Why adaptively collected data have negative bias and how to correct for it.

Anonymous Author(s)
Affiliation
Address
email

Abstract

From scientific experiments to online A/B testing, the previously observed data often affects how future experiments are performed, which in turn affects which data will be collected. Such adaptivity introduces complex correlations between the data and the collection procedure. In this paper, we prove that when the data collection procedure satisfies natural conditions, then sample means of the data have systematic *negative* biases. As an example, consider an adaptive clinical trial where additional data points are more likely to be tested for treatments that show initial promise. Our surprising result implies that the average observed treatment effects would underestimate the true effects of each treatment. We quantitatively analyze the magnitude and behavior of this negative bias in a variety of settings. We also propose a novel debiasing algorithm based on selective inference techniques. In experiments, our method can effectively reduce bias and estimation error.

1 Introduction

Much of modern data science is driven by data that is collected adaptively. A scientist often starts off testing multiple experimental conditions, and based on the initial results may decide to collect more data points from some conditions and less data from other settings. A sequential clinical trial initially groups the participants into different treatment regimes, and depending on the continuous feedback, may reallocate participants into the more promising treatments. In e-commerce, companies often use online A/B tests to collect user data from multiple variants of a project, and could adaptively collect more data from a subset of the variants (multi-arm bandit algorithms are often used here to decide which variant to collect data from as a function of the data log history).

The key characteristic of adaptively collected data is that the analyst sequentially collects data from multiple alternatives (e.g. different treatments, products, etc.). The choice of which alternative to gather data from at a particular time depends on the previously observed data from all the options. The collected data could be used in many different ways. In some settings, the analyst simply wants to use it to identify the single best alternative, and may not care about the data beyond this goal (this setting motivates many bandit problems). In many other settings, the data itself could be used to estimate various statistical parameters. In the sequential clinical trial example, many scientists would like to use the data to estimate the effects of each of the treatments. Even if the company sponsoring the trials may care most about identifying the best treatment, other scientist using the data may care about the effect size estimates of other treatments in the data for their own applications.

Our contributions. We study the problem of estimation using adaptively collected data. We prove that when the adaptive data collection procedure satisfies two natural conditions (precisely defined in Sec. 2), then the sample mean of the collected data is negatively biased as an estimator for the true mean. This means that the effect size empirically observed is systematically less than the true effect

Submitted to 31st Conference on Neural Information Processing Systems (NIPS 2017). Do not distribute.

size for every alternative. We provide intuition for this counter-intuitive result, and compare and analyze the magnitude of this negative bias across different conditions and collection procedures. We then propose a novel randomized algorithm called the conditional Maximum Likelihood Estimator (cMLE) based on selective inference to reduce this ubiquitous bias, and compare it a simple approach using an independent set of held-out data. We validate the performance of our bias-reduction algorithm in extensive experiments. All the proofs and additional experiments are in the Appendix.

Related works. Multi-arm bandits and its variations are extensively studied in machine learning. The goal of our work is different from that of the standard bandit setting. In bandits, the data sampled from an arm (i.e. one of the alternatives) is considered a reward and the objective is to *design* adaptive algorithms to pick arms so to maximize total reward (or minimize regret). Our goal is not to design such algorithms and we are agnostic to the reward. We take the perspective of an analyst who is given such an adaptively collected dataset and wants to estimate statistical parameters.

Xu et al [20] empirically observed estimation bias due to selection in specific multi-arm bandit algorithms. They were primarily interested in estimating the values of the top two arms, and used data splitting with a held-out set in their experiments to reduce bias. We are the first one to rigorously prove that such underestimation is a general phenomenon. Our cMLE approach builds upon recent advances in selective inference [15, 18], which derives valid confidence intervals accounting for selection effects of the algorithm. Selective inference has been applied to regression problems (e.g. LASSO, Stepwise regression), and has not been considered for the adaptive data collection setting before. We build upon results from recent developments in this area [18, 17, 9].

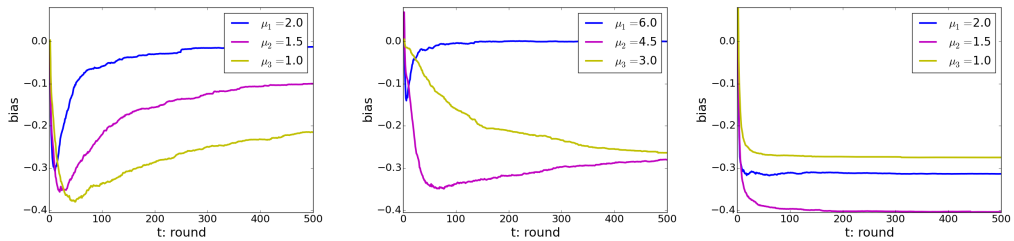
The problem of selection bias has been extensively studied, especially in the context of Winner's Curse in genetic association studies [10]. There is no adaptive data collection component to this selection bias; rather the bias arise from selective reporting. There is a related line of recent work [6] [13] in adaptive data analysis that is complementary to ours. There the data is fixed (and is typically i.i.d.) and the adaptivity is in the analyst. In contrast, in our work the data collection itself is adaptive.

2 Adaptive data collection has negative bias

Model of adaptive data collection. We have K unknown distributions that we would like to collect data from. There are T rounds of data collection and at round $t \in [T]$ the distribution $s_t \in [K]$ is selected, and we draw $X_t^{(s_t)}$, an independent sample, from s_t. The data collection procedure can be modeled by a selection function $s_t = f(\Lambda_t)$, where Λ_t is the history of the observed samples up to time t. More precisely, let $X_i^{(k)}$ denote the i-th sample from distribution k and $N_t^{(k)}$ denote the number of times that distribution k is sampled by round t, which could be a random variable, then $\Lambda_t = \{\{X_1^{(1)}, ..., X_{N_t^{(1)}}^{(1)}\}, ..., \{X_1^{(K)}, ..., X_{N_t^{(K)}}^{(K)}\}\}$. The history of distribution k up to round t is denoted by $\Lambda_t^{(k)} = \{X_1^{(k)}, ..., X_{N_t^{(K)}}^{(k)}\}$. We use $\Lambda_t^{(-k)}$ to denote the history up to round t of all the distributions except for the k-th one; $\Lambda_t^{(-k)} = \{\{X_1^{(i)}, ..., X_{N_t^{(K)}}^{(i)}\}\}_{i \in [K] \setminus k}$. We allow f to be a randomized function, and will sometimes write $f(\Lambda_t, \omega)$, where $\omega \in \Omega$ is a random seed, to highlight this randomness. Let $\overline{X_t^{(k)}} \equiv \frac{\sum_{i=1}^{N_t^{(k)}} X_i^{(k)}}{N_t^{(k)}}$ denote the sample average of distribution k at round t. Appendix B gives examples of the selection function f.

Many adaptive data collection procedures correspond to a selection function f that satisfies two natural properties: *Exploit* and *Independence of Irrelevant Option (IIO)*. *Exploit* means that all else being equal, if distribution k is selected in a scenario where it has lower sample average, then k would also be selected in a scenario where it has higher sample average. *IIO* means that if distribution k is not selected then the precise values observed from k does not affect which of the other distribution is selected. We precisely define these two properties next.

Definition 1 (Exploit). *Given any $t \in [T]$, $k \in [K]$, realization $\Lambda_t^{(-k)}$ and random seed ω. Suppose $\Lambda_t^{(k)}$ and $\Lambda_t^{'(k)}$ are two sample histories of distribution k of length n with sample means $\overline{X_t^{(k)}} \le \overline{X_t^{'(k)}}$. Then $f(\Lambda_t^{(k)} \cup \Lambda_t^{(-k)}, \omega) = k$ implies $f(\Lambda_t^{'(k)} \cup \Lambda_t^{(-k)}, \omega) = k$. In words,* Exploit *states that given*



(a) lil' UCB, μ scale = 1 (b) lil' UCB, μ scale = 3 (c) Greedy, μ scale = 1

Figure 1: In (a-b), we plot the bias of the empirical mean estimates of three unknown distributions running lil' UCB with horizon T=500. Each is distributed according to $\mathcal{N}(\mu_i, 1)$, specified in the legends of the plot. We see that as we scale up μ_i's, so they become more spread out, the bias increases/decreases depending how far the μ_i's are from each other, and what is the order of the distributions. (c) plots the bias of the three unknown distributions running Greedy.

the same context specified by $\Lambda_t^{(-k)}$ and ω, if k is selected when it has smaller sample mean then it should also be selected when it has a larger mean.

Exploit captures the intuition that when we are looking for options that work well, we are more likely to try out the options that show more promise early on. It's easy to show that examples of standard multi-arm bandit algorithms all satisfy *Exploit* (see Proposition. 1).

