"One Weird Trick" for Advertising Outcomes: An Exploration of the Multi-Armed Bandit for Performance-Driven Marketing

by

Giuseppe Antonio Burtini

B.A., University of British Columbia, 2013

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Science

in

THE COLLEGE OF GRADUATE STUDIES
(Interdisciplinary Studies)

THE UNIVERSITY OF BRITISH COLUMBIA
(Okanagan)

October 2015

Abstract

In this work, we explore an online reinforcement learning problem called the multi-armed bandit for application to improving outcomes in a web marketing context. Specifically, we aim to produce tools for the efficient experiment design of variations of a website with the goal of increasing some desired behavior such as purchases.

We provide a detailed reference, with a statistical lens, of the existing research in variants and associated policies known for the problem, then produce a set of theoretical and empirical analyses of specific application area questions. Concretely, we provide a number of contributions:

First, we present a new standardized simulation platform integrating knowledge and techniques from the existing literature for the evaluation of bandit algorithms in a set of pre-defined worlds. To the best of our knowledge, this is the first comprehensive simulation platform for multi-armed bandits capable of arbitrary arms, parameterization, algorithms and repeatable experimentation.

Second, we integrate Thompson sampling into linear model techniques and explore a number of implementation questions, finding both that replication of Thompson sampling and adjusting for estimative uncertainty is a plausible mechanism for improving the results.

Third, we explore novel techniques for dealing with certain types of structural non-stationarity such as drift and find that the technique of *weighted least squares* is a strong tool for handling both known and unknown drift. Empirically, in the unspecified case, an exponential decaying weight provides good performance in a large variety of cases; in the specified case, an experimenter can select a weighting strategy to reflect their known drift achieving state-of-the-art results.

Fourth, we present the first known oracle-free measure of *regret* called statistical regret, which utilizes intuitions from the confidence interval to produce a type of interval metric by replaying late-experiment knowledge over prior actions to determine how performant an experimenter can believe their results to be.

Fifth, we present preliminary results on a specification-robust and computationally efficient sampling technique called the Simple Optimistic Sampler which shows promising outcomes via a technique which requires no modelling assumptions to implement.

Preface

This thesis is the original and independent work of the author, Giuseppe A. Burtini. The research was identified, designed, performed and analyzed by the author.

Sections 3.2 (Linear Model Thompson Sampling: LinTS) and 3.4 (Nonstationary Time Series Techniques) draw heavily from the published work Burtini et al. [36] (2015a), where Drs. Jason Loeppky and Ramon Lawrence provided an advisory role.

A variant of the work which appears in chapter 2, in which Drs. Jason Loeppky and Ramon Lawrence provided an advisory and editorial role, has been submitted to Statistics Surveys Burtini et al. [37] and published on the preprint archive `arXiv.org`.

The work which appears in sections 3.3, 3.5 and 3.6 is intended to be submitted for external publication either in whole or in part at a future date.

All other work is unpublished as of this date.

The title "One Weird Trick" for Advertising Outcomes refers to a style of advertising popularized in 2013 after the acknowledgment of some influential experimental results in consumer psychology – highlighting just how fundamental, and even formulaic, the scientific approach of advertising has become. The language of "One Weird Trick" itself has become memetic in online advertising and even in a minority of academic work [97]. This work discusses an approach to performance-driven experimentation appropriate for scientific advertising.

Table of Contents

List of Tables

List of Figures

List of Symbols

This listing provides a reference for some of the common symbols used within this work.

$x_{i,t}$ The payoff received after selecting arm i of a multi-armed bandit process at time t.

x_i The payoff received after selecting arm i of a multi-armed bandit process explicitly assumed to be stationary in time.

E_θ Expectation taken over the distribution of an arm (equivalently, over the prior parameter θ for the arm distribution.)

$\mathbb{E}$ Expectation to be taken over both the random selection of a *a priori* fixed matrix of rewards and the actions of the player.

μ_i The mean payoff from arm i. An alternative expression of $E_\theta(x_i)$.

μ^* The highest mean payoff of an arm. Alternatively, $\max E_\theta(x_i)$.

K The number of arms available in a multi-armed bandit. Usually a constant positive integer.

H The horizon or number of time periods to be played in a multi-armed bandit. A positive integer or infinity, often unknown.

n The number of time periods consumed *thus far* when considering measures of regret taken prior to completion of the process.

n_j The number of time periods *thus far* in which an arm j has been selected. Equivalently, the number of observations of arm j.

S_t The sequence of arm selections made by the player.

R One of the many measures of regret. See Section 2.1.2.

Acknowledgments

An immense thank you to Drs. Ross Hickey and Jason Loeppky for their unwavering willingness (despite very busy schedules!) to consistently make time to hear my thoughts, listen to my concerns and contribute – in their interest, applications, excitement, ideas, motivation and encouragement – to my research and in a broader sense to my curiosity.

Of course, thank you to *all* my friends, but especially Graeme Douglas and Scott Fazackerley who have made a regular effort to remain informed and interested in my research, critically examine my ideas and writing and provide feedback of all varieties, positive and negative, as necessary. In many ways, critical feedback is both the hardest to find and, to me, the most valuable and these lifelong friends have been willing to provide it in every step of my work from inception to now.

Thank you to everyone *(and my bicycle!)* who put up with me when frustration, imposter syndrome or technical trouble pushed me to the edge and made me a little crazy, but especially to my parents, for reading my work after long hours of writing (even when it made no sense), for helping keep me on track in hard times and for their relentless lifelong support for me in absolutely everything I've ever tried to do.

Finally, thank you to Dr. Ramon Lawrence, my supervisor throughout the entirety of this work for both the opportunity to work in his research group and the incredible motivation he provides through his drive, organization, attitude, assurances and encouragement. Ramon is an incredibly talented and driven individual, and his drive is contagious in a way which directly produces inspiration in those around him and indirectly produces a phenomenal network effect of mutual inspiration among his students.

