mathtt{alg}_2$. Therefore, $\mathcal{M}_{2,t}$ stochastically dominates $\mathcal{N}_{1,t}$.

$$\begin{aligned}
\mathtt{PMR}_1(t) &= \mathop{\mathbb{E}}_{n\sim\mathcal{N}_{1,t}} [\mathtt{rew}_1(n+1)] \\
&\leq \mathop{\mathbb{E}}_{m\sim\mathcal{M}_{2,t}} [\mathtt{rew}_1(m+1)] \qquad (20)\\
&< \mathop{\mathbb{E}}_{m\sim\mathcal{M}_{2,t}} \left[(1-p)\cdot\mathtt{rew}_1(m+1) + p\cdot\mathtt{rew}'_1(m+1)\right] \qquad (21)\\
&= \mathtt{PMR}_2(t).
\end{aligned}$$

Here inequality (20) holds because $\mathtt{rew}_1(\cdot)$ is monotone and $\mathcal{M}_{2,t}$ stochastically dominates $\mathcal{N}_{1,t}$, and inequality (21) holds because $\mathtt{rew}_1(n_0) < \mathtt{rew}_2(n_0)$ and $\mathcal{M}_{2,t}(n_0) > 0$.[7] □

## 6 SoftMax response function

This section is devoted to the SoftMax model. We recover a positive result under the assumptions from Theorem 5.1 (albeit with a weaker conclusion), and then proceed to a much more challenging

[7] If $\mathtt{rew}_1(\cdot)$ is strictly increasing, then inequality (20) is strict, too; this is because $\mathcal{M}_{2,t}(t-1) > \mathcal{N}_{1,t}(t-1)$.

result under weaker assumptions. We start with a formal definition:

**Definition 6.1.** A response function $f_{\mathtt{resp}}$ is $\mathtt{SoftMax}$ if the following conditions hold:

- $f_{\mathtt{resp}}(\cdot)$ is bounded away from 0 and 1: $f_{\mathtt{resp}}(\cdot) \in [\epsilon, 1-\epsilon]$ for some $\epsilon \in (0, \frac{1}{2})$,
- the response function $f_{\mathtt{resp}}(\cdot)$ is "smooth" around 0:

$$\exists \text{ constants } \delta_0, c_0, c_0' > 0 \qquad \forall x \in [-\delta_0, \delta_0] \qquad c_0 \leq f'_{\mathtt{resp}}(x) \leq c_0'. \tag{22}$$

- fair tie-breaking: $f_{\mathtt{resp}}(0) = \frac{1}{2}$.

*Remark* 6.2. This definition is fruitful when parameters $c_0$ and $c_0'$ are close to $\frac{1}{2}$. Throughout, we assume that $\mathtt{alg}_1$ is better than $\mathtt{alg}_2$, and obtain results parameterized by $c_0$. By symmetry, one could assume that $\mathtt{alg}_2$ is better than $\mathtt{alg}_1$, and obtain similar results parameterized by $c_0'$.

Our first result is a version of Theorem 5.1, with the same assumptions about the algorithms and essentially the same proof. The conclusion is much weaker: we can only guarantee that each agent $t \geq n_0$ chooses principal 1 with probability slightly larger than $\frac{1}{2}$. This is essentially unavoidable in a typical case when both algorithms satisfy $\mathtt{BIR}(n) \to 0$, by Definition 6.1.

**Theorem 6.3.** *Assume* $\mathtt{SoftMax}$ *response function. Suppose* $\mathtt{alg}_1$ *has better* $\mathtt{BIR}$ *in the sense of* (15), *and* $\mathtt{alg}_2$ *satisfies the condition* (16). *Then each agent* $t \geq n_0$ *chooses principal* 1 *with probability*

$$\Pr[i_t = 1] \geq \tfrac{1}{2} + \tfrac{c_0}{4}\, \mathtt{BIR}_2(t). \tag{23}$$

*Proof Sketch.* We follow the steps in the proof of Theorem 5.1 to derive

$$\mathtt{PMR}_1(t) - \mathtt{PMR}_2(t) \geq \mathtt{BIR}_2(t)/2 - q, \quad \text{where } q = \exp(-\epsilon_0 t/12).$$

This is at least $\mathtt{BIR}_2(t)/4$ by (16). Then (23) follows by the smoothness condition (22). □

We recover a version of Corollary 5.3, if each principal's utility is the number of users (rather than the more general model in (5)). We also need a mild technical assumption that cumulative Bayesian regret ($\mathtt{BReg}$) tends to infinity. $\mathtt{BReg}$ is a standard notion from the literature (along with $\mathtt{BIR}$):

$$\mathtt{BReg}(n) := n \cdot \mathop{\mathbb{E}}_{\mu \sim \mathcal{P}_{\text{mean}}}\left[\max_{a \in A} \mu_a\right] - \sum_{n=1}^{n} \mathtt{rew}(n') = \sum_{n'=1}^{n} \mathtt{BIR}(n'). \tag{24}$$

**Corollary 6.4.** *Assume that the response function is* $\mathtt{SoftMax}$, *and each principal's utility is the number of users. Consider the restricted competition game with special algorithm* $\mathtt{alg}$, *and assume that all other allowed algorithms satisfy* $\mathtt{BReg}(n) \to \infty$. *Then, for any sufficiently large time horizon* $T$, *this game has a unique Nash equilibrium: both principals choose* $\mathtt{alg}$.

Further, we prove a much more challenging result in which the condition (15) is replaced with a much weaker "BIR-dominance" condition. For clarity, we will again assume that both algorithms are well-defined for an infinite time horizon. The *weak BIR dominance* condition says there exist constants $\beta_0, \alpha_0 \in (0, 1/2)$ and $n_0$ such that

$$(\forall n \geq n_0) \quad \frac{\mathtt{BIR}_1((1-\beta_0)\, n)}{\mathtt{BIR}_2(n)} < 1 - \alpha_0. \tag{25}$$

If this holds, we say that $\texttt{alg}_1$ *weakly BIR-dominates* $\texttt{alg}_2$. Note that the condition (18) involves sufficiently small multiplicative factors (resp., $\epsilon_0/2$ and $\frac{1}{2}$), the new condition replaces them with factors that can be arbitrarily close to 1.