Definition 2 (Independent of Irrelevant Options (IIO))**.** *Given any $t \in [T]$ and $k \in [K]$. Let $\Lambda_t = \Lambda_t^{(k)} \cup \Lambda_t^{(-k)}$ and $\Lambda_t' = \Lambda_t^{'(k)} \cup \Lambda_t^{(-k)}$, i.e. Λ_t and Λ_t' have the same histories for distributions $i \neq k$ and could have arbitrary histories for distribution k. Then $\forall\, i \neq k$,*

$$\Pr\left[f\left(\Lambda_t\right) = i | f\left(\Lambda_t\right) \neq k\right] = \Pr\left[f\left(\Lambda_t'\right) = i | f\left(\Lambda_t'\right) \neq k\right].$$

In words, so long as k is not chosen, which other distribution is selected depends only on the history $\Lambda_t^{(-k)}$ of those distributions.

Estimation bias. In this paper, we are interested in the fundamental problem of estimating the true mean, $\mu_k = \mathbb{E}[X^{(k)}]$, of each of the distributions given a sample history dataset, Λ_T, which is collected through an adaptive procedure. This models the adaptive clinical trials example, where the scientist is interested in estimating $\{\mu_k\}_{k\in[K]}$, the true effects of the treatments. Of course, if the scientist can collect her own data, she could just collect a non-adaptive set of samples and obtain unbiased estimates of $\{\mu_k\}_{k\in[K]}$. However, in many settings like the clinical trials, the scientist does not collect the data; rather it is adaptively collected by a pharmaceutical company with a different objective of finding an optimal treatment or demonstrating efficacy. The simplest and most common approach is to use the sample average $\overline{X_T^{(k)}}$ to estimate the true mean μ_k. Our main result shows that in expectation, the sample average underestimates the true mean if f satisfies *Exploit* and *IIO*: $\mathbb{E}\left[\overline{X_T^{(k)}}\right] \leq \mu_k, \forall k \in [K]$.

Theorem 1. *Suppose $X^{(k)}, k \in [K]$ is a sample drawn from a distribution with finite mean $\mu_k = \mathbb{E}[X^{(k)}]$, and the selection function f satisfies* Exploit *and* IIO*. Then $\forall k$ and $\forall T$, $\mathbb{E}\left[\overline{X_T^{(k)}}\right] \leq \mu_k$. Moreover, the equality holds only if the number of times distribution k is selected, $N_T^{(k)}$, does not depend on the observed history $\Lambda_T^{(k)}$ of k.*

Many standard multi-arm bandit algorithms can be modeled by a selection function f that satisfies *Exploit* and *IIO*. While Greedy (defined in Appendix B) only has sample mean as its input, upper confidence bound (UCB) type algorithms also account for the number of observations and give preference for the less explored distributions. lil' UCB is the state-of-the-art UCB algorithm [11] and its details are presented in Appendix A.

Proposition 1. *lil' UCB, Greedy, ϵ-Greedy are all equivalent to selection functions $f(\Lambda_t)$ that satisfy* Exploit *and* IIO.

In Appendix I, we extend Proposition 1 to Thompson Sampling [16, 1]. When $K = 2$, we do not need the *IIO* condition in order for the bias to be non-positive.

Proposition 2. *Suppose $X^{(1)}, X^{(2)}$ are samples drawn from distributions with finite means μ_1, μ_2 and the selection function f satifies* Exploit*. Then for $k \in \{1, 2\}$ and all T, $\mathbb{E}\left[\overline{X_T^{(k)}}\right] \le \mu_k$. Moreover the equality holds only if the number of times distribution k is selected, $N_T^{(k)}$, does not depend on observed values $\Lambda_T^{(k)}$ of k.*

We empirically characterize the bias in Figure 1. See Appendix C for more detailed descriptions of experiment setups, and an analytic example with explicit bias.

3 Debiasing algorithms and experiments

Data splitting A simple approach to obtain unbiased estimators of μ_k's is to split the data. Let k be the distribution the selection function f chooses at time t. Instead of taking one sample from k, we maintain a "held-out" set by taking an additional independent sample from k. We use the first samples as the sample history for f which determines the future selections, and use the "held-out" set composed of the second samples for mean estimation. Since the "held-out" set is composed of i.i.d. samples that are independent of the selection process, its sample average is an unbiased estimate of μ_k. However, if the total number of samples collected is fixed at T rounds, then data splitting suffers from high variance, since half of all the samples are discarded in estimation.

Conditional Maximum Likelihood Estimator (cMLE) Data splitting is a general approach since it is agnostic to the selection function f. If we know the f used to collect the data, then more powerful debiasing could be achieved by explicitly condition on f and the observed data in a maximum likelihood framework. Consistency results have been proved in [18, 12]. To illustrate this approach, we consider the special case where the decision on which distribution to sample at round t is based on comparing the decision statistics of the form,

$$\mathbf{U}_t \triangleq (U(\overline{X_t^{(1)}}, N_t^{(1)}), \dots, U(\overline{X_t^{(K)}}, N_t^{(K)})). \tag{1}$$

$\mathbf{U}_t$ depends only on the empirical average $\overline{X_t^{(k)}}$'s and the number of samples $N_t^{(k)}$'s for $k \in [K]$. In other words, the selection function f depends on the history of rewards Λ_t only through $\mathbf{U}_t$. In Greedy, $U(\overline{X_t^{(k)}}, N_t^{(k)}) = \overline{X_t^{(k)}}$, while in UCB type algorithms, $U_t^{(k)}$ will be the upper confidence bounds that depend on both $\overline{X_t^{(k)}}$'s and $N_t^{(k)}$'s, where $U_t^{(k)}$ is shorthand for $U(\overline{X_t^{(k)}}, N_t^{(k)})$.

Theorem 2. *Suppose the distributional function for distribution k has density $h_{\theta^{(k)}}$, then the conditional likelihood of the adaptive data collection problem is proportional to*

$$p(\Lambda_T \mid s_t,\ t = 1, \dots, T) \propto \prod_{k=1}^{K} \prod_{m=1}^{N_T^{(k)}} h_{\theta^{(k)}}(X_m^{(k)}) \cdot \prod_{t=K}^{T-1} \Pr\left[f(\mathbf{U}_t) = s_{t+1} \mid \mathbf{U}_t\right]. \tag{2}$$

To maximize the conditional likelihood, we need to solve the following optimization problem,

$$\max_{\theta} \sum_{k=1}^{K} \sum_{m=1}^{N_T^{(k)}} \log\left[h_{\theta^{(k)}}(X_m^{(k)})\right] + \sum_{t=K}^{T-1} \log\left[\Pr\left[f(\mathbf{U}_t) = s_{t+1} \mid \mathbf{U}_t\right]\right] - \log Z(\theta), \tag{3}$$

where $\theta = (\theta^{(1)}, \dots, \theta^{(K)})$ are the parameters of interest and $Z(\theta)$ is the partition function in Eqn. (2)*, that only depends on the parameters θ.*

Theorem 2 gives an explicit form for the likelihood function of the adaptive data collection problem (up to a constant). We give a proof of Theorem 2 in Appendix D, and give examples of computing the conditional likelihood functions of common bandit algorithms in Appendix E

We solve the cMLE optimization problem using contrastive divergence [4]. The details of the algorithm is in the Appendix G. The computational bottleneck of the optimization is in evaluating $\Pr\left[f(\mathbf{U}_t) = s_{t+1} \mid \mathbf{U}_t\right]$, because it can induce singularities along the hard boundaries in the sample space. Details see Appendix F. To overcome this difficulty, we introduce additional randomization when selecting a distribution.

Table 1: **Bias reduction.** With $K = 5$, each distribution is drawn from $\mathcal{N}(\mu_i, 1)$. where $\mu_1 = 1.0, \mu_2 = 0.75, \mu_3 = 0.5, \mu_4 = 0.38, \mu_5 = 0.25$. In the left columns under each algorithm, we record the bias of the original algorithm at different time steps T. In the right columns, we record the percentage of the original bias that still remains after we run cMLE by adding gumbel noise $\epsilon_g \sim G_\tau$, with scale parameter $\tau = 1.0$, and contrastive divergence with 600 gradient descent iterations. All results are averaged across 1000 independent trials.

	lil' UCB		ϵ-Greedy	
	orig.	cMLE	orig.	cMLE
T=20	-0.32	14.9%	-0.31	9.1%
T=40	-0.35	14.2%	-0.27	8.8%

Adding additional noise to the sample values to improve cMLE optimization We propose adding Gumbel noise to the decision statistics $\mathbf{U}_t$ to smooth out $\Pr[f(\mathbf{U}_t) = s_{t+1} \mid \mathbf{U}_t]$ (Details see Appendix G). For lil' UCB or Greedy, we can compute $\mathbf{U}_t$ deterministically from $\overline{\mathbf{X}_t}$ and $\mathbf{N}_t$. The selection function after Gumbel randomization is defined as

$$f(\mathbf{U}_t) = \arg\max_k U_t^{(k)} + \epsilon_t^{(k)}, \quad \epsilon_t^{(k)} \stackrel{\text{iid}}{\sim} G_{\tau_t},$$

where G_τ is a Gumbel distribution of mean 0 and scale parameter τ.