Chapter 1

Introduction

The real world has a wealth of circumstance where one must simultaneously explore their surroundings, options or choices while also maintaining or maximizing some variable: their output, well-being or wealth. Indeed the pursuit of a good life can be represented as finding a balance between "exploration," or the investigation of new options and "exploitation," the utilization of the knowledge one has already accrued. People make these decisions on a regular basis, from career and education ("should I continue to invest in education, exploration, or begin exploiting the knowledge I have accrued?") to shopping and selection ("should I purchase a product I have already experienced and am satisfied with, or should I explore other options?") and everything in between.

These problems have been studied extensively in a human decision making context, a game theory "maximizing" context, and the positive psychology "satisficing" context of increasing happiness through decision making. This type of decision making also arises in learning problems of the statistical and machine learning variety and is represented within a broad class of problems called *reinforcement learning.* One common form of this *exploration vs. exploitation tradeoff* is called the *multi-armed bandit problem* where the world can be imagined as a collection of slot machines, each with a different but unknown payoff, and the player must play, repeatedly, to maximize his wealth. This is the form of the *sequential decision making problem* for exploration vs. exploitation that we will explore in this work.

The model for this exploration vs. exploitation tradeoff that we consider in this work is called the multi-armed bandit or the multi-armed bandit problem. It describes the environment where an unknowledgeable player in a casino has to make repeated decisions about which slot machine (colloquially, a "one-armed bandit") to play in order to maximize his or her total reward. Peter Whittle, an early researcher in the area, captured the elegant application of the multi-armed bandit and the entirety of the importance of exploration vs. exploitation tradeoffs in his 1989 quote:

> *"Bandit problems embody in essential form a conflict evident in all human action: information versus immediate payoff."*

When the first efficient, tractable solution to the multi-armed bandit was presented [65], Whittle recalls [66] an early conversation with a colleague that highlighted the immense difficulty with which the problem was first seen:

> Colleague 1: *"What would you say if you were told that the multi-armed bandit problem had been solved?"*
> Colleague 2: *"Sir, the multi-armed bandit problem is not of such a nature that it can be solved."*

Indeed the problem was said [67] to have historically been seen as a weapon of war: Allied scientists proposed to deliver a related problem to Germany "*as the ultimate instrument of intellectual sabotage.*"

Before we begin with a more specific definition of the problem, a final quote to set the tone and a perspective which inspires the theory of the multi-armed bandit.

> *"To consult the statistician after an experiment is finished is often merely to ask him to conduct a post-mortem examination. He can perhaps say what the experiment died of."*
>
> — R. A. Fisher at the Presidential Address to the First Indian Statistical Congress, 1938.

1.1 Problem Definition

Often these exploration vs. exploitation problems arise in a business context, where the efficiency of such choices is a strong determinant of business success. This work addresses the challenges in multivariate experimental optimization in an online advertising firm environment. The theoretical lens which we develop for use within this context is a variant of the multi-armed bandit (MAB) problem. The multi-armed bandit problem, first[1] proposed by Robbins (1952) which built the new model on the sequential analysis work of Wald (1947) and others, has been extensively used to model the exploration-exploitation tradeoff in reinforcement learning and experiment design. The traditional multi-armed bandit problem is the prime example of a sequential exploration-exploitation tradeoff problem. In the problem, an agent aims to balance gaining new knowledge

[1] Thompson (1933) provides an answer to a related question: how to identify the probability of a distribution being better than all others from a set of distributions, and has thusly been sometimes credited as the origin of the multi-armed bandit. Even more confounding on the origins of the multi-armed bandit, Dr. Peter Whittle said in review of the 1979 paper of Gittins [67] the following "As I said, the problem is a classic one; it was formulated during the war, and efforts to solve it so sapped the energies and minds of Allied analysts that the suggestion was made that the problem be dropped over Germany, as the ultimate instrument of intellectual sabotage. In the event, it seems to have landed on Cardiff Arms Park. And there is justice now, for if a Welsh Rugby pack scrumming down is not a multi-armed bandit, then what is?" As World War II ended in 1945, this provides evidence that the problem was under discussion at least privately by the military if not elsewhere prior to the Robbins (1952) paper. Robbins (1952) is the first indexed paper to call the problem the multi-armed bandit and provides a formulation similar to the formulation used to date.

with exploiting its current knowledge. This applies in many variants in both commercial contexts — where vendors may wish to explore new products or new solutions while exploiting existing knowledge — and in numerous research contexts from clinical trials to simulating animal behaviour to adaptive network routing.

The specific application explored in this thesis is motivated from the view of an online marketing firm aiming to maximize its revenue through an efficient balance of exploration (trying new solutions, learning) and exploitation (utilizing learned knowledge to profit). Specifically, the model with which we approach the problem is one where there is a hierarchical funnel of traffic, with each stage having its own vector of parameters and specific objective, but a payout not being achieved until the visitor has successfully performed an action in all the stages. The stages are outlined as follows.

1. $T(\overrightarrow{t})$**: Traffic acquisition**. This level is where most of the demographic information about the user is decided. Variables at this level are either individually fixed (gender, geographic location) or fluid (time of day) from the perspective of the system and are either directly selectable (such as age or gender on some targeting platforms) by the targeting vector $\overrightarrow{t}$ or indirectly influenceable (such as factors related to state of mind, which may be influenced by the advertisement text and imagery, or immediately preceding thoughts, which may be influenced in aggregate by selection of a particular medium to purchase advertisements on). Every "trial" at this level has a cost which may or may not be fixed across repeated trials, depending on the particular media chosen. In one version of the model, traffic acquisition is assumed to come from a source that is much larger than the firm where the firm cannot appreciably influence either the supply or price of available traffic. In another version of the model, the firm is large in relation to the traffic source and traffic can be exhausted or prices increased as demand increases.