We make a mild assumption on $\texttt{alg}_1$ that its $\mathsf{BIR}_1(n)$ tends to 0. Formally, for any $\epsilon > 0$, there exists some $n(\epsilon)$ such that

$$(\forall n \geq n(\epsilon)) \qquad \mathsf{BIR}_1(n) \leq \epsilon. \tag{26}$$

We also require a slightly stronger version of the technical assumption (16):for some $n_0$,

$$(\forall n \geq n_0) \qquad \mathsf{BIR}_2(n) \geq \frac{4}{\alpha_0} \exp\left(\frac{-\min\{\epsilon_0, 1/8\}n}{12}\right) \tag{27}$$

**Theorem 6.5.** *Assume the* $\texttt{SoftMax}$ *response function. Suppose* $\texttt{alg}_1$ *weakly-BIR-dominates* $\texttt{alg}_2$, $\texttt{alg}_1$ *satisfies* (26), *and* $\texttt{alg}_2$ *satisfies* (27). *Then there exists some* $t_0$ *such that each agent* $t \geq t_0$ *chooses principal* 1 *with probability*

$$\Pr[i_t = 1] \geq \tfrac{1}{2} + \tfrac{c_0 \alpha_0}{4}\, \mathsf{BIR}_2(t). \tag{28}$$

The main idea behind our proof is that even though $\texttt{alg}_1$ may have a slower rate of learning in the beginning, it will gradually catch up and surpass $\texttt{alg}_2$. We will describe this process in two phases. In the first phase, $\texttt{alg}_1$ receives a random agent with probability at least $f_{\text{resp}}(-1) = \epsilon_0$ in each round. Since $\mathsf{BIR}_1$ tends to 0, the difference in BIRs between the two algorithms is also diminishing. Due to the $\texttt{SoftMax}$ response function, $\texttt{alg}_1$ attracts each agent with probability at least $1/2 - O(\beta_0)$ after a sufficient number of rounds. Then the game enters the second phase: both algorithms receive agents at a rate close to $\frac{1}{2}$, and the fractions of agents received by both algorithms — $n_1(t)/t$ and $n_2(t)/t$ — also converge to $\frac{1}{2}$. At the end of the second phase and in each global round afterwards, the counts $n_1(t)$ and $n_2(t)$ satisfy the weak BIR-dominance condition, in the sense that they both are larger than $n_0$ and $n_1(t) \geq (1-\beta_0)\, n_2(t)$. At this point, $\texttt{alg}_1$ actually has smaller BIR, which reflected in the PMRs eventually. Accordingly, from then on $\texttt{alg}_1$ attracts agents at a rate slightly larger than $\frac{1}{2}$. We prove that the "bump" over $\frac{1}{2}$ is at least on the order of $\mathsf{BIR}_2(t)$.

*Proof of Theorem 6.5.* Let $\beta_1 = \min\{c_0'\delta_0, \beta_0/20\}$ with $\delta_0$ defined in (22). Recall each agent chooses $\texttt{alg}_1$ with probability at least $f_{\text{resp}}(-1) = \epsilon_0$. By condition (26) and (27), there exists some sufficiently large $T_1$ such that for any $t \geq T_1$, $\mathsf{BIR}_1(\epsilon_0 T_1/2) \leq \beta_1/c_0'$ and $\mathsf{BIR}_2(t) > e^{-\epsilon_0 t/12}$. Moreover, for any $t \geq T_1$, we know $\mathbb{E}[n_1(t+1)] \geq \epsilon_0\, t$, and by the Chernoff Bounds (Theorem 3.1), we have $n_1(t+1) \geq \epsilon_0 t/2$ holds with probability at least $1 - q_1(t)$ with $q_1(t) = \exp(-\epsilon_0 t/12) < \mathsf{BIR}_2(t)$. It follows that for any $t \geq T_1$,

$$\begin{aligned}
\mathsf{PMR}_2(t) - \mathsf{PMR}_1(t) &= \mathop{\mathbb{E}}_{m_1 \sim \mathcal{N}_{1,t},\, m_2 \sim \mathcal{N}_{2,t}} [\mathsf{BIR}_1(m_1+1) - \mathsf{BIR}_2(m_2+1)] \\
&\leq q_1(t) + \mathop{\mathbb{E}}_{m_1 \sim \mathcal{N}_{1,t}} [\mathsf{BIR}_1(m_1+1) \mid m_1 \geq \epsilon_0 t/2 - 1] - \mathsf{BIR}_2(t) \\
&\leq \mathsf{BIR}_1(\epsilon_0 T_1/2) \leq \beta_1/c_0'
\end{aligned}$$

Since the response function $f_{\text{resp}}$ is $c_0'$-Lipschitz in the neighborhood of $[-\delta_0, \delta_0]$, each agent after round $T_1$ will choose $\texttt{alg}_1$ with probability at least

$$p_t \geq \frac{1}{2} - c_0'\,(\mathsf{PMR}_2(t) - \mathsf{PMR}_1(t)) \geq \frac{1}{2} - \beta_1.$$

Next, we will show that there exists a sufficiently large $T_2$ such that for any $t \geq T_1 + T_2$, with high probability $n_1(t) > \max\{n_0, (1-\beta_0)n_2(t)\}$, where $n_0$ is defined in (25). Fix any $t \geq T_1 + T_2$. Since each agent chooses $\texttt{alg}_1$ with probability at least $1/2 - \beta_1$, by Chernoff Bounds (Theorem 3.1) we have with probability at least $1 - q_2(t)$ that the number of agents that choose $\texttt{alg}_1$ is at least $\beta_0(1/2 - \beta_1)t/5$, where the function

$$q_2(x) = \exp\left(\frac{-(1/2-\beta_1)(1-\beta_0/5)^2 x}{3}\right).$$

Note that the number of agents received by $\texttt{alg}_2$ is at most $T_1 + (1/2+\beta_1)t + (1/2-\beta_1)(1-\beta_0/5)t$.