We summarize the debiasing procedure in Algorithm 1.

Algorithm 1 Algorithm for debiasing adaptive data collection

Add Gumbel noise when choosing which distribution to sample from. Instead of applying the selection function directly to $\mathbf{U}_t$, we apply it to

$$(U_t^{(k)} + \epsilon_t^{(k)}), \quad k = 1, \dots, K$$

where $\epsilon_t^{(k)} \stackrel{\text{iid}}{\sim} G_{\tau_t}$.
Compute conditional likelihood by computing the selection probabilities,

$$\Pr_{\epsilon_t}[f(\mathbf{U}_t) = s_{t+1} \mid \mathbf{U}_t].$$

Note that here f also incorporates the randomness of Gumbel randomizations $\{\epsilon_t^{(k)}\}_{k \in [K]}$ as well as the randomness in the original bandit algorithm.
Compute cMLE using approximate gradient descent with contrastive divergence.

Table 2: **Mean Squared Error(MSE) reduction** Same experiments as in Table 1. The leftmost columns under each algorithm is the MSE of the original algorithm. The second to the left columns are the MSE percentage ratio of the data splitting with a held-out set compared to the MSE of the original algorithm. The right columns are the MSE percentage ratio of the cMLE algorithm after debiasing compared to the MSE of the original algorithm. For ϵ-Greedy, we additionally run propensity matching (prop). Note that both data splitting and prop suffer from high variance despite achieving consistent estimation.

	lil' UCB			$\epsilon-$Greedy			
	orig.	held	cMLE	orig.	held	prop	cMLE
T=20	0.57	112%	**99%**	0.52	123%	401%	**94%**
T=40	0.54	104%	**52%**	0.39	135%	312%	**62%**

Debiasing experiments We empirically show that the cMLE algorithm can reduce bias significantly and reduce the mean squared error (MSE) as well. In Table 1, we see significant bias reduction for the lil' UCB and ϵ-Greedy using the cMLE debiasing algorithm, in the $K = 5$ cases, where K is the number of distributions. More extensive experiments for lil' UCB and ϵ-Greedy, along with Greedy and Thompson Sampling are included in Appendix H and Appendix I. Table 2 show the reduction of MSE. The data splitting algorithm achieves consistent estimates, but it incurs high variance since the effective sample size is halved by maintaining a held-out set. Empirically we observe that data splitting suffers from high MSE. All experiments use gradient descent learning rate $\eta = 0.01$, 30 steps of MCMC (with the first half of the steps as burn-in), 600 gradient descent iterations, and have adjusted the stepsize of MCMC to ensure the acceptance ratio is between $20\% - 50\%$. The convergence of the mean estimates with gradient descent is shown in Figure 2(f) in Appendix C. We see that cMLE significantly reduces the bias, while improving the MSE. We also experimented with propensity matching, a commonly used method that weights each observed value of a distribution by one over the probability that this distribution is selected [3]. Propensity matching is unbiased, but has very large variance and thus a much greater MSE by several fold compared to cMLE. We discuss it in more detail in Appendix H.

4 Discussion

Our main result shows that adaptively collected data is negatively biased when the data collection algorithm f satisfies *Exploit* and *IIO*. This seems counterintuitive at first because we typically associate optimization (as in exploitative algorithms) with a positive selection bias ala Winner's Curse. For example, if we draw 10 samples from $\mathcal{N}(0, 1)$ and report the $\max$, then we have positive reporting bias. The reason between these phenomena is that for any sample history of data, the "best" option k's sample mean is likely to be larger than its true mean. However who is the "best" varies in different sample path, and the bias of every k is negative in expectation.

We explored data splitting and cMLE as two approaches to reduce this bias. Data splitting is unbiased but suffers larger MSE because it ignores half of the samples during estimation. cMLE can reduce bias close to 0 while also reducing MSE. The trade-off is that it requires specific knowledge about f and also requires one to add additional noise to the collected data. Both approaches requires modifying the data collection procedure and cannot be generically applied to debias existing adaptively collected data. Considering that adaptively collected data is ubiquitous, developing flexible debiasing approaches to debias existing data is an important direction of future research.

References

[1] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem.

[2] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

[3] Peter C Austin. An introduction to propensity score methods for reducing the effects of confounding in observational studies. *Multivariate behavioral research*, 46(3):399–424, 2011.

[4] Miguel A Carreira-Perpinan and Geoffrey E Hinton. On contrastive divergence learning. In *AISTATS*, volume 10, pages 33–40. Citeseer, 2005.

[5] DR Cox. A note on data-splitting for the evaluation of significance levels. *Biometrika*, 62(2):441–444, 1975.

[6] Cynthia Dwork, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Aaron Leon Roth. Preserving statistical validity in adaptive data analysis. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, pages 117–126. ACM, 2015.

[7] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research*, 7(Jun):1079–1105, 2006.

[8] Emil Julius Gumbel and Julius Lieblein. Statistical theory of extreme values and some practical applications: a series of lectures. 1954.

[9] Xiaoying Tian Harris, Snigdha Panigrahi, Jelena Markovic, Nan Bi, and Jonathan Taylor. Selective sampling after solving a convex problem. *arXiv preprint arXiv:1609.05609*, 2016.

[10] Iuliana Ionita-Laza, Angela J Rogers, Christoph Lange, Benjamin A Raby, and Charles Lee. Genetic association analysis of copy-number variation (cnv) in human disease pathogenesis. *Genomics*, 93(1):22–26, 2009.

[11] Kevin Jamieson, Matthew Malloy, Robert Nowak, and Sébastien Bubeck. lil'ucb: An optimal exploration algorithm for multi-armed bandits. In *Conference on Learning Theory*, pages 423–439, 2014.

[12] Snigdha Panigrahi, Jonathan Taylor, and Asaf Weinstein. Bayesian post-selection inference in the linear model. *arXiv preprint arXiv:1605.08824*, 2016.

[13] Daniel Russo and James Zou. Controlling bias in adaptive data analysis using information theory. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, AISTATS*, 2016.

[14] Robert Sladek, Ghislain Rocheleau, Johan Rung, Christian Dina, Lishuang Shen, David Serre, Philippe Boutin, Daniel Vincent, Alexandre Belisle, Samy Hadjadj, et al. A genome-wide association study identifies novel risk loci for type 2 diabetes. *Nature*, 445(7130):881–885, 2007.

[15] Jonathan Taylor and Robert J Tibshirani. Statistical learning and selective inference. *Proceedings of the National Academy of Sciences*, 112(25):7629–7634, 2015.

[16] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[17] Xiaoying Tian, Nan Bi, and Jonathan Taylor. Magic: a general, powerful and tractable method for selective inference. *arXiv preprint arXiv:1607.02630*, 2016.

[18] Xiaoying Tian and Jonathan E Taylor. Selective inference with a randomized response. *To Appear in the Annals of Statistics*, 2015.

[19] Larry Wasserman and Kathryn Roeder. High dimensional variable selection. *Annals of statistics*, 37(5A):2178, 2009.

[20] Min Xu, Tao Qin, and Tie-Yan Liu. Estimation bias in multi-armed bandit algorithms for search advertising. In *Advances in Neural Information Processing Systems*, pages 2400–2408, 2013.

A lil' UCB Algorithm

lil' UCB Algorithm is proposed by [11], and achieves optimal regret. It has become one of the most popular upper confidence bound type algorithms.

In lil' UCB, the selection function

$$f(\Lambda_t) = \arg\max_k \overline{X_t^{(k)}} + (1+\beta)(1+\sqrt{\epsilon})\sqrt{\frac{2(1+\epsilon)\log(\frac{\log((1+\epsilon)n)}{\delta})}{N_t^{(k)}}} \tag{4}$$

where ϵ, δ, β are lil' UCB hyperparameters as specified in [11].

B Examples for the selection function f

The simplest example of adaptive data collection is the Greedy algorithm. In Greedy, at round t, the selection function chooses to sample the distribution from which we have observed the highest empirical mean. Then $f(\Lambda_t) = \arg\max_{k\in[K]} \overline{X_t^{(k)}}$. Often in practice, a randomized version of Greedy, called ϵ-Greedy, is also used. In ϵ-Greedy with probability ϵ we uniformly randomly select a distribution and with probability $1-\epsilon$, we perform Greedy. This corresponds to the selection

$$f(\Lambda_t, \omega) = \begin{cases} \arg\max_{k\in[K]} \overline{X_t^{(k)}}, & \text{if } \omega > \epsilon \\ k, k \in [K] & \text{if } \frac{\epsilon}{K}\cdot(k-1) < \omega < \frac{\epsilon}{K}\cdot k \end{cases}$$

where $\omega \sim$ Unif$[0,1]$. All the algorithms used for multi-arm bandits can be modeled as a selection function f.