2. $P(T, \overrightarrow{w})$**: Pre-sales**. During pre-sales, the user is shown text, video and graphical content on a website principally designed by a marketing professional with the assistance of both domain expertise and statistical testing. The controllable factors are denoted $\overrightarrow{w}$, however, the dimensionality of this vector is extremely high with many of the variables going largely unexplored. This includes variables ranging from color to word choice to ordering and page design and layout and much more. In many contexts, the pre-sales process may involve other interaction including e-mail or phone call lead generation. In our model, the pre-sales function P is influenced by the result of the traffic acquisition step in a high-dimensional way: specifically, which users are selected affects the impact of each variable in the feature vector $\overrightarrow{w}$. This is the level at which we focus the majority of our experimentation, however, it is easy to extend this to testing at the targeting vector or product selection level.

3. $C(P, \beta)$**: Conversion**. This is the final step where the user becomes a customer. As the

pre-sales step is often designed to be general, this step may choose a different product β (with a fixed vector of features, which we do not control and can only observe with uncertainty) for different users or different pre-sales vector results. In the case of an indirect marketing firm relationship, this may be the set of steps after pre-sale and product selection not controlled from within the system, such as price or order form design. In a more direct consulting relationship, all steps leading prior to sale are under control and this step is simply the recording of a success condition, with a given value, in the bandit system.

The goal at any step of the process is to bring the user to the next step successfully. Broadly, our task is to choose $S = \{\overrightarrow{t}, \overrightarrow{w}, \beta\}$ from a finite[2] set of alternatives to maximize the sum of time-discounted revenue and satisfy competitive business objectives such as risk tolerance and ethical factors. In practice, this is generally achieved with experts selecting each of the choice variables. In our model, we explore the algorithms necessary to replace or augment this expert with an automated experiment.

In a real world application with risk management considerations, it is often important to consider the problem within the context of "runs" and "horizons." Specifically, for a given run of the problem, the system will settle on a specific group of parameters that can be expected to run for some horizon before exogenous factors change the state of the world or the business moves on. For a variety of reasons, it may not be optimal to maximize immediate profit with that consideration, if the selection of parameters that currently maximize profit will influence how future users respond. In modern business, this takes the form of ethical constraints (including "price gouging" and discriminatory pricing) and an expectation of certain constraints on the pre-sales vector $\overrightarrow{w}$. These competing objectives, from legal concerns to ethical constraints and other-than-profit business objectives, complicate the problem significantly.

Fundamentally, this problem (and many applications of the multi-armed bandit problem) is at the intersection and forefront of economics, operations research, game theory, decision theory, statistics and computer science in the way it combines computational learning and statistical estimates with an appropriate level of dynamic decision making and risk-coordination in a real, applied business environment.

[2] In a variant of the problem, some of the variables are discrete and finite and others are theoretically continuous and unbounded, such as color or font size selection; in practice, most selection parameters can be considered bounded to a relatively small range. Similarly, most observed factors about our users or experimental subjects (covariates or contextual variables) are categorical or easily discretized.

1.2 Motivation

1.2.1 Specific Interest Area

The area of online marketing can be considered one where a semi-random[3] flow of users (each treated as an experimental subject) arrive to a website, — each with its own context vector which describes properties of uncertain relevance to the objective function considered — the users are delivered a *treatment* (a variant of the "pre-sales" vector) and then at some time in the (near) future, a reward (e.g., a purchase of a product) of variable value (possibly zero) is observed for that user. Each user has an explicit cost to the business that can range from very small (e.g., if they arrived via search engine traffic or word of mouth) to a large fraction of total expected revenue (e.g., if they arrived to the website via a pay-per-click or pay-per-lead service).

As a concrete, but by no means limiting, example, we can imagine the *treatment* being a selection between a red button and a blue button, with our objective (reward) being 0 or 1 indicating whether a user clicked the button or not. The colour of the button could plausibly effect the likelihood of drawing the user's attention and thus attracting their click. The variants can range in scale from being small micro-experiments (such as buttons, language choice or colours) to fully transformed versions of a website, as long as the objective (reward) is well-defined for all variants.

The motivation in exploring this area is one of improving business performance on the web. There is clear room for improvement in how testing is currently performed within the marketing context. The environment is well-modelled by the multi-armed bandit, but in the applied context, there is little application of the multi-armed bandit to this space.

Bandit strategies, in particular, appear especially useful for performing experimentation with the same type of complicating factors that arise in web experiments. In relatively small sample size environments, for example, by properly balancing exploration and exploitation, multi-armed bandit algorithms can achieve payoff rates that significantly exceed that of traditional testing models in general, but especially in the case where a traditional model would not have yet finished its experiment. Additionally, many of the explored bandit strategies do not require a fixed exploration period *a priori*, which is more appropriate in a business context where total users, horizon and other variables may not be predictable.

Importantly, the bandit model, as we will explore in future sections, generally does not require selection of any fixed significance level or other statistical assumptions prior to application. This is important, as experiment designers in the online advertising context may not be statistically sophisticated and simply wish to apply a prebuilt solution to their exploration.

While web experimentation is the motivating field for the remainder of this work, we do not experiment specifically in a "real world" environment, rather, we work within a simulation envi-

[3] This level of the funnel can often be controlled, at least in part, by targeted advertising and selective user acquisition mechanisms.

ronment to test and resolve a number of theoretical and empirical questions that are relevant to the application of the multi-armed bandit in a variety of environments.

1.2.2 Other Applications

Online advertising is not the only appropriate application of the multi-armed bandit problem. While not the focus of this work, the literature is ripe with many applications, many of which may benefit from a hierarchical or contextual analysis and provide further inspiration for immediate application. Most of the other applications discussed here are predictive or prescriptive in the sense that they act as a decision theory mechanism, however, there is a range of descriptive literature which utilizes bandit models as tools to attempt to explain empirically estimated behaviour [24, 25, 84, 91, 110, 127]. These are not explored in detail as they do not share much analytical similarity with the problem at hand.