Then as long as $T_2 \geq \frac{5T_1}{\beta_0}$, we can guarantee that $n_1(t) > n_2(t)(1-\beta_0)$ and $n_1(t) > n_0$ with probability at least $1 - q_2(t)$ for any $t \geq T_1 + T_2$. Note that the weak BIR-dominance condition in (25) implies that for any $t \geq T_1 + T_2$ with probability at least $1 - q_2(t)$,

$$\mathtt{BIR}_1(n_1(t)) < (1-\alpha_0)\mathtt{BIR}_2(n_2(t)).$$

It follows that for any $t \geq T_1 + T_2$,

$$\begin{aligned}
\mathtt{PMR}_1(t) - \mathtt{PMR}_2(t) &= \mathop{\mathbb{E}}_{m_1\sim\mathcal{N}_{1,t},\, m_2\sim\mathcal{N}_{2,t}}\left[\mathtt{BIR}_2(m_2+1) - \mathtt{BIR}_1(m_1+1)\right] \\
&\geq (1-q_2(t))\alpha_0\mathtt{BIR}_2(t) - q_2(t) \\
&\geq \alpha_0\mathtt{BIR}_2(t)/4
\end{aligned}$$

where the last inequality holds as long as $q_2(t) \leq \alpha_0\mathtt{BIR}_2(t)/4$, and is implied by the condition in (27) as long as $T_2$ is sufficiently large. Hence, by the definition of our $\mathtt{SoftMax}$ response function and assumption in (22), we have

$$\Pr[i_t = 1] \geq \frac{1}{2} + \frac{c_0\alpha_0\mathtt{BIR}_2(t)}{4}. \qquad \square$$

Similar to the condition (15), we can also generalize the weak BIR-dominance condition (25) to handle the dependence on $T$: there exist some $T_0$, a function $n_0(T) \in \text{polylog}(T)$, and constants $\beta_0, \alpha_0 \in (0, 1/2)$, such that

$$(\forall T \geq T_0, n \geq n_0(T)) \quad \frac{\mathtt{BIR}_1((1-\beta_0)n \mid T)}{\mathtt{BIR}_2(n \mid T)} < 1 - \alpha_0. \tag{29}$$

We also provide a version of Theorem 6.3 under this more general weak BIR-dominance condition; its proof is very similar and is omitted. The following is just a direct consequence of Theorem 6.3 with this general condition.

**Corollary 6.6.** *Assume that the response function is* $\mathtt{SoftMax}$, *and each principal's utility is the number of users. Consider the restricted competition game in which the special algorithm* $\texttt{alg}$ *weakly-BIR-dominates the other allowed algorithms, and the latter satisfy* $\mathtt{BReg}(n) \to \infty$. *Then, for any sufficiently large time horizon* $T$, *there is a unique Nash equilibrium: both principals choose* $\texttt{alg}$.

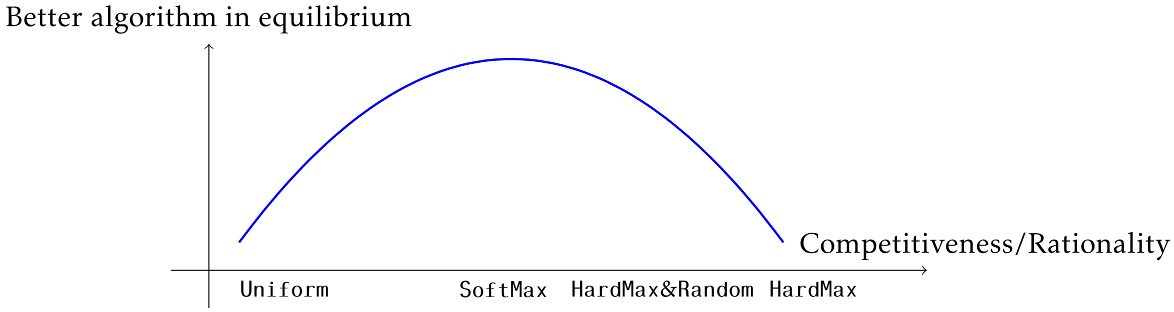


Figure 3: The stylized inverted-U relationship in the "main story".

# 7 Economic implications

We frame our contributions in terms of the relationship between *competitiveness* and *rationality* on one side, and adoption of better algorithms on the other. Recall that both *competitiveness* (of the game between the two principals) and *rationality* (of the agents) are controlled by the response function $f_{\texttt{resp}}$.

**Main story.** Our main story concerns the restricted competition game between the two principals where one allowed algorithm `alg` is "better" than the others. We track whether and when `alg` is chosen in an equilibrium. We vary *competitiveness/rationality* by changing the response function from `HardMax` (full rationality, very competitive environment) to `HardMax&Random` to `SoftMax` (less rationality and competition). Our conclusions are as follows:

- Under `HardMax`, no innovation: `DynamicGreedy` is chosen over `alg`.
- Under `HardMax&Random`, some innovation: `alg` is chosen as long as it BIR-dominates.
- Under `SoftMax`, more innovation: `alg` is chosen as long as it weakly-BIR-dominates.[8]

These conclusions follow, respectively, from Corollaries 4.2, 5.3 and 6.4. Further, we consider the uniform choice between the principals. It corresponds to the least amount of rationality and competition, and (when principals' utility is the number of agents) uniform choice provides no incentives to innovate.[9] Thus, we have an inverted-U relationship, see Figure 3.