C Quantitative characteristics of bias

Analytic example with explicit bias. Consider the setting where $K=2$, $X^{(1)} \sim Bern(\mu_1)$ and $X^{(2)} \sim Bern(\mu_2)$, with $\mu_1 \geq \mu_2$. A greedy data collection procedure is to draw one sample from each distribution in the first two rounds, and at $T=3$ sample from the distribution with the larger sample. In the event of a tie, i.e. both samples are 0 or 1, then distribution 1 is selected for $T=3$ by default. We can explicitly compute the bias of each arm at $T=3$.

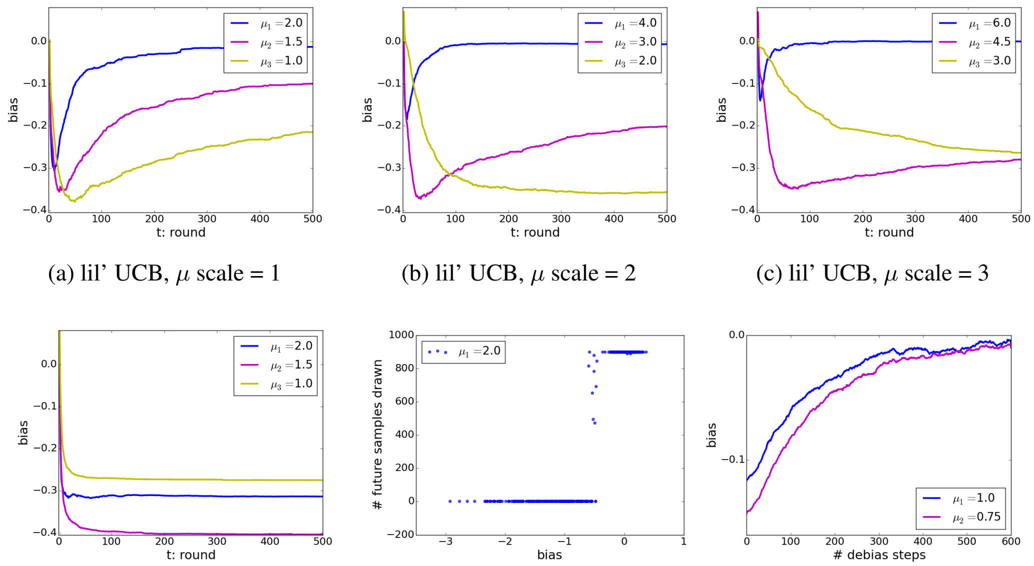
$$\text{bias}_1 \equiv \mathbb{E}\left[\overline{X_3^{(1)}}\right] - \mu_1 \quad = \quad -\frac{1}{2}\mu_1(1-\mu_1)\mu_2 \tag{5}$$

$$\text{bias}_2 \equiv \mathbb{E}\left[\overline{X_3^{(2)}}\right] - \mu_2 \quad = \quad -\frac{1}{2}\mu_2(1-\mu_2)(1-\mu_1). \tag{6}$$

When $0 < \mu_1, \mu_2 < 1$, both biases are strictly negative.

Note that the distribution with the highest mean does not always have the least bias. Using Eqn. 5, the ratio of the biases is $\frac{\text{bias}_1}{\text{bias}_2} = \frac{\mu_1}{1-\mu_2}$. Therefore bias_2 is worse than bias_1 when μ_1, μ_2 are both close to 1, and bias_1 is worse than bias_2 when μ_1, μ_2 are both close to 0. This point is further illustrated empirically in Figure 2(d) in the Gaussian case.

The insight from our proof of Theorem 1 is that the bias of distribution k at time t should be large if how likely we are to choose k in the future (after t) is sensitive to the value $\overline{X_t^{(k)}}$. This sensitivity increases if there is *consequential competition* for distribution k at time t, i.e. if there are other distribution(s), i, whose empirical average $\overline{X_t^{(i)}}$ is in some middle range from the empirical average of distribution k. When they are too far apart, the particular sample values drawn from k are not consequential to the chance of it getting sampled again. If they are too close, having one bad sample value also does not affect the chance of k being drawn as much. It is only when the distance between the distribution means are in some middle range, does it incur the most negative bias. We demonstrate the above remarks empirically in the next section.



(a) lil' UCB, μ scale = 1

(b) lil' UCB, μ scale = 2

(c) lil' UCB, μ scale = 3

(d) Greedy, μ scale = 1

(e) # future samples drawn given bias at t=100, with horizon T=1000

(f) cMLE debiasing

Figure 2: In (a-c), we plot the bias of the empirical mean estimates of three unknown distributions running lil' UCB with horizon T=500. Each is distributed according to $\mathcal{N}(\mu_i, 1)$, where μ_i is the mean of the i-th distribution, specified in the legends of the plot. We see that as we scale up μ_i's, so they become more spread out, the bias increases/decreases depending how far the μ_i's are from each other, and what is the order of the distributions. (d) plots the bias of the three unknown distributions running Greedy. (e) plots the number of future samples drawn from distribution 1 given its bias at $t = 100$, running lil' UCB. Here T=1000 with two distributions, $\mathcal{N}(2, 1)$ and $\mathcal{N}(1.5, 1)$. This is a scatter plot over 1000 independent trials. (f) plots the bias as the estimate of the mean converges to the true mean across 600 gradient descent iterations

Experiments quantifying negative bias. We explore the effects on the bias from moving the distribution means apart. We used the lil' UCB algorithm, with algorithm specific parameters $\alpha = 9, \beta = 1, \epsilon = 0.01, \delta = 0.005$, which are the same as in the experiment section of [11]. We ran 1000 independent trials, with horizon $T = 500$. We have three unknown distributions, all of the form $\mathcal{N}(\mu_i, 1)$, with $\mu_1 = 2, \mu_2 = 1.5, \mu_3 = 1$. In this experiment, we scale the μ's by a scaling factor of $1, 2, 3$, and observe the bias of the empirical mean estimates of the three distributions. In Figure 2(a) (b) (c), we plot the bias with the number of rounds.

We first observe all distributions have negatively biased estimates of their true means. Further, the distribution with the second best mean has worse bias as we scale up the μ's. We hypothesize the exact sample values we receive from this distribution matter a lot more when it is farther from the distribution with the highest mean. When they are close together, having one bad sample value does not affect its chance of being sampled again as much as when their means are further apart. On the other hand, for the distribution with the lowest true mean, we observe its bias becomes worse first and then better as we scale up the μ's. The reason why it goes down first is the same as why the second best distribution has worse bias as μ scales up - that is, they are both in the *consequential competition* regime. However, as we further scale up the μ's, the bad sample values from the distribution with the lowest mean does not affect its future chances of being drawn much more than the good samples values, since its true mean is far from the distribution with the highest mean.

Next we compare lil' UCB with Greedy, see subfigure (a) and (d) in Figure 2. First, we observe that with Greedy in our setting, the empirical mean estimates for distribution with the lowest mean has the least bias, followed by the distribution with the highest true mean. This is an example in which the distribution with the highest mean might not incur the least bias. With lil' UCB, the bias for the distribution with the highest true mean converges to 0 quickly, but with Greedy it plateaus. In lil' UCB, since it achieves optimal regret, the algorithm finds the distribution the highest true mean in finite number of time steps. The samples we get from that distribution become close to i.i.d. samples as t increases, since the effect of the competition from other arms is reduced over time. In Greedy it's known that the algorithm can be stuck on drawing from a suboptimal distribution, in which case the empirical average of the particular samples we have drawn from the distribution with the highest true mean must have a negative bias for this to happen. The bias of the best distribution thus doesn't converge to 0.

Figure 2(e) shows at round step $t = 100$ with horizon $T = 1000$, running lil' UCB with the same hyperparameters in the same setting as in Figure 2(a), we plot the number of future samples drawn from the distribution with the highest mean (i.e. $\mu = 2.0$) vs. the bias from the empirical average of samples drawn so far from this distribution at time $t = 100$. This confirms our intuition that large negative bias is correlated with fewer future chances of getting sampled.

D Proofs of the main results

Proof of Theorem 1. Without loss of generality, we focus on showing that distribution 1 has negative bias. The argument applies directly to every other distribution. For a given history Λ_t, $f(\Lambda_t)$ is a random variable over $[K]$. We define two independent random variables based on $f(\Lambda_t)$. Let $g(\Lambda_t)$ be a binary random variable such that $\Pr[g(\Lambda_t) = 1] = \Pr[f(\Lambda_t) = 1]$ and $\Pr[g(\Lambda_t) = 0] = \Pr[f(\Lambda_t) \neq 1]$. Let $h\left(\Lambda_t^{(-1)}\right)$ be a random variable with support $\{2, ..., K\}$, such that for $k \in \{2, ..., K\}$,

$$\Pr\left[h\left(\Lambda_t^{(-1)}\right) = k\right] = \Pr[f(\Lambda_t) = k | f(\Lambda_t) \neq 1] = \frac{\Pr[f(\Lambda_t) = k]}{\sum_{i=2}^{K} \Pr[f(\Lambda_t) = i]}.$$

Note that f satisfies ***IIO*** implies that the law of h is only a function of $\Lambda_t^{(-1)}$, which is the history only of the distributions $2, ..., K$ up to time t. It's clear that distribution selection by $s_{t+1} = f(\Lambda_t)$ is equivalent to (i.e. have the same law as)

$$s_{t+1} = \begin{cases} 1, & \text{if } g(\Lambda_t) = 1. \\ k, & \text{if } g(\Lambda_t) = 0,\, h(\Lambda_t^{(-1)}) = k,\, k \in [2, K]. \end{cases} \tag{7}$$

Since this equivalence holds for every t, the adaptive data collection procedure is defined by the independent random variables $g(\Lambda_t)$ and $h(\Lambda_t^{(-1)})$.