Clinical Trials

One of the most widely discussed applications of the multi-armed bandit problem is that of differential allocation in clinical trials [82, 153]. In a clinical trial, real patients are being treated with medications with unknown properties – selecting between which is an exploration-exploitation tradeoff problem. In the multi-armed bandit application, the goal is to simultaneously learn the properties of the medication (explore) and cure the patients (exploit): as specific medication starts to show statistical promise, it could be prescribed to a larger fraction of the patients creating more positive outcomes in expectation. The ethical implications of minimizing tail-effect losses, constraining total regret and monitoring for model error make this an interesting research area with high value impact.

The medical domain is particularly amiable to a variant of the multi-armed bandit problem where there are covarying and confounding factors. In this variant, patient criteria for admission, selection into a particular trial or particular medication and patient factors such as age, sex and other variables may influence treatment or interact with the medication in a meaningful way. The ability to consider this contextual information is also important when it can provide a more rapid or more accurate understanding of the world, as there is an extremely high cost associated with errors in this domain.

Adaptive Routing

The multi-armed bandit problem has been applied in the context of learning ideal routing paths through a computer network. This is specifically interesting as it can be phrased as a hierarchical (and hierarchical-contextual, if we have a vector of features for each node) problem traversing a directed (or undirected) graph. Simply, each route is tested and median time, variance or other objective value of interest is monitored and minimized.

Portfolio Design

Hoffman, Brochu, and de Freitas [77] propose a mechanism that relies on a multi-armed bandit strategy to select among competing portfolio selection strategies called acquisition functions. This indicates a common theme in multi-armed bandit applications where competing mechanisms (or experts) are chosen via a bandit process. Sorensen [136] explores modelling venture capital decisions within a bandit framework.

Natural Resource Exploration

In a number of natural resource exploration contexts, bandit models with side-information have been applied to balance "earning" (exploitation) and "learning" (exploration) within the search procedure. Given a fixed set of resources, a firm's decision between further exploration within their region and the harvest of the sites with maximal expected value based on currently observed information has been compared primarily in oil exploration [20, 21, 28]. There is room for similar application in this space for mineral exploration and optimal farmland utilization exploration as well as room to extend the model in general to support covariates of a planar type such as coordinate and geographic covarying parameters.

Research and Development Investment

Weitzman [158] introduced an example of the bandit problem in which a firm explores two competing technologies, the benefits of which are uncertain, to produce a commodity at a minimal cost. A substantial amount of work since has produced further development in how firms can optimally balance exploration and exploitation [45, 122] in research and development expenditure. Some of the literature extends the concept of the bandit model to have "switching costs" between arms [17], representative of the often extremely high cost of changing technique or investigating new production methods in a commercial context.

Employee Resource Allocation

Related to the research investment problem is the general problem of human resource management. Farias and Madan [58] produce a modification of the contextually informed model where arms cannot be discarded once pulled and utilize packing problem type heuristic to admit a practical solution. This "cannot be discarded" trait approximates the hiring problem for a variable skill, large number of staff employment environment. Many other papers have investigated the employee resource allocation problem within a variety of contexts: allocating tasks to staff, allocating staff to departments, selecting staff, and more. This is very different than the exploratory stochastic model we explore, but it demonstrates the generality of the model.

Crowdsourcing

Related to the employee resource allocation problem, crowdsourcing has recently become a popular solution for collecting and distributing human capital for a variety of creative and computational problems from graphic design[4] to microcomputation to research task allocation. Jain, Gujar, Bhat, Zoeter, and Narahari [80] introduce a multi-armed bandit model for assuring accuracy and minimizing total cost in a crowd-worker environment. The mechanism they propose, called Constrained Confidence Bound, is one that identifies "quality consensus" to combine multiple competing advisors in an cost-optimal way. This problem arises again in our work when we introduce the expert system to our training function. The problem of combining competing expertise has been considered extensively in the psychology and management literature [85] however, rarely have actionable decision mechanisms been proposed. Recently, a variety of other researchers have produced adaptive algorithms for multi-armed bandit treatment of the crowdsourcing and task assignment problems [2, 75, 146].

General Real World Explore/Exploit Tradeoffs

Dumitriu et al. [54] discusses an approach to the multi-armed bandit problem in the stylized context of professional golfers selecting a brand of golf ball. The real world is ripe with exploration-exploitation tradeoffs from selecting optimal fishing tackle for an angler to deciding upon an optimal career path for a student, many of which involve hierarchical or context-informed decisions. A sufficiently simple algorithm to be performed by humans or a decision making toolset for professional management could provide significant guidance in a variety of practical situations.

1.3 Research Contribution

In this section we enumerate and begin to motivate our research contributions, which include a standardized simulation platform, a set of new policies or algorithms for the multi-armed bandit problem and some insight into an online measurement of regret and so-called *optimism in the face of uncertainty.* Each of these contributions takes the form of concrete experimental, developmental or algorithmic work. In addition to these explicit contributions, the background section provides a significant contribution itself: an extensive survey of the literature, concisely enumerating and providing a taxonomy for much of the significant progress throughout the space of multi-armed bandit problems and their variants.

1.3.1 A Simulation Platform for Multi-Armed Bandits

To date, there is no standardized simulation platform for comparing stochastic multi-armed bandit policies. We produce a repeatable, parallelized and extensible platform for experimentation and

[4] For example, the online business 99Designs uses crowdsourcing to operate contests for producing logos, website layouts and other design elements.

a set of standardized *worlds* for comparisons. The count and associated parameters of underlying payoff distributions are configurable and support non-stationarity in both changepoint and drift fashion, mixed distribution models and contextual covariates. Further, the platform supports integration with the PASCAL Exploration vs. Exploitation (EvE) Challenge dataset [79] and the Yahoo! real world news click-through data [103], two popular datasets for comparing the performance of stochastic bandit policies.