**Secondary story.** Let us zoom in on the symmetric `HardMax&Random` model. Competitiveness and rationality within this model are controlled by the baseline probability $\epsilon_0 = f_{\texttt{resp}}(-1)$, which goes smoothly between the two extremes of `HardMax` ($\epsilon_0 = 0$) and the uniform choice ($\epsilon_0 = \frac{1}{2}$). Smaller $\epsilon_0$ corresponds to increased rationality and increased competitiveness. For clarity, we assume that principal's utility is the number of agents.

We consider the marginal utility of switching to a better algorithm. Suppose initially both principals use some algorithm `alg`, and principal 1 ponders switching to another algorithm `alg'` which BIR-dominates `alg`. We are interested in the marginal utility of this switch. Then:

[8] This is a weaker condition, the better algorithm is chosen in a broader range of scenarios.

[9] On the other hand, if principals' utility is somewhat aligned with agents' welfare, as in (5), then a monopolist principal is incentivized to choose the best possible MAB algorithm (namely, to minimize cumulative Bayesian regret $\mathtt{BReg}(T)$). Accordingly, monopoly would result in better social welfare than competition, as the latter is likely to split the market and cause each principal to learn more slowly. This is a very generic and well-known effect regarding economies of scale.

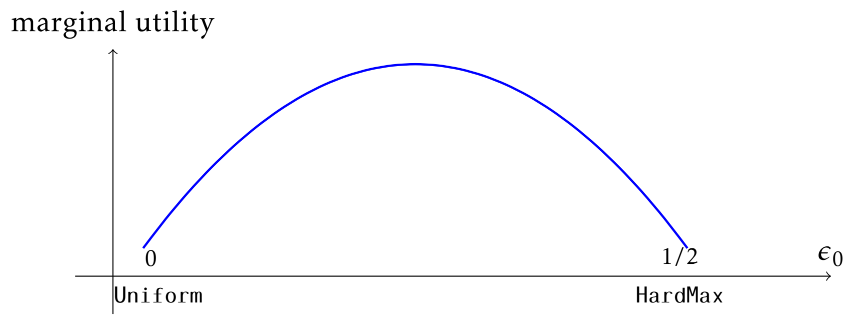


Figure 4: The stylized inverted-U relationship from the "secondary story"

- $\epsilon_0 = 0$ (HardMax): the marginal utility can be negative if alg is DynamicGreedy.
- $\epsilon_0$ near 0: only a small marginal utility can be guaranteed, as it may take a long time for alg′ to "catch up" with alg, and hence less time to reap the benefits.
- "medium-range" $\epsilon_0$: large marginal utility, as alg′ learns fast and gets most agents.
- $\epsilon_0$ near $\frac{1}{2}$: small marginal utility, as principal 1 gets most agents for free no matter what.

The familiar inverted-U shape is depicted in Figure 4.

## Acknowledgements

The authors would like to thank Glen Weyl for discussions of related work in economics.

## References

Alekh Agarwal, Sarah Bird, Markus Cozowicz, Miro Dudik, John Langford, Lihong Li, Luong Hoang, Dan Melamed, Siddhartha Sen, Robert Schapire, and Alex Slivkins. Multiworld testing: A system for experimentation, learning, and decision-making, 2016. A white paper, available at https://github.com/Microsoft/mwt-ds/raw/master/images/MWT-WhitePaper.pdf.

Philippe Aghion, Nicholas Bloom, Richard Blundell, Rachel Griffith, and Peter Howitt. Competition and innovation: An inverted u relationship. *Quaterly J. of Economics*, 120(2):701–728, 2005.

Susan Athey and Ilya Segal. An efficient dynamic mechanism. *Econometrica*, 81(6):2463–2485, November 2013. A preliminary version has been available as a working paper since 2007.

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002a.

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1):48–77, 2002b. Preliminary version in *36th IEEE FOCS*, 1995.

Eduardo Azevedo and Daniel Gottlieb. Perfect competition in markets with adverse selection. *Econometrica*, 85(1):67–105, 2017.

Moshe Babaioff, Yogeshwer Sharma, and Aleksandrs Slivkins. Characterizing truthful multi-armed bandit mechanisms. *SIAM J. on Computing (SICOMP)*, 43(1):194–230, 2014. Preliminary version in *10th ACM EC*, 2009.

Moshe Babaioff, Robert Kleinberg, and Aleksandrs Slivkins. Truthful mechanisms with implicit payment computation. *J. of the ACM*, 62(2):10, 2015. Subsumes the conference papers in *ACM EC 2010* and *ACM EC 2013*.

Gal Bahar, Rann Smorodinsky, and Moshe Tennenholtz. Economic recommendation systems. In *16th ACM Conf. on Electronic Commerce (EC)*, 2016.

Dirk Bergemann and Juuso Välimäki. The dynamic pivot mechanism. *Econometrica*, 78(2):771–789, 2010. Preliminary versions have been available since 2006, as *Cowles Foundation Discussion Papers* #1584 (2006), #1616 (2007) and #1672(2008).

Kostas Bimpikis, Yiangos Papanastasiou, and Nicos Savva. Crowdsourcing exploration. *Management Science*, 2017. Forthcoming.

Patrick Bolton and Christopher Harris. Strategic Experimentation. *Econometrica*, 67(2):349–374, 1999.

Sébastien Bubeck and Nicolo Cesa-Bianchi. Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. *Foundations and Trends in Machine Learning*, 5(1), 2012.

Yeon-Koo Che and Johannes Hörner. Optimal design for social learning. Preprint, 2015. First draft: 2013.

Nikhil Devanur and Sham M. Kakade. The price of truthfulness for pay-per-click auctions. In *10th ACM Conf. on Electronic Commerce (EC)*, pages 99–106, 2009.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *J. of Machine Learning Research (JMLR)*, 7: 1079–1105, 2006.

Peter Frazier, David Kempe, Jon M. Kleinberg, and Robert Kleinberg. Incentivizing exploration. In *ACM Conf. on Economics and Computation (ACM EC)*, pages 5–22, 2014.

Xavier Gabaix, David Laibson, Deyuan Li, Hongyi Li, Sidney Resnick, and Casper G. de Vries. The impact of competition on prices with numerous firms. *J. of Economic Theory*, 165:1–24, 2016.