To study distribution 1 we condition on the realization Θ, where Θ includes the realizations of distributions k for $k \in \{2, ..., K\}$ and T random seeds for g and h, $\{\omega_{g,t}, \omega_{h,t}\}_{t=1}^{T}$. More precisely, $\Theta = \{\{x_t^{(k)}\}_{t=1}^{T}, \{\omega_{g,t}, \omega_{h,t}\}_{t=1}^{T}, k \in [K]\}$, where $x_t^{(k)}$ is a realized value of a sample drawn from distribution k at round t. Then given any realization of distribution 1, $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_T)$, $\sigma_i \in \mathbb{R}$, conditioning on Θ induces a deterministic mapping $S(\sigma) = (t_1, ..., t_T)$, where t_i is a positive integer corresponding to the time when the i-th sampling of distribution 1 occurs. Note that $t_i \in [T] \cup *$, where $t_i = *$ indicates that the i-th pull occurs after time T. Since all the other distribution's realization and randomness are fixed, t_i is a deterministic function of $(\sigma_1, ..., \sigma_{i-1})$. Let $\tilde{t}_j$ indicate the round at which distribution 1 is *not* selected the j-th time, then IIO implies $s_{\tilde{t}_j} = h(\Lambda_{\tilde{t}_j - 1}^{(-1)}, \omega_{h,j})$. Which distribution among $2, \ldots, K$ is selected is determined by $\Lambda_{\tilde{t}_j - 1}^{(-1)}$, which is the history of distributions $2, \ldots, K$ up to time $\tilde{t}_j - 1$. Note that $s_{\tilde{t}_j}$ is a function of $\omega_{h,j}$ not $\omega_{h,\tilde{t}_j}$; i.e. the random seeds $\omega_{h,j}$ is only used when distribution 1 is not selected. From this observation, we see an important property of conditioning on Θ.

Property 1. If $\tilde{t}_j$ indicate the round at which distribution 1 is *not* selected for the j-th time, then the history $\Lambda_{\tilde{t}_j}^{(-1)}$ is completely determined by the index j.

Our goal is to show that for an arbitrary realization Θ, $\mathbb{E}\left[\overline{X_T^{(1)}} | \Theta\right] \leq \mu_1$. Then it would follow that $\mathbb{E}\left[\overline{X_T^{(1)}}\right] \leq \mu_1$. As we discussed above, after conditioning on Θ, the data collection procedure is equivalent to a mapping $S((\sigma_1, ..., \sigma_T)) = (t_1, ..., t_T)$. For a given path $\sigma = (\sigma_1, ..., \sigma_T)$, let $n_\sigma = |\{t_i : t_i \leq T\}|$ be the number of times distribution 1 is selected by round T. S depends on Θ, but we'll not write this explicitly to simplify notation. Moreover, $\Pr[\sigma|\Theta] = \Pr[\sigma]$ since the values of distribution 1 is independent of the realizations of the other distributions and the randomness in the selections. Therefore,

$$\mathbb{E}\left[\overline{X_T^{(1)}}|\Theta\right] = \sum_{\sigma} \Pr[\sigma] \frac{\sum_{i=1}^{n_\sigma} \sigma_i}{n_\sigma}.$$

Our proof strategy is to show that any mapping S from paths σ to sets of times $(t_1, ..., t_T)$ which satisfies *Exploit* condition must have bias ≤ 0. It suffices to consider the mapping S corresponding to the largest $\mathbb{E}\left[\overline{X_T^{(1)}}|\Theta\right]$ and still satisfies *Exploit*. We show that such a mapping S must have the property that n_σ is the same constant for all path σ. For such an S, it is immediate that $\mathbb{E}\left[\overline{X_T^{(1)}}|\Theta\right] = \mu_1$.

Suppose for a maximal mapping S, n_σ differs for different σ. Let l be the largest integer for which there exists two paths σ and σ' such that $\sigma_i = \sigma'_i$ for $i < l$ and $n_\sigma \neq n_{\sigma'}$. So σ and σ' agree up to the $l-1$st drawing of distribution 1. We denote $\alpha \equiv \sigma_l$ and $\alpha' \equiv \sigma'_l$; without loss of generality we can assume $\alpha < \alpha'$.

Property 2. The fact that l is the largest such index implies that if σ'' is any other path such that $\sigma''_i = \sigma_i$ for $i \leq l$ then $n_{\sigma''} = n_\sigma$. Similarly if $\sigma''_i = \sigma'_i$ for $i \leq l$ then $n_{\sigma''} = n_{\sigma'}$.

There are two possible cases and we show that they both lead to contradictions. This would complete the proof by contradiction.

Case 1: $n_\sigma > n_{\sigma'}$. Consider the two path $\lambda = (\sigma_1, ..., \sigma_{l-1}, \alpha, \lambda_{l+1}, ..., \sigma_T)$ and $\lambda' = (\sigma_1, ..., \sigma_{l-1}, \alpha', \lambda_{l+1}, ..., \lambda_T)$, where $\lambda_{l+1}...\lambda_T$ is some arbitrary fixed string of realizations. Property 2 implies that $n_\lambda = n_\sigma > n_{\sigma'} = n_{\lambda'}$. Under the mapping S, λ and λ' maps onto two sets of times $\{t_{\lambda,i}\}_{i=1}^T$ and $\{t_{\lambda',i}\}_{i=1}^T$, where $t_{\lambda,i}$ (resp. $t_{\lambda',i}$) is the round at which distribution 1 is drawn the i-th time under the realization λ (resp. λ'). Since at least the first $l-1$ terms of λ and λ' are equal, at least the first l terms of $t_{\lambda,i}$ and $t_{\lambda',i}$ are equal. Let $l_1 > l$ be the first index where $t_{\lambda,l_1} < t_{\lambda',l_1}$. There must exist such a l_1 in order for $n_\lambda > n_{\lambda'}$.

Consider the round $t^* = t_{\lambda,l_1} - 1$. The histories up to round t^* of paths λ and λ', i.e. $\Lambda_{\lambda,t^*}^{(-1)}$ and $\Lambda_{\lambda',t^*}^{(-1)}$, are identical because in both paths distribution 1 has been selected $l_1 - 1$ times by round t^* (by Property 1). Moreover the empirical average of distribution 1 under λ is strictly lower than the average under λ'. *Exploit* property states that $g(\Lambda_{\lambda,t^*}, \omega_{g,t^*}) = 1 = f(\Lambda_{\lambda,t^*}, \omega_{g,t^*})$ implies $f(\Lambda_{\lambda',t^*}, \omega_{g,t^*}) = 1 = g(\Lambda_{\lambda',t^*}, \omega_{g,t^*})$. This implies that $t_{\lambda,l_1} = t_{\lambda',l_1}$, contradicting $t_{\lambda,l_1} < t_{\lambda',l_1}$. Therefore the scenario $n_\sigma > n_{\sigma'}$ is not possible if f satisfies *Exploit*. Note that for any Λ_t, we can use the same probability space Ω for $g(\Lambda_t)$ and $f(\Lambda_t)$ such that $\{\omega : g(\Lambda_t, \omega) = 1\} = \{\omega : f(\Lambda_t, \omega) = 1\}$.