The importance of a preset standardized set of experiments is hard to overstate. Researcher selected experiments are prone to an unintentional form of selection bias which may result in a loss of the quality of results and a slower pace of research understanding and progress. While our available worlds aim to be general in many dimensions and violations of assumptions, the platform is extensible in a way that allows the creation of new worlds and new sets of experiments easily.

The platform computes as many meaningful empirical measures that we could define, including everything computable presented in Section 2.1 of this work. This includes all the computable variants of regret, variance and empirical confidence intervals of regret and other measures, divergence measures of difficulty, computation time and more. We include a small set of data visualization tools to transform the detailed iterate- and replicate- level outputs of the simulator into graphical representations of regret over iterates (for evaluation of small sample and convergence behavior) and regret comparisons across varying categorizations of the underlying world (assumption violations, distributions, or size/scale of the problem) and policy or policy configuration.

This platform emerges as the tool which drives our further experimentation. The extensible and repeatable nature allows us to rapidly develop tests for a variety of application-level and parameter-level questions, as well as experiment with new policies and variants of existing policies.

1.3.2 LinTS: The Regression Thompson Sampler

We first extend a popular policy called LinUCB [101] to solve a variant of multi-armed bandit problems with (linear) covarying factors called the contextual bandit problem. This extension replaces a fixed substrategy called UCB (upper confidence bounds) with a technique from the efficient probability matching paradigm proposed by Thompson [144], producing an easily implemented, tractable, regression-based Thompson sampler which we call Linear Thompson Sampling (LinTS). We then utilize the simulation platform to show experimentally how various choices in LinTS affect the final results in terms of measures of success.

1.3.3 Experiments in Thompson Sampling

We return to analyzing a number of application area questions in Thompson sampling, especially those related to the impact of *optimism in the face of uncertainty* to the policy, finding some interesting trends in the performance of such a model. We show that *excessive optimism*, to an extent not previously considered in the literature, may provide a beneficial approach in certain probability matching environments, and we discuss the plausible implications of this result on the understanding of optimism. We further find a number of interesting results in this light including a strict reduction in regret from a prescient uncertainty correction — to the extent it is available — which applies in the regression sampling case and again consider the interesting interaction with this effect and optimism in light of our understanding of optimistic exploration. Together, these results provide both a guidance in the implementation-level decision making with regard to the deployment of an efficient bandit process and an interesting theoretical direction for the understanding of optimism.

1.3.4 Time-series Techniques for Non-Stationary Bandits

Extending our work in defining LinTS, the regression-based Thompson sampler, we explore a number of time-series approaches to removing underlying trends and non-stationarities in the rewards process. One especially promising technique is an exponentially-weighted decay process which allows the handling of *a priori* unknown forms of underlying drift and performs well on all tested variants of drift. Further, we show that if there is knowledge of the form of drift, one can perform near optimally using an appropriately calibrated decay technique. We explore a number of other plausible approaches to time-series detrending that do not find significant promise in the unknown drift-form application.

1.3.5 Statistical Regret for Applied Bandit Models

In general, the measure used to quantify the result of a bandit process is the *regret* – that is, a loss function type measure which captures the difference in the largest achievable payoff and the payoff actually achieved. To compute regret it is necessary to know the largest achievable payoff – that is, to have an *oracle* which provides the correct answer for comparison. Outside of a simulation environment, this is not a computable quantity: if an oracle was available, there would be no need to perform online exploration or the associated exploration vs. exploitation tradeoff decisions. We define a *prediction interval*-based measure which allows the *ex-post* computation of a calibrated regret upper and lower bound. We show the interval behaves like a traditional confidence interval, allowing the user to configure γ, the interval's confidence level which produces a tradeoff between tightness of the interval and the frequency with which the interval correctly includes the true value.

This is a significant contribution for the measurement, diagnostics and parameter selection of multi-armed bandit policies in an applied context. By the nature of the problem, there is little room for traditional experimentation to identify the ideal parameter structure and an online analytical method is required. Statistical regret shows that an evaluation process for the quality of parameter selection can be produced and that differing individual runs of a bandit process can be examined and compared quantitatively in practice.

1.3.6 Simple Efficient Sampling for Optimistic Surrogate Models

The principle of optimism in the face of uncertainty arises repeatedly in multi-armed bandit policies. Similarly, the principle of probability matching, especially the probability matching sampling technique of Thompson [144] is a regular tool used in the treatment of these problems. Combined, we provide a treatment of optimistic sampling that provides easily implemented efficient techniques which both accurately capture the intended optimistic model and perform in constant computational complexity. First, we present the most simple case, a technique for sampling optimistically from a symmetric distribution, then we extend that to a sampling technique for the optimistic surrogates of any generalized nonsymmetric distribution. These techniques, while both simple, are important, as some results in the literature have understated the effect of optimism by erroneously selecting a sampling technique which, while optimistic in the strict sense, does not accurately capture the intended surrogate model.

Finally, in this section, we provide a new technique for producing a distribution-free sampler which can take on both optimism and contextual variables as augmenting behaviors. We show with experimentation that this technique is at least as strong (in terms of regret) as the parametric, distribution-dependent sampler, while providing increased robustness to misspecification.

1.4 Outline

The rest of this thesis proceeds as follows. In Chapter 2, we rigorously introduce the formal problem, introduce the different variables that are worthy of consideration for each potential selection policy and explore the policies that have been applied in the literature thus far, and finally discuss a selection of variants to the pure formal problem that are relevant to our application. When discussing policies, we have chosen to err on the side of intuitive and applicable explanations leaving any exposition of correctness or asymptotics to the referenced papers. In Chapter 3, we discuss our research contributions from both a theoretical and applied perspective and integrate the results into the existing literature. Finally, in the last chapter, we discuss future work in a broad sense and explore avenues for further research that would both advance the state of the theoretical literature and provide large value in an economic sense.