Arpita Ghosh and Patrick Hummel. Learning and incentives in user-generated content: multi-armed bandits with endogenous arms. In *Innovations in Theoretical Computer Science Conf. (ITCS)*, pages 233–246, 2013.

John Gittins, Kevin Glazebrook, and Richard Weber. *Multi-Armed Bandit Allocation Indices*. John Wiley & Sons, 2011.

Ramakrishna Gummadi, Ramesh Johari, and Jia Yuan Yu. Mean field equilibria of multiarmed bandit games. In *13th ACM Conf. on Electronic Commerce (EC)*, 2012.

Chien-Ju Ho, Aleksandrs Slivkins, and Jennifer Wortman Vaughan. Adaptive contract design for crowdsourcing markets: Bandit algorithms for repeated principal-agent problems. *J. of Artificial Intelligence Research*, 55:317–359, 2016. Preliminary version appeared in *ACM EC 2014*.

Harold Hotelling. Stability in competition. *The Economic Journal*, 39(153):41–57, 1929.

Nicole Immorlica, Adam Tauman Kalai, Brendan Lucier, Ankur Moitra, Andrew Postlewaite, and Moshe Tennenholtz. Dueling algorithms. In *43rd ACM Symp. on Theory of Computing (STOC)*, pages 215–224, 2011.

Sham M. Kakade, Ilan Lobel, and Hamid Nazerzadeh. Optimal dynamic mechanism design and the virtual-pivot mechanism. *Operations Research*, 61(4):837–854, 2013.

Godfrey Keller, Sven Rady, and Martin Cripps. Strategic Experimentation with Exponential Bandits. *Econometrica*, 73(1):39–68, 2005.

Robert D. Kleinberg, Bo Waggoner, and E. Glen Weyl. Descending price optimally coordinates search. Working paper, 2016. Preliminary version in *ACM EC 2016*. Under submission to *Econometrica*.

Ilan Kremer, Yishay Mansour, and Motty Perry. Implementing the wisdom of the crowd. *J. of Political Economy*, 122:988–1012, 2014. Preliminary version in *ACM EC 2014*.

Tze Leung Lai and Herbert Robbins. Asymptotically efficient Adaptive Allocation Rules. *Advances in Applied Mathematics*, 6:4–22, 1985.

Yishay Mansour, Aleksandrs Slivkins, and Vasilis Syrgkanis. Bayesian incentive-compatible bandit exploration. In *15th ACM Conf. on Electronic Commerce (EC)*, 2015.

Yishay Mansour, Aleksandrs Slivkins, Vasilis Syrgkanis, and Steven Wu. Bayesian exploration: Incentivizing exploration in bayesian games. Working paper, 2016. available at `https://arxiv.org/abs/1602.07570`. Preliminary version in *ACM EC 2016*.

Paul Milgrom and Nancy Stokey. Information, trade and common knowledge. *J. of Economic Theory*, 26(1): 17–27, 1982.

Michael Mitzenmacher and Eli Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.

Jeffrey M. Perloff and Steven C. Salop. Equilibrium with product differentiation. *Review of Economic Studies*, LII:107–120, 1985.

Michael Rothschild and Joseph Stiglitz. Equilibrium in competitive insurance markets: An essay on the economics of imperfect information. *Quaterly J. of Economics*, 90(4):629–649, 1976.

Marc Rysman. The economics of two-sided markets. *J. of Economic Perspectives*, 23(3):125–144, 2009.

Joseph Schumpeter. *Capitalism*, *Socialism and Democracy*. Harper & Brothers, 1942.

Adish Singla and Andreas Krause. Truthful incentives in crowdsourcing tasks using regret minimization mechanisms. In *22nd Intl. World Wide Web Conf. (WWW)*, pages 1167–1178, 2013.

Aleksandrs Slivkins. Introduction to multi-armed bandits, 2017. A book draft, available at http://research.microsoft.com/en-us/people/slivkins.

Andre Veiga and Glen Weyl. Product design in selection markets. *Quarterly J. of Economics*, 131(2):1007–1056, 2016.

Xavier Vives. Innovation and competitive pressure. *J. of Industrial Economics*, 56(3), 2008.

Glen Weyl and Alexander White. Let the right one' win: Policy lessons from the new economics of platforms. *Competition Policy International*, 12(2):29–51, 2014.

Yisong Yue and Thorsten Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In *26th Intl. Conf. on Machine Learning (ICML)*, pages 1201–1208, 2009.

Yisong Yue, Josef Broder, Robert Kleinberg, and Thorsten Joachims. The k-armed dueling bandits problem. *J. Comput. Syst. Sci.*, 78(5):1538–1556, 2012. Preliminary version in COLT 2009.

# A Background on multi-armed bandits

This appendix provides some pertinent background on multi-armed bandits (*MAB*). We discuss BIR and monotonicity of several MAB algorithms, touching upon: DynamicGreedy and StaticGreedy (Section A.1), "naive" MAB algorithms that separate exploration and exploitation (Section A.2), and "smart" MAB algorithms that combine exploration and exploitation (Section A.3).

As we do throughout the paper, we focus on MAB with i.i.d. rewards and a Bayesian prior; we call it *Bayesian MAB* for brevity.

## A.1 DynamicGreedy and StaticGreedy

We provide an example when DynamicGreedy and StaticGreedy have constant BIR, and prove monotonicity of DynamicGreedy. For the example, it suffices to consider *deterministic rewards* (for each action $a$, the realized reward is always equal to the mean $\mu_a$) and *independent priors* (according to the prior $\mathcal{P}_{\mathtt{mean}}$, random variables $\mu_1, \dots, \mu_K$ are mutually independent) each of *full support*.