Case 2: $n_\sigma < n_{\sigma'}$. By Property 2, all the path where the first l terms are $\sigma_1...\sigma_{l-1}\alpha$ have n_σ total number of draws. The contribution of these paths to the average $\overline{X_T^{(1)}}$ is

$$\mathbb{E}\left[\overline{X_T^{(1)}}|\Theta, \sigma_1, ..., \sigma_{l-1}, \alpha\right] = \frac{\sum_{i=1}^{l-1} \sigma_i + \alpha + (n_\sigma - l)\mu_1}{n_\sigma}.$$

Similarly, all the path where the first l terms are $\sigma_1...\sigma_{l-1}\alpha'$ have $n_{\sigma'}$ total number of draws. The contribution of these paths to the average $\overline{X_T^{(1)}}$ is

$$\mathbb{E}\left[\overline{X_T^{(1)}}|\Theta, \sigma_1, ..., \sigma_{l-1}, \alpha'\right] = \frac{\sum_{i=1}^{l-1} \sigma_i + \alpha' + (n_{\sigma'} - l)\mu_1}{n_{\sigma'}}.$$

Since $\frac{\sum_{i=1}^{l-1} \sigma_i + \alpha}{l} < \frac{\sum_{i=1}^{l-1} \sigma_i + \alpha'}{l}$, we must have either of the following hold:

1. $\frac{\sum_{i=1}^{l-1} \sigma_i + \alpha}{l} < \mu_1$. If this holds true, then the paths where the first l terms are $\sigma_1...\sigma_{l-1}\alpha$ can have m instead of n_σ total number of draws, where $n_\sigma < m \leq n_{\sigma'}$. Note that

$\frac{\sum_{i=1}^{l-1}\sigma_i+\alpha+(n_\sigma-l)\mu_1}{n_\sigma} < \frac{\sum_{i=1}^{l-1}\sigma_i+\alpha+(m-l)\mu_1}{m}$. This modification preserves *Exploit* property while increasing $\mathbb{E}\left[\overline{X_T^{(1)}}|\Theta,\sigma_1,...,\sigma_{l-1},\alpha\right]$, and thus increasing the $\mathbb{E}\left[\overline{X_T^{(1)}}|\Theta\right]$ of S. This contradicts the assumption that S is the maximal mapping.

2. $\frac{\sum_{i=1}^{l-1}\sigma_i+\alpha'}{l} > \mu_1$. If this holds true, then the paths where the first l terms are $\sigma_1...\sigma_{l-1}\alpha'$ can have m' instead of $n_{\sigma'}$ total number of draws, where $n_\sigma \leq m < n_{\sigma'}$. Note that $\frac{\sum_{i=1}^{l-1}\sigma_i+\alpha'+(n_{\sigma'}-l)\mu_1}{n_{\sigma'}} < \frac{\sum_{i=1}^{l-1}\sigma_i+\alpha'+(m-l)\mu_1}{m}$. This modification preserves *Exploit* property while increasing $\mathbb{E}\left[\overline{X_T^{(1)}}|\Theta,\sigma_1,...,\sigma_{l-1},\alpha'\right]$, and thus increasing the $\mathbb{E}\left[\overline{X_T^{(1)}}|\Theta\right]$ of S. This contradicts the assumption that S is the maximal mapping.

The case analysis proves that in order for S to be the mapping corresponding to the maximal $\left[\overline{X_T^{(1)}}|\Theta\right]$ it must assign the same constant n_σ for all path σ, i.e. the number of times distribution 1 is selected does not depend on its observed values. Such a mapping is unbiased: $\left[\overline{X_T^{(1)}}|\Theta\right] = \mu_1$. □

Proof of Proposition. 1. For any algorithm with the following form of the selection function,

$$f\left(\Lambda_t^{(k)} \cup \Lambda_t^{(-k)}\right) = \arg\max_{k\in[K]} U_t^{(k)}\left(\overline{X_t^{(k)}}, N_t^{(k)}, \omega\right), \tag{8}$$

such that conditioning on $\Lambda_t^{(k)}$ and $\Lambda_t^{'(k)}$ with $N_t^{(k)} = N_t^{'(k)}$, and $\overline{X_t^{(k)}} < \overline{X_t^{'(k)}}$, and fixing $\Lambda_t^{(-k)}$ and ω, we have $U_t^{(k)}(\overline{X_t^{(k)}}, N_t^{(k)}, \omega) < U_t^{'(k)}(\overline{X_t^{'(k)}}, N_t^{'(k)}, \omega)$, then it satisfies Exploit by definition. We show lil' UCB, Greedy, and ϵ-Greedy can all be written in the form of Eqn. 8.

In lil' UCB,

$$U_t^{(k)}\left(\overline{X_t^{(k)}}, N_t^{(k)}, \omega\right) = U_t^{(k)}\left(\overline{X_t^{(k)}}, N_t^{(k)}\right) = \overline{X_t^{(k)}} + (1+\beta)(1+\sqrt{\epsilon})\sqrt{\frac{2(1+\epsilon)\log(\frac{\log((1+\epsilon)n)}{\delta})}{N_t^{(k)}}} \tag{9}$$

where ϵ, δ, β are lil' UCB hyperparameters as specified in [11]. In Greedy,

$$U_t^{(k)}(\overline{X_t^{(k)}}, N_t^{(k)}, \omega) = U_t^{(k)}(\overline{X_t^{(k)}}) = \overline{X_t^{(k)}} \tag{10}$$

In ϵ-Greedy,

$$U_t^{(k)}(\overline{X_t^{(k)}}, N_t^{(k)}, \omega) = \begin{cases} \overline{X_t^{(k)}}, & \text{if } \omega > \epsilon \\ - & \text{if } \omega < \epsilon \end{cases} \tag{11}$$

In Eqn. 11, when $\omega < \epsilon$, since we condition on ω, it is trivially true that $f(\Lambda_t^{(k)} \cup \Lambda_t^{(-k)}) = k$ implies $f(\overline{\Lambda_t^{(k)}} \cup \Lambda_t^{(-k)}) = k$. In all of the above algorithms, $U_t^{(k)}$ monotonically increases as $\overline{X_t^{(k)}}$ increases, conditioning on ω and $N_t(k)$ fixed. Thus all three algorithms satisfy Exploit.

lil' UCB and greedy trivially satisfy IIO because they are deterministic algorithms. For ϵ-Greedy, conditioning on $f(\Lambda_t) \neq k$ and $f(\Lambda_t) \neq k$, and $\Lambda_t^{(-k)}$, if $\omega < \epsilon$, then $f(\Lambda_t, \omega)$ is determined by $\Lambda_t^{(-k)}$. If $\omega > \epsilon$, then all the $K-1$ arms are uniformly chosen in both cases. □

Proof of Proposition. 2. Without loss of generality, we focus on showing that distribution 1 has negative bias. We modify the arguments used to prove Theorem 1. To study distribution 1 we condition on the realization Θ, where Θ includes the realization of distribution 2 and T random seeds for f, $\{\omega_t\}_{t=1}^T$. Then given any realization of distribution 1, $\sigma = (\sigma_1, \sigma_2, ..., \sigma_T)$, $\sigma_i \in \mathbb{R}$, conditioning on Θ induces a deterministic mapping $S(\sigma) = \{t_1, ..., t_T\}$, where t_i is a positive integer corresponding to the time when the $i-th$ pull of distribution 1 occurs. Note that $t_i \in [T] \cup *$, where $t_i = *$ indicates that the i-th pull occurs after time T. Since the realizations of distribution 2 and the randomness in f are fixed, t_i is a deterministic function of $\{\sigma_1, ..., \sigma_{i-1}\}$. We also have the following property as a consequence.

Property 1. If $\tilde{t}_j$ indicate the j-th time where distribution 2 is selected, then the history $\Lambda^{(2)}_{\tilde{t}_j}$ is completely determined by the index j.

The rest of the proof is identical to the proof of Theorem 1. □

Proof of Theorem 2. The conditional likelihood is related to the original likelihood via *selective likelihood ratio (LR)* .

$$LR(\mathbf{U} \mid s_t, t = 1, \ldots, T) \propto \prod_{t=K}^{T-1} \Pr\left[f(\mathbf{U}_t) = s_{t+1} \mid \mathbf{U}_t\right], \tag{12}$$

where $\mathbf{U} = (\mathbf{U}_t)_{t=1}^T$. The index starts from K because we always draw samples from each distribution once in the beginning. The probability is taken over the extra randomness in the selection function f, fixing the decision statistics $\mathbf{U}_t$'s and the sequence of choices s_t's. Moreover, note that conditioning on the sequence of distribution to select s_t's means we are also fixing $\mathbf{N}_t$'s as they are equivalent.

Using the change of variable formula and the selective likelihood ratio in Eqn. 12, we have

$$\begin{aligned}
&p_{\Lambda_T}(\Lambda_T \mid s_t,\ t = 1, \ldots, T)\\
=&p_{\mathbf{U}}(\mathbf{U} \mid s_t,\ t = 1, \ldots, T) \times |\det \mathbf{J}_{\mathbf{\Lambda_T} \to \mathbf{U}}|\\
=&h_{\mathbf{U}}(\mathbf{U}) LR(\mathbf{U} \mid s_t,\ t = 1, \ldots, T) \times |\det \mathbf{J}_{\mathbf{\Lambda_T} \to \mathbf{U}}|\\
=&h_{\Lambda_T}(\Lambda_T) \times |\det \mathbf{J}_{\mathbf{U} \to \mathbf{\Lambda_T}}| \times LR(\mathbf{U} \mid s_t,\ t = 1, \ldots, T) \times |\det \mathbf{J}_{\mathbf{\Lambda_T} \to \mathbf{U}}|\\
=&h_{\Lambda_T}(\Lambda_T) \times \prod_{t=K}^{T-1} \Pr\left[f(\mathbf{U}_t) = s_{t+1} \mid \mathbf{U}_t\right],
\end{aligned}$$

where $\mathbf{J}_{\mathbf{\Lambda_T} \to \mathbf{U}}$ is the Jacobian matrix for the map from $\Lambda_T \to \mathbf{U}$. $h_{\Lambda_T}(\Lambda_T)$ is the unconditional likelihood of the data generating distribution. Note the last equation is due to that there is an invertible (linear) map between Λ_T and $\mathbf{U}$.