Chapter 2

Background

Chronologically, Robbins [125] introduces the idea of the important tradeoff between exploration and exploitation in recurring decision problems with uncertainty, building on the prior *sequential decision problem* work of Wald [154] and Arrow et al. [8]. Lai and Robbins [99] produce the first asymptotic analysis of the objective function of such a decision process showing a bound of $O(\log t)$ for the regret of the standard stochastic, finite-armed, multi-armed bandit, and produced a regret-efficient solution where the rewards of a given arm are stationary and *independent and identically distributed* (i.i.d.) with no *contextual information* or covariates. In the coming sections, we will explore some of the large number of algorithms which have been proposed since Lai and Robbins [99] including ϵ-exploration approaches [152, 156], upper-confidence bound techniques [16], probability matching techniques [6] and others. Many variants of the initial problem have also been investigated in the literature including the many- or infinite-armed, adversarial, contextual, and non-stationary cases.

Initially, bandit problems were discussed in the context of the *sequential design of experiments* [99, 125] and *adaptive experiment design* as a natural progression of the experimental design literature which seeks to design efficient tools for identifying causal effects. Recently, a policy or decision-theoretic lens has been used, characterizing the problem as a recurrent policy selection to maximize some utility function or minimize some regret function.

The terminology “bandit” originates from the colloquialism “one-armed bandit” used to describe slot machines. It is important to note immediately that this stylized terminology suggests the negative expected value embodied in the context of the word “bandit,” despite that not being a requirement, nor being common in most applications of the model. In one common formalization of the multi-armed bandit model, the player can sequentially select to play any arm from the $K \geq 1$ arms and obtain some payoff $x \in \mathbb{R}$ with some probability $p \in [0, 1]$. This specific formalization has explicitly defined binomial arms (fixed payoff, random variable payoff probability), however in our interpretation of the problem, the distribution on the arms can (and often is) be from any distribution.

In all variants of the bandit problem only the payoff for the **selected** arm at any time step is observed and *not* the payoffs for non-selected arms. This is the "partial information" nature of the bandit problem which distinguishes the problem from generalized reinforcement learning, where "full information" (observing the payoff of all arms, whether selected or not) is usually assumed.

In many variants of the problem, the reward distribution itself is not assumed, however, rewards are assumed to be *i.i.d.* across arms and across prior plays. Some variants of the problem relax both of these assumptions. Similarly, it is generally assumed that both K, the number of machines (or "arms") and x, the payoff function are finite and stationary (time-invariant), however, variants relaxing these assumptions have also been considered. An important variant of the problem introduces the concept of a known, finite time-horizon, H to bound play time. Two important variants, the contextual problem and the adversarial problem, remove assumptions such that there are fewer or even no statistical assumptions made about the reward generating process.

In our motivating web development example, the "arms" in the multi-armed bandit become the variants of a website being experimented with. For example, there may be three arms: one where a red button is displayed to the user, one where a yellow button is displayed and a final variant where a blue button is displayed.

2.1 The Stochastic Multi-Armed Bandit Model

2.1.1 A Stylized Example

Throughout the rest of this work, an example will be valuable to elucidate variants, applications and algorithms. Imagine yourself walking into a casino and being presented with a row of slot machines. Prior to your visit, you had absolutely zero knowledge about the payoff probabilities of the machines: they may have negative, positive or neutral expected value, they may all be the same or they may not. Your prior information is complete uncertainty[1]. The challenge, broadly, is to identify how to play the machines to maximize your return. This is a challenge of balancing exploration (collecting new, high-value information to minimize uncertainty) with exploitation (using the knowledge you have already acquired to gain the expected reward). An obvious intuitive solution, is to start recording results and play each machine some fixed number of times (or randomly) to get an estimate of the payoff rate of each machine, then, if the expected payoff is positive, play the strongest machine indefinitely into the future. This solution is very close to the strategy known as ϵ-first, which, while intuitive, has a number of undesirable properties and can be outperformed by more sophisticated strategies which we will explore in more detail later in this chapter.

[1] However, in some Bayesian bandit contexts, perhaps you wish to consider the variant where you have prior information and wish to explore further.

2.1.2 Considerations

In the following section, we explore some of the important considerations and measures which vary across different problems and policies.

Measures of Regret

There are a number of possible objective functions to evaluate a bandit process that have been used within the literature. In general, regret-based objectives are loss functions which compare the performance of a process with an oracle (a policy which *ex-ante* knows all information about the underlying distributions) over a finite and discrete time horizon $t = 1, ..., H$. The definitions vary on how the stochastic nature of the process is considered and what information and selections are made available to the oracle. The sequence of arms selected S_t can itself be a random variable (such as in the Thompson sampling case discussed later). Expectation E_θ is taken over the parameters of the arm distribution(s) θ which is taken as fixed per-arm and time *a priori*.

For most practical applications of multi-armed bandit models this is the most important conceptual variable to consider. Unfortunately, regret is not consistently defined in the literature as a single variable of interest, but generally takes one of a variety of forms. In a pragmatic sense, regret has the same meaning as it has in English: the remorse (which would be) felt as a result of dissatisfaction with the agents choices. We enumerate some of the varieties of regret that are discussed (explicitly or implicitly) within the literature here.