The following claim is immediate from the definition of the CDF function

**Claim A.1.** *Assume independent priors. Let $F_i$ be the CDF of the mean reward $\mu_i$ of action $a_i \in A$. Then, for any numbers $z_2 > z_1 > \mathbb{E}[\mu_2]$ we have $\Pr[\mu_1 \le z_1$ and $\mu_2 \ge z_2] = F_1(z_1)(1 - F_2(z_2))$.*

We can now draw an immediate corollary of the above claim

**Corollary A.2.** *Consider any problem instance of Bayesian MAB with two actions and independent priors which are full support. Then:*

*(a) With constant probability,* StaticGreedy *has a constant* BIR *for all steps.*

*(b) Assuming deterministic rewards, with constant probability* DynamicGreedy *has a constant* BIR *for all steps.*

*Remark* A.3. A similar result holds for rewards which are distributed as Bernoulli random variables. In this case we consider accumulative reward of an action as a random walk, and use a high probability variation of the law of iterated logarithms. (Details omitted.)

Next, we show that DynamicGreedy is monotone.

**Lemma A.4.** DynamicGreedy *is monotone, in the sense that* $\mathtt{rew}(n)$ *is non-decreasing. Further,* $\mathtt{rew}(n)$ *is strictly increasing for every time step $n$ with* $\Pr[a_n \neq a_{n+1}] > 0$.

*Proof.* We prove by induction on $n$ that $\mathtt{rew}(n) \le \mathtt{rew}(n+1)$ for DynamicGreedy. Let $a_n$ be the random variable recommended at time $t$, then $\mathbb{E}[\mu_{a_n} | \mathcal{I}_n] = \mathtt{rew}(n)$. We can rewrite this as:

$$\mathtt{rew}(n) = \mathop{\mathbb{E}}_{\mathcal{I}_n}[\mathop{\mathbb{E}}_{r_n}[\mu_{a_n} | r_n, \mathcal{I}_n]] = \mathop{\mathbb{E}}_{\mathcal{I}_{n+1}}[\mu_{a_n} | \mathcal{I}_{n+1}]$$

since $\mathcal{I}_{n+1} = (\mathcal{I}_n, r_n)$. At time $n+1$ DynamicGreedy will select an action $a_{n+1}$ such that:

$$\mathtt{rew}(n+1) = \mathbb{E}[\mu_{a_{n+1}} | \mathcal{I}_{n+1}] \ge \mathbb{E}[\mu_{a_n} | \mathcal{I}_n] = \mathtt{rew}(n)$$

which proves the monotonicity. In cases that $\Pr[a_n \neq a_{n+1}] > 0]$ we have a strict inequality, since with some probability we select a better action then the realization of $a_n$. □

## A.2 "Naive" MAB algorithms that separate exploration and exploitation

MAB algorithm ExplorExploit ($m$) initially explores each action with $m$ agents and for the remaining $T - |A|m$ agents recommends the action with the highest observed average. In the explore phase it assigns a random permutation of the $mK$ recommendations.

**Lemma A.5.** *The* ExplorExploit ($T^{2/3} \log|A|/\delta$) *algorithm has, with probability* $1-\delta$*, for any* $n \geq |A|T^{2/3}$ *we have* BIR $(n) = O(T^{-1/3})$*. In addition,* ExplorExploit ($m$) *is monotone.*

*Proof.* In the explore phase we we approximate for each action $a \in A$, the value of $\mu_a$ by $\hat{\mu}_a$. Using the standard Chernoff bounds we have that with probability $1-\delta$, for every action $a \in A$ we have $|\mu_a - \hat{\mu}_a| \leq T^{-1/3}$.

Let $a^* = \arg\max_a \mu_a$ and $a^{ee}$ the action that ExplorExploit selects in the explore phase after the first $|A|T^{2/3}$ agents. Since $\hat{\mu}_{a^*} \leq \hat{\mu}_{a^{ee}}$, this implies that $\mu_{a^*} - \mu_{a^{ee}} = O(T^{-1/3})$.

To show that ExplorExploit ($m$) is monotone, we need to show only that $\mathtt{rew}(mK) \leq \mathtt{rew}(mK+1)$. This follows since for any $t < mK$ we have $\mathtt{rew}(t) = \mathtt{rew}(t+1)$, since the recommended action is uniformly distributed for each time $t$. Also, for any $t \geq mK+1$ we have $\mathtt{rew}(t) = \mathtt{rew}(t+1)$ since we are recommending the same exploration action. The proof that $\mathtt{rew}(mK) \leq \mathtt{rew}(mK+1)$ is the same as for DynamicGreedy in Lemma A.4. □

We can also have a a phased version which we call PhasedExplorExploit ($m_t$), where time is partition in to phases. In phase $t$ we have $m_t$ agents and a random subset of $K$ explore the actions (each action explored by a single agent) and the other agents exploit. (This implies that we need that $m_t \geq K$ for all $t$. We also assume that $m_t$ is monotone in $t$.)

**Lemma A.6.** *Consider the case that* $K = 2$ *and the rewards of the actions are Bernoulli r.v. with parameter* $\mu_i$ *and* $\Delta = \mu_1 - \mu_2$*. Algorithm* PhasedExplorExploit ($m_t$) *is monotone and for* $m_t = \sqrt{t}$ *it has* $\mathrm{BIR}(n) = O(n^{-1/3} + e^{-O(\Delta^2 n^{2/3})})$*).*

*Proof.* We first show that it is monotone. Recall that $\mu_1 > \mu_2$. Let $S_i = \sum_{j=1}^{t} r_{i,j}$ be the sum of the rewards of action $i$ up to phase $t$. We need to show that $\Pr[S_1 > S_2] + (1/2)\Pr[S_1 = S_2]$ is monotonically increasing in $t$. Consider the random variable $Z = S_1 - S_2$. At each phase it increases by +1 with probability $\mu_1(1-\mu_2)$, decreases by $-1$ with probability $(1-\mu_1)\mu_2$ and otherwise does not change.

Consider the values of $Z$ up to phase $t$. We really care only about the probability that is shifted from positive to negative and vice versa.