Finally, we note that the unconditional distribution of Λ_T is

$$h_{\Lambda_T}(\Lambda_T) = \prod_{k=1}^{K} \prod_{m=1}^{N_T^{(k)}} h_{\theta^{(k)}}(X_m^{(k)})$$

and the selective likelihood ratio is proportional to the right-hand-side of Eqn.12. □

E Examples of computing the conditional likelihood

Here are some examples of computing the explicit forms of the conditional likelihood. We see from Eqn. 2 that it suffices to compute the selective likelihood ratios through Eqn. 12 for the different algorithms. The explicit form of the conditional likelihood for Thompson Sampling can be found in Appendix I.

1. **Additive Gumbel randomizations** for Greedy or lil' UCB algorithms: per Lemma 1,

$$\Pr\left[f(\mathbf{U}_t) = k \mid \mathbf{U}_t\right] = \frac{\exp\left[U_t^{(k)}/\tau_t\right]}{\sum_{i=1}^K \exp\left[U_t^{(i)}/\tau_t\right]},$$

2. ϵ**-Greedy**:

$$\Pr\left[f(\overline{\mathbf{X}_t}) = k\right] = \frac{\epsilon}{K} + (1-\epsilon)\mathbb{I}\left(\arg\max_i \overline{X_t^{(i)}} = k\right).$$

ϵ**-Greedy + Gumbel**: the selection function will be

$$f(\overline{\mathbf{X}_t}) = \begin{cases} \arg\max_k \overline{X_t^{(k)}} + \epsilon_t^{(k)}, & \text{w.p. } 1-\epsilon \\ \text{chooses } k \text{ uniformly at random} & \text{w.p. } \epsilon \end{cases}, \quad \epsilon_t^{(k)} \overset{\text{iid}}{\sim} G_{\tau_t}.$$

and the selection probabilities are

$$\Pr\left[f(\overline{\mathbf{X}_t}) = k\right] = \frac{\epsilon}{K} + (1-\epsilon) \cdot \frac{\exp[\overline{X_t^{(k)}}/\tau_t]}{\sum_{i=1}^K \exp[\overline{X_t^{(i)}}/\tau_t]}.$$

We see that with Gumbel randomization, the only difference is that we replace argmax with the softmax function.

F Details on the computational difficulty of evaluating the selection likelihood

As an example, in Greedy,

$$\Pr\left[f(\mathbf{U}_t) = s_{t+1} \mid \mathbf{U}_t\right] = \mathbb{I}\left(\arg\max_k \overline{X_t^{(k)}} = s_{t+1}\right) \tag{13}$$

which means to compute the cMLE, we need to maximize the log-likelihood in a constrained region of the sample space. However, since the comparisons are made on the sample average $\overline{\mathbf{X}_t} = \left(\overline{X_t^{(1)}}, \ldots, \overline{X_t^{(K)}}\right)$, it induces a complicated constrained region on the sample history Λ_T. Optimization on such a region is no easy task. Moreover, since the hard-max function induces singularity along the boundary of the constrained region, the cMLE will be ill-behaved, c.f. [18, 12]. To overcome this difficulty, we introduce additional randomization when selecting a distribution.

G Optimization the cMLE with contrastive divergence

As stated above, Theorem 2 gives an explicit formula for likelihood function up to a normalizing constant (partition function). Since it is infeasible to get an explicit formula for this partition function, we use Contrastive Divergence (CD) proposed in [4] for solving the Maximum Likelihood Estimation problem.

To maximize the log-likelihood,

$$\max_\theta \; \log p(\Lambda_T \mid s_t,\ t = 1, \ldots, T; \theta)$$

we compute its approximate gradient descent using CD. Suppose

$$p(\Lambda_T \mid s_t,\ t = 1, \ldots, T; \theta) = \frac{\ell(\Lambda_T \mid s_t,\ t = 1, \ldots, T; \theta)}{Z(\theta)},$$

then the approximate gradient step for θ would be

$$\theta_{i+1} = \theta_i + \eta \left(\left.\frac{\partial \ell}{\partial \theta}\right|_{\Lambda_T} - \left.\frac{\partial \ell}{\partial \theta}\right|_{\Lambda_T'} \right),$$

where Λ_T' is a single step of MCMC from the density $p(\Lambda_T \mid s_t,\ t = 1, \ldots, T; \theta_i)$, η is the step size. Contrastive Divergence can be seen as a form of stochastic gradient descent where the gradient $\frac{\partial \log Z(\theta)}{\partial \theta} = \mathbb{E}_{\Lambda_T}\left[\frac{\partial \ell}{\theta}\right]$ is approximated by a single sample from the MCMC chain. In practice, to stabilize the gradient, we may take multiple samples from the MCMC chain and average the gradient to reduce variance.

The following is the algorithm for finding the (conditional) MLE using Contrastive Divergence,

Gumbel noise is chosen so that

$$\Pr\left[f(\mathbf{U}_t) = k \mid \mathbf{U}_t\right] = \frac{\exp[U_t^{(k)}/\tau_t]}{\sum_{i=1}^K \exp[U_t^{(i)}/\tau_t]}, \tag{14}$$

due to the Gumbel-max trick [8] (also see Lemma 1 in Appendix G). Eqn. 14 is smooth and is much easier to optimize over compared to Eqn. 13. Similarly, we can also add Gumbel noise to ϵ-Greedy to derive smooth conditional probabilities.

With these smooth $\Pr\left[f(\mathbf{U}_t) = k \mid \mathbf{U}_t\right]$, we can now optimize the cMLE Eqn. 3 using contrastive divergence[4].

Algorithm 2 Algorithm for computing cMLE for adaptive data collection

Initialize $\theta_0 = \left(\overline{X_T^{(1)}}, \dots, \overline{X_T^{(K)}}\right)$ to be the empirical means.
repeat
Obtain MCMC samples $(\Lambda_T^{'(1)}, \dots, \Lambda_T^{'(R)})$ from the density in Eqn. 2 at θ_i, where R is the number of MCMC samples we take.
Update θ through the gradient step,

$$\theta_{i+1} = \theta_i + \eta \left(\frac{\partial \ell}{\partial \theta}\bigg|_{\Lambda_T} - \frac{1}{R}\sum_{r=1}^{R} \frac{\partial \ell}{\partial \theta}\bigg|_{\Lambda_T^{\prime(r)}} \right),$$

$i \mapsto i+1$
until θ_i converges

Lemma 1 (Gumbel-Max trick)**.** *For any fixed vectors $U = (U^{(1)}, \dots, U^{(K)}) \in \mathbb{R}^K$, we have*

$$\Pr_{\epsilon}\left[\arg\max_i U^{(i)} + \epsilon^{(i)} = k\right] = \frac{\exp(U^{(k)}/\tau)}{\sum_{i=1}^K \exp(U^{(k)}/\tau)},$$

where $\epsilon^{(k)} \overset{iid}{\sim} G_\tau$, where G_τ is Gumbel distribution with scale τ.

Proof. Let $t(x) = \exp(-x/\tau)$, then we have

$$\begin{aligned}
&\Pr_{\epsilon}\left[U^{(k)} + \epsilon^{(k)} > U^{(i)} + \epsilon^{(i)},\ i \neq k\right] \\
&= \Pr_{\epsilon^{(k)}}\left[\prod_{1 \le i \le K, i \neq k} e^{-t(U^{(k)} + \epsilon^{(k)} - U^{(i)})}\right] \\
&= \int_{\epsilon^{(k)} \in \mathbb{R}} \exp\left(-\sum_{1 \le k \le K, i \neq k} t(U^{(k)} + \epsilon^{(k)} - U^{(k)})\right) \frac{1}{\tau} t(\epsilon^{(k)}) e^{-t(\epsilon^{(k)})} d\epsilon^{(k)} \\
&= \int_{\epsilon^{(k)} \in \mathbb{R}} \exp\left(-\sum_{i=1}^{K} t(\epsilon^{(k)} + U^{(k)} - U^{(i)})\right) \frac{1}{\tau} t(\epsilon^{(k)}) d\epsilon^{(k)} \\
&= \int_{\epsilon^{(k)} \in \mathbb{R}} \exp\left(-t(\epsilon^{(k)}) \sum_{i=1}^{K} t(U^{(k)} - U^{(i)})\right) \frac{1}{\tau} t(\epsilon^{(k)}) d\epsilon^{(k)} \\
&= -\int_{-\infty}^{0} \exp\left(-s \sum_{i=1}^{K} t(U^{(k)} - U^{(i)})\right) ds \\
&= \frac{1}{\sum_{i=1}^K t(U^{(k)} - U^{(i)})} = \frac{e^{U^{(k)}/\tau}}{\sum_{i=1}^K e^{U^{(i)}/\tau}}.
\end{aligned}$$