1. **Expected-Payoff (Strong-) Regret ($\bar{R}^P$).** This is the difference between the expected payoff an oracle would have earned, selected per play ($\sum \max_{i=1,2,...,K} x_{i,t}$) and what the algorithm being tested ($\sum x_{S_t,t}$) earned. Formally,

$$\bar{R}^P = \sum_{t=1}^{H} \left(\max_{i=1,2,...,K} \mathrm{E}_{\theta_{\mathrm{i,t}}}[x_{i,t}] \right) - \sum_{t=1}^{H} x_{S_t,t}. \tag{2.1}$$

In this measure, regret is a random variable in both θ and S_t. It is possible to produce an $\bar{R}^P$ that is negative (indicating that the policy outperformed the expectation), and it is possible for two runs to display regret incongruent with the best policy in expectation. The Figure 2.1 shows a two-arm example where expected-payoff regret would compute a regret of negative 2 (for a single play) assuming the policy picked Arm 1 and drew a value at the solid vertical green line. The dashed lines indicate the expectation. This regret of $-2 = 8 - 10$ is despite playing the suboptimal (in expectation) arm (arm 1). This measure has also been referred to as *empirical regret* by, for example, Eckles and Kaptein [55] but other works such as Maillard [104], Seldin, Szepesvári, Auer, and Abbasi-Yadkori [134] have used the term empirical regret with a range of different meanings.

The main motivation for this form of regret is its ability to compute meaningful results appro-

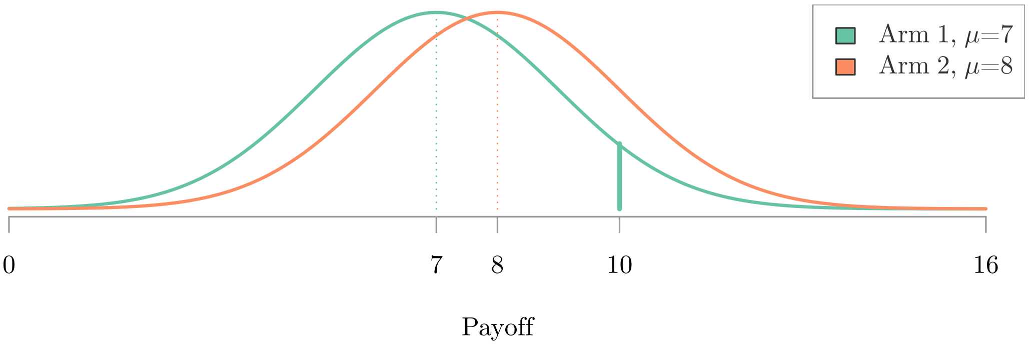

Figure 2.1: An example of playing an expected-suboptimal arm but achieving a high reward due to random variation. The solid vertical line indicates the payoff realized ($x_{1,t} = 10$) from the selected arm ($S_t = 1$).

priate for risk-aware bandits (such as an application in medical trials with risk limitations or portfolio design with maximum drawdown limitations). Some interesting statistics possible with this form of regret include the standard deviation and γ-percentile.

2. **Expected-Expected (Strong-) Regret ($\bar{R}^E$).** Similar to expected-payoff regret, this form takes the expectation of the arm payoffs in addition. In simulations with well-defined expected values, expected-expected regret can quantify the achievement of the policy without considering statistical variance. Formally,

$$\bar{R}^E = \sum_{t=1}^{H} \left(\max_{i=1,2,\ldots,K} \mathrm{E}_{\theta_{\mathrm{i,t}}}[x_{i,t}] \right) - \sum_{t=1}^{H} \mathrm{E}_{\theta_{\mathrm{S_t,t}}}[x_{S_t,t}]. \tag{2.2}$$

In general, existing regret proofs hinge on this definition of regret. Expected-expected regret may be a *biased* measure of the true parameter of interest in the case of a non-symmetric sampling process (S_t) when there is a focus on risk (such as in the case of a drawdown sensitive financial market policy or other loss-mitigation objective function). In this measure, regret is no longer a random variable in θ. An arm selection fully determines the regret measure for a single draw, independent of the drawn value. In the stationary case, the expectation $\mathrm{E}_{\theta_{\mathrm{i,t}}}$ can be written without the t in the subscript as the distribution is not dependent on time. In our prior example, if arm 1 (the suboptimal arm) is selected and a payoff of 10 is observed as shown, the expected-expected regret will still be $\mathrm{E}_\theta[x_2] - \mathrm{E}_\theta[x_1] = 8 - 7 = 1$.

3. **Adversarial Regret ($\underline{R}$).** Adversarial regret, also called "weak regret", variants can be

constructed of each of the prior definitions of regret. In adversarial regret, the oracle must pick only one arm for all plays. Specifically, the maximization operator is moved outside the summation to select only the single arm which is optimal in aggregate over all plays. In summary, adversarial regret measures the difference in the single best arm being played repeatedly and the choices the algorithm made. Formally,

$$\underset{\sim}{R} = \left(\max_{i=1,2,\ldots,K} \sum_{t=1}^{H} \mathrm{E}_{\theta_{\mathrm{i,t}}}[x_{i,t}] \right) - \sum_{t=1}^{H} \mathrm{E}_{\theta_{\mathrm{S_t,t}}}[x_{S_t,t}]. \tag{2.3}$$

With the maximization outside the summation, we select a single arm *i for all t*, a result that compares played arms only to the best in aggregate, naive to per-iterate changes in the ideal arm.

This is used most often in the adversarial bandits literature where "best arm per play" regret results may no longer be ideal due to the assumptions of the model, but it is also sometimes used in non-adversarial literature for reasons of accident or provability. In particular, in the adversarial regret model, the learning problem can be made arbitrarily hard under traditional measures of regret, for example, a large number of low payoff arms may be provided among one high paying arm selected randomly. In a traditional measure, this would result in a high level of regret while in the adversarial measure, only the arm which has the highest total payoff over all iterates is considered.