First, consider the probability that $Z = 0$. We can partition it to $S_1 = S_2 = r$ events, and let $p(r,r)$ be the probability of this event. For each such event, we have $p(r,r)\mu_1$ moved to $Z = +1$ and $p(r,r)\mu_2$ moved to $Z = -1$. Since $\mu_1 > \mu_2$ we have that $p(r,r)\mu_1 \geq p(r,r)\mu_2$ (note that $p(r,r)$ might be zero, so we do not have a strict inequality).

Second, consider the probability that $Z = +1$ or $Z = -1$. We can partition it to $S_1 = r+1; S_2 = r$ and $S_1 = r; S_2 = r+1$ events, and let $p(r+1,r)$ and $p(r,r+1)$ be the probabilities of those events. It is not hard to see that $p(r+1,r)\mu_2 = p(r,r+1)\mu_1$. This implies that the probability mass moved from $Z = +1$ to $Z = 0$ is identical to that moved from $Z = -1$ to $Z = 0$.

We have showed that $\Pr[S_1 > S_2] + (1/2)\Pr[S_1 = S_2]$ and therefore the expected valued of the exploit action is non-decreasing. Since we have that the size of the phases are increasing, the BIR is strictly increasing between phases and identical within each phase.

We now analyze the BIR regret. Note that agent $n$ is in phase $O(n^{2/3})$ and the length of his phase is $O(n^{1/3})$. The BIR has two parts. The first is due to the exploration, which is at most $O(n^{-1/3})$. The second is due to the probability that we exploit the wrong action. This happens with probability $\Pr[S_1 < S_2] + (1/2)\Pr[S_1 = S_2]$ which we can bound using a Chernoff bound by $e^{-O(\Delta^2 n^{2/3})}$, since we explored each action $O(n^{2/3})$ times. □

*Remark* A.7. Actually we have a tradeoff depending on the parameter $m_t$ between the regret due to exploration and exploitation. (Note that the monotonicity is always guarantee assuming $m_t$ is monotone.) If we can set that $m_t = 2^t$ then at time $n$ we have $2/n$ probability of an exploit action. For the explore action we are in phase $\log n$ so the probability of a sub-optimal explore action is $n^{-O(\Delta^{-2})}$. This should give us $\mathtt{BIR}(n) = O(n^{-O(\Delta^{-2})})$.

### A.3 "Smart" MAB algorithms that combine exploration and exploitation

MAB algorithm `SuccesiveEliminationReset` works as follows. It keeps a set of surviving actions $A_s \subseteq A$, where initially $A_s = A$. The agents are partition into phases, where each phase is a random permutation of the non-eliminated actions. Let $\hat{\mu}_{i,t}$ be the average of the rewards of action $i$ up to phase $t$ and $\hat{\mu}^* = \max_i \hat{\mu}_{i,t}$. We eliminate action $i$ at the end of phase $t$, i.e., delete it from $A_s$, if $\hat{\mu}^*_t - \hat{\mu}_{i,t} > \log(T/\delta)/\sqrt{t}$. In `SuccesiveEliminationReset` we simply reset the algorithm with $A = A_s - A_{e,t}$, where $A_{e,t}$ is the set of eliminated actions after phase $t$. Namely, we restart $\hat{\mu}_{i,t}$ and ignore the old rewards before the elimination.

**Lemma A.8.** *The algorithm* `SuccesiveEliminationReset`*, has, with probability* $1-\delta$, BIR $(n) = O(\log(T/\delta)/\sqrt{n/K})$.

*Proof.* Let the best action be $a^* = \arg\max_a \mu_a$. With probability $1-\delta$ at any time $n$ we have that for any action $i \in A_s$ that $|\hat{\mu}_i - \mu_i| \le \log(T/\delta)/\sqrt{n/K}$, and $a^* \in A_s$. This implies that any action $a$ such $\mu_{a^*} - \mu_a > 3\log(T/\delta)/\sqrt{n/K}$ is eliminated. Therefore, any action in $A_s$ has BIR $(n)$ of at most $6\log(T/\delta)/\sqrt{n/K}$. □

**Lemma A.9.** *Assume that if $\mu_i \ge \mu_j$ then the rewards $r_i$ stochastically dominates the rewards $r_j$. Then,* `SuccesiveEliminationReset` *is monotone*

*Proof.* Consider the first time $T$ an action is eliminated, and let $T = \tau$ be a realized value of $T$. Then, clearly for $n < \tau$ we have $\mathtt{rew}(n) = \mathtt{rew}(1)$ .

Consider two actions $a_1, a_2 \in A$, such that $\mu_{a_1} \ge \mu_{a_2}$. At time $T = \tau$, the probability that $a_1$ is eliminated is smaller than the probability that $a_2$ is eliminated. This follows since $\hat{\mu}_{a_1}$ stochastically dominates $\hat{\mu}_{a_2}$, which implies that for any threshold $\theta$ we have $\Pr[\hat{\mu}_{a_1} \ge \theta] \ge \Pr[\hat{\mu}_{a_2} \ge \theta]$.

After the elimination we consider the expected reward of the eliminated action $\sum_{i \in A} \mu_i q_i$, where $q_i$ is the probability that action $i$ was eliminated in time $T = \tau$. We have that $q_i \le q_{i+1}$, from the probabilities of elimination.

The sum $\sum_{i \in A} \mu_i q_i$ with $q_i \le q_{i+1}$ and $\sum_i q_i = 1$ is maximized by setting $q_i = 1/|A|$. (We can see that if there are $q_i \neq 1/|A|$, then there are two $q_i < q_{i+1}$, and one can see that setting both to $(q_i + q_{i+1})/2$ increases the value.) Therefore we have that the $\mathtt{rew}(\tau) \ge \mathtt{rew}(\tau - 1)$.