□

H More extensive debiasing experiments

H.1 Propensity Matching

Propensity Matching [3] is an unbiased estimator that is commonly used in selection functions that make choices based on the probability of selecting a distribution, such as in EXP3 suggested by [2]. The estimator achieves consistent estimates by

$$\hat{X^{(k)}} = \mathbb{I}\left(f(\Lambda_t) = k\right) \cdot \frac{X^{(k)}_{N_t^{(k)}}}{\Pr[f(\Lambda_t) = k]}. \tag{15}$$

Table 3: **Bias reduction.** With $K = 2$, each distribution is drawn from $\mathcal{N}(\mu_i, 1)$. where $\mu_1 = 1.0, \mu_2 = 0.75$. With $K = 5$, each distribution is drawn from $\mathcal{N}(\mu_i, 1)$. where $\mu_1 = 1.0, \mu_2 = 0.75, \mu_3 = 0.5, \mu_4 = 0.38, \mu_5 = 0.25$. In the left columns under each algorithm, we record the bias of the original algorithm at different time steps T. In the right columns, we record the percentage of the original bias that still remains after we run cMLE by adding gumbel noise $\epsilon_g \sim G_\tau$, with scale parameter $\tau = 1.0$, and contrastive divergence with 600 gradient descent iterations. All results are averaged across 1000 independent trials.

	lil' UCB		ϵ-Greedy ($\epsilon = 0.1$)		Greedy	
	orig.	cMLE	orig.	cMLE	orig.	cMLE
T=8,K=2	-0.26	6.2%	-0.25	7.3%	-0.29	2.8%
T=16,K=2	-0.29	5.2%	-0.25	1.6%	-0.32	8.3%
T=20,K=5	-0.32	14.9%	-0.31	9.1%	-0.35	18.0%
T=40,K=5	-0.35	14.2%	-0.27	8.8%	-0.37	15.9%

Table 4: **Mean Squared Error(MSE) reduction** Same experiments as in Table 3. The leftmost columns under each algorithm is the MSE of the original algorithm. The middle columns are the MSE percentage ratio of the data splitting with a held-out set compared to the MSE of the original algorithm. Note that despite data splitting achieves consistent estimates, it has very high variance because it uses half of the sample size for estimation. The right columns are the MSE percentage ratio of the cMLE algorithm after debiasing compared to the MSE of the original algorithm.

	lil' UCB			$\epsilon-$Greedy($\epsilon = 0.1$)			Greedy		
	orig.	held-out	cMLE	orig.	held-out	cMLE	orig	held-out	cMLE
T=8,K=2	0.56	108%	**86%**	0.51	123%	**76%**	0.56	108%	**78%**
T=16,K=2	0.50	101%	**40%**	0.38	123%	**52%**	0.53	107%	**45%**
T=20,K=5	0.57	112%	**99%**	0.52	123%	**94%**	0.59	111%	**89%**
T=40,K=5	0.54	104%	**52%**	0.39	135%	**62%**	0.54	107%	**52%**

for any $t \in [T]$, and $k \in [K]$. This estimator also suffers from high variance, as observed in Table 2. Additionally, this estimator is only relevant to be applied if the selection function f outputs a probability distribution over which one of the K distributions to select at each timestep.

H.2 Additional Results

Here we include additional results with $K = 2$ and $K = 5$ arms, as well as the results of the Greedy algorithm.

I Extensions to Thompson Sampling

Thompson Sampling is another common bandit algorithm [16, 1]. We extend Proposition. 1 to Thompson sampling, and then show how to apply cMLE, and finally show empirical results.

I.1 Extension of Proposition. 1 to Thompson Sampling

Lemma 2. *For Thompson sampling, we impose the following constraints. Let $\{\theta_i^{(k)}\}$ be a set of M parameters that are updated after each pull of arm k. Let $F_{\theta_i^{(k)}}$ be the CDF of $\theta_i^{(k)}$. Assume it's strictly monotonic and continuous, and for any $q_1, \cdots, q_M \in [0, 1]$*

$$\mathbb{E}\left[X^{(k)}|F^{-1}_{\theta_1^{(k)}|\overline{X}_t^{(k)}}(q_1), \cdots, F^{-1}_{\theta_M^{(k)}|\overline{X}_t^{(k)}}(q_M)\right] > \mathbb{E}\left[X^{(k)}|F^{-1}_{\theta_1^{(k)}|\overline{X}_t^{(k)\prime}}(q_1), \cdots, F^{-1}_{\theta_M^{(k)}|\overline{X}_t^{(k)\prime}}(q_M)\right] \tag{16}$$

if $\overline{X}_t^{(k)} > \overline{X}_t^{(k)\prime}$. Then Thompson sampling is also equivalent to selection function $f(\Lambda_t, \omega = \{q_i\}_{i=1}^M)$ that satisfies Exploit *and* IIO.

Proof. Since we condition on a fixed realization of $q_1, \cdots, q_M$ drawn for each arm at each time it receives a pull, given Equation (**??**) is satisfied, *Exploit* is trivially satisfied. For *IIO*, since the posterior of $\theta_i^{(k)}$ is a deterministic function of the history Λ_i, it is also trivially satisfied. □

I.2 cMLE for Thompson Sampling

For **Thompson sampling**:

$$\Pr\left[f(\overline{\mathbf{X}}_t) = k\right] = \Pr_{\hat{\mu}_t}\left[\hat{\mu}_t^{(k)} > \hat{\mu}_t^{(j)},\ j \neq k\right],$$

where $\hat{\mu}_t^{(k)} \sim N(\mu_t^{(k)}, \sigma_t^{(k)2})$. Unfortunately, because the $\hat{\mu}$'s have different means and variances, the above probability will not have a closed form expression. Numerical evaluations can be expensive. To address this difficulty, we can instead condition on the observed expected posterior reward $\hat{\mu}$'s which determines the choice z_t. The conditional likelihood would then be proportional to

$$\prod_{k=1}^{K} \prod_{m=1}^{N_T^{(k)}} f_{\theta^{(k)}}(X_m^{(k)}) \prod_{t=K}^{T-1} \prod_{k=1}^{K} \phi\left(\frac{\hat{\mu}_t^{(k)} - \mu_t^{(k)}}{\sigma_t^{(k)}}\right),$$

where $\phi(\cdot)$ is the PDF of the standard normal distribution.

For **Thompson + Gumbel**, additional Gumbel noises are added to the expected reward $\hat{\mu}_t^{(k)}$'s. In other words, the selection function will be

$$f\big((\mu_t, \sigma_t^2)\big) = \arg\max_k \hat{\mu}_t^{(k)} + \epsilon_t^{(k)}, \quad \hat{\mu}_t^{(k)} \sim N(\mu_t^{(k)}, \sigma_t^{(k)2}), \quad \epsilon_t^{(k)} \overset{\text{iid}}{\sim} G_{\tau_t}.$$

the conditional likelihood is proportional to

$$\prod_{k=1}^{K} \prod_{m=1}^{N_T^{(k)}} f_{\theta^{(k)}}(X_m^{(k)}) \prod_{t=K}^{T-1} \prod_{k=1}^{K} \phi\left(\frac{\hat{\mu}_t^{(k)} - \mu_t^{(k)}}{\sigma_t^{(k)}}\right) \prod_{t=K}^{T-1} \frac{\exp[S_t^{(z_{t+1})}/\tau_t]}{\sum_{i=1}^{K} \exp[S_t^{(i)}/\tau_t]},$$

where the softmax terms come from the additional Gumbel randomizations.

I.3 Experimental results

We compare the bias and MSE of the original Thompson Sampling (TS) algorithm, and the debiased results after running cMLE. The debiasing runs 3000 gradient descent steps, 30 steps of MCMC with the first half as burn-in. The scale of the Gumbel distribution is 1.0.

Table 5: In the left table, we compare the bias of the original Thompson Sampling (TS) algorithm and the bias after running cMLE, for K=2 and K=5 arms, with different stopping values T. The left column is the bias of the original algorithm, and the right column is the percentage of bias that is left after running cMLE. In the right table, we compare the MSE of the original algorithm, data splitting (held-out), and cMLE. We see that data splitting suffers from high variance, and cMLE improves MSE.

	TS	
	orig.	cMLE
T=24,K=2	-0.19	18.7%
T=32,K=2	-0.17	20.5%
T=60,K=5	-0.23	37.3%
T=80,K=5	-0.11	28.8%

	TS		
	orig.	held-out	cMLE
T=24,K=2	0.32	130.0%	**90.0%**
T=32,K=2	0.28	110.0%	**77.0%**
T=60,K=5	0.34	123.0%	**85.0%**
T=80,K=5	0.16	125.0%	**62.0%**