Now we can continue by induction. For the induction, we can show the property for *any* remaining set of at most $k-1$ actions. The main issue is that `SuccesiveEliminationReset` restarts from scratch, so we can use induction. □

## B Non-degeneracy via a random perturbation

We show that Assumption (4) holds almost surely under a small random perturbation of the prior. We focus on problem instances with 0-1 rewards, and assume that the prior $\mathcal{P}_{\texttt{mean}}$ is independent across arms and has a finite support.[10] Consider the probability vector in the prior for arm $a$:

$$\vec{p}_a = (\, \Pr[\mu_a = \nu] : \; \nu \in \texttt{support}(\mu_a) \,).$$

We apply a small random perturbation independently to each such vector:

$$\vec{p}_a \leftarrow \vec{p}_a + \vec{q}_a, \quad \text{where} \quad \vec{q}_a \sim \mathcal{N}_a. \tag{30}$$

Here $\mathcal{N}_a$ is the noise distribution for arm $a$: a distribution over real-valued, zero-sum vectors of dimension $d_a = |\texttt{support}(\mu_a)|$. We need the noise distribution to satisfy the following property:

$$\forall x \in [-1,1]^{d_a} \setminus \{0\} \qquad \Pr_{q \sim \mathcal{N}_a} [x \cdot (\vec{p}_a + q) \neq 0] = 1. \tag{31}$$

**Theorem B.1.** *Consider an instance of MAB with 0-1 rewards. Assume that the prior $\mathcal{P}_{\texttt{mean}}$ is independent across arms, and each mean reward $\mu_a$ has a finite support that does not include* 0 *or* 1. *Assume that noise distributions $\mathcal{N}_a$ satisfy property* (31). *If random perturbation* (30) *is applied independently to each arm a, then Eq. 4 holds almost surely for each history h.*

*Remark* B.2. As a generic example of a noise distribution which satisfies Property (31), consider the uniform distribution $\mathcal{N}$ over the bounded convex set

$$Q = \left\{ q \in \mathbb{R}^{d_a} \mid q \cdot \vec{1} = 0 \text{ and } \|q\|_2 \leq \epsilon \right\},$$

where $\vec{1}$ denotes the all-1 vector. If $x = a\vec{1}$ for some non-zero value of $a$, then (31) holds because

$$x \cdot (p+q) = x \cdot p = a \neq 0.$$

Otherwise, denote $p = \vec{p}_a$ and observe that $x \cdot (p+q) = 0$ only if $x \cdot q = c \triangleq x \cdot (-p)$. Since $x \neq \vec{1}$, the intersection $Q \cap \{x \cdot q = c\}$ either is empty or has measure 0 in $Q$, which implies $\Pr_q[x \cdot (p+q) \neq 0] = 1$.

To prove Theorem B.1, it suffices to focus on two arms, and perturb one of them. Since realized rewards have finite support, there are only finitely many possible histories. Therefore, it suffices to focus on a fixed history $h$.

**Lemma B.3.** *Consider an instance of MAB with 0-1 rewards. Assume that the prior $\mathcal{P}_{\texttt{mean}}$ is independent across arms, and that $\texttt{support}(\mu_1)$ is finite and does not include* 0 *or* 1. *Fix history h. Suppose random perturbation* (30) *is applied to arm* 1, *with noise distribution $\mathcal{N}_1$ that satisfies* (31). *Then $\mathbb{E}[\mu_1 \mid h] \neq \mathbb{E}[\mu_2 \mid h]$ almost surely.*

[10] The assumption of 0-1 rewards is for clarity. Our results hold under a more general assumption that for each arm $a$, rewards can only take finitely many values, and each of these values is possible (with positive probability) for every possible value of the mean reward $\mu_a$.

*Proof.* Note that $\mathbb{E}[\mu_a \mid h]$ does not depend on the algorithm which produced this history. Therefore, for the sake of the analysis, we can assume w.l.o.g. that this history has been generated by a particular algorithm, as long as this algorithm can can produce this history with non-zero probability. Let us consider the algorithm that deterministically chooses same actions as $h$.

Let $S = \mathtt{support}(\mu_1)$. Then:

$$\begin{aligned}\mathbb{E}[\mu_1 \mid h] &= \sum_{\nu \in S} \nu \cdot \Pr[\mu_1 = \nu \mid h] = \sum_{\nu \in S} \nu \cdot \Pr[h \mid \mu_1 = \nu] \cdot \Pr[\mu_1 = \nu] \,/\, \Pr[h], \\ \Pr[h] &= \sum_{\nu \in S} \Pr[h \mid \mu_1 = \nu] \cdot \Pr[\mu_1 = \nu].\end{aligned}$$

Therefore, $\mathbb{E}[\mu_1 \mid h] = \mathbb{E}[\mu_2 \mid h]$ if and only if

$$\sum_{\nu \in S} (\nu - C) \cdot \Pr[h \mid \mu_1 = \nu] \cdot \Pr[\mu_1 = \nu] = 0, \quad \text{where} \quad C = \mathbb{E}[\mu_2 \mid h].$$

Since $\mathbb{E}[\mu_2 \mid h]$ and $\Pr[h \mid \mu_1 = \nu]$ do not depend on the probability vector $\vec{p}_1$, we conclude that

$$\mathbb{E}[\mu_1 \mid h] = \mathbb{E}[\mu_2 \mid h] \quad \Leftrightarrow \quad x \cdot \vec{p}_1 = 0,$$

where vector

$$x := (\, (\nu - C) \cdot \Pr[h \mid \mu_1 = \nu] : \ \nu \in S \,) \in [-1, 1]^{d_1}$$

does not depend on $\vec{p}_1$.

Thus, it suffices to prove that $x \cdot \vec{p}_1 \neq 0$ almost surely under the perturbation. In a formula:

$$\Pr_{q \sim \mathcal{N}_1} \left[ x \cdot (\vec{p}_1 + q) \neq 0 \right] = 1 \tag{32}$$

Note that $\Pr[h \mid \mu_1 = \nu] > 0$ for all $\nu \in S$, because $0, 1 \notin S$. It follows that at most one coordinate of $x$ can be zero. So (32) follows from property (31). $\square$