4. **Simple Regret ($\vec{R}$).** Like adversarial regret, simple regret variants can be constructed of each of the prior variants of regret. Simple regret contrasts other regret measures by being non-cumulative, that is, it only considers the regret at the current time. Simple regret at time H is the difference in the selected arm and the best arm at that point in time, with no consideration for prior plays. Formally, for expected-expected simple regret,

$$\vec{R}^{E} = \left(\max_{i=1,2,\ldots,K} \mathrm{E}_{\theta_{\mathrm{i,H}}}[x_{i,H}] \right) - \mathrm{E}_{\theta_{\mathrm{S_t,H}}}[x_{S_H,H}]. \tag{2.4}$$

The intuition of the value of simple regret is in the limiting properties of a well-behaved bandit algorithm. It is expected that a bandit algorithm will, in the probability limit, converge to play the best arm, and as such, simple regret will converge to zero. In some variants of the problem, such as infinitely-armed bandits [39], simple regret is available where other regret measures are not. A common variant of simple regret which draws on the same intuitions is *long-term average regret* defined as any traditional regret measure (say expected-expected regret) divided by the horizon ($\bar{R}^{E}/H$), as the horizon tends to infinity.

5. **Bayesian Regret ($\bar{R}^{\mathrm{Bayes}}$).** The election science literature (among others) gives us a technique called *Bayesian regret* that lends itself well to bandit analysis through the frequent

application of Bayesian-derived strategies. Bubeck and Liu [31], Kaufmann et al. [89] and others provide the first use of variants of this measure in a bandit context. The difference in Bayesian regret and the traditional regret measures is only that the parameters of the distributions ($\theta_{i,t}$) are taken explicitly to be themselves drawn from some prior distribution parameterized on Θ. Formally,

$$\bar{R}^{\text{Bayes}} = \mathrm{E}_{\Theta}[\bar{R}], \tag{2.5}$$
$$\theta_{i,t} \sim f(\Theta).$$

Many bandit policies (especially sampling-based techniques) are inherently Bayesian in nature (despite often being analyzed in a frequentist framework) lending Bayesian regret credibility as a reliable formalization of the concept.

6. **Suboptimal Plays ($N_\vee$).** Suboptimal plays is a simple measure of regret accrued by counting the number of plays that were *not* the best arm. This does not account for the scale of the difference. Formally,

$$N_\vee = \sum_{t=1}^{H} \mathbb{1}[S_t \neq \operatorname*{arg\,max}_{i=1,2,\ldots,K} \mathrm{E}_{\theta_{i,t}}[x_{i,t}]]. \tag{2.6}$$

   In the simple (stationary, context-free) two arm case, suboptimal plays is a sufficient measure for comparing algorithms as the gap between all suboptimal decisions (there is only one) is the same. In the three arm case shown in Figure 2.2, we can see that arm 3 is much worse than the next best arm (arm 1), however, if either arm is played, we increment a simple counter of suboptimality.

   The biggest advantages to suboptimal plays over other measures is that of unit independence and scale invariance. This simple measure provides a count of plays that were not *the best* with no consideration of how suboptimal they were.

Two further definitions of regret show up in the literature, generally referred to as *expected regret* and denoted $\mathbb{E}R$ and *pseudo-regret*, referred to as $\bar{R}$. These arise more frequently in work that is derived from the adversarial and non-stochastic approaches to representing bandit problems as they require a different perspective on the problem than we take. Specifically, if a bandit process is represented as a fixed *a priori* matrix of dimensions H (time horizon) by K (number of arms) where each entry represents a payoff with possibly fixed structure[2], we can eliminate the underlying

[2] Structure allows representation of stationary or non-stationary payoffs, distribution-derived payoffs and other complicating factors.

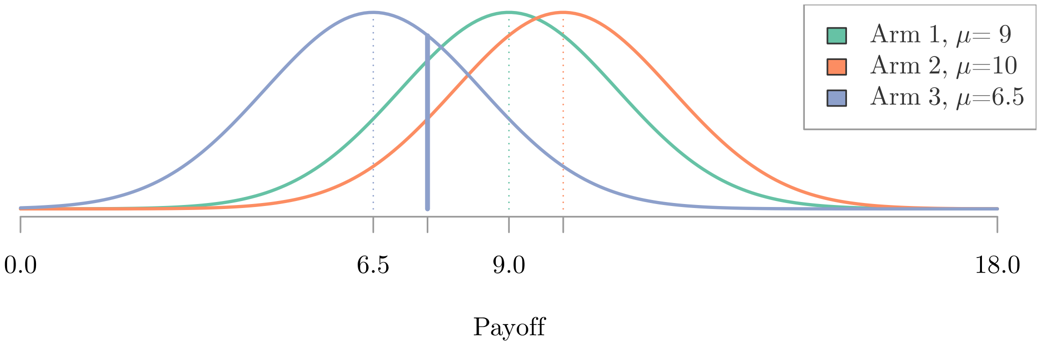

Figure 2.2: A play from arm 3 (the *most* suboptimal with expectation of 6.5) is drawn earning a reward of (7.5). Giving an expected-expected regret of $10 - (6.5) = 3.5$, an expected-payoff regret of $10 - (7.5) = 2.5$ and a suboptimal plays regret of just 1, a counter for the number of times an arm other than the optimal was played. The solid vertical line indicates the payoff realized ($x_{3,t} = 7.5$) from the selected arm ($S_t = 3$).

essential randomness from the consideration[3]. We use a capital X to denote a two dimensional matrix of the form previously described.

7. **Expected Regret ($\mathbb{E}R_n$)** [30]. Expected regret is treated as the gold-standard in the literature which uses it. This is distinct from what we are calling expected-expected regret ($\bar{R}^E$) in what is meant by the word "expected." Formally,

$$\mathbb{E}R_n = \mathbb{E}\left[\max_{i=1,2,\ldots,K} \sum_{t=1}^{n} X_{i,t} - \sum_{t=1}^{n} X_{S_t,t}\right]. \tag{2.7}$$

In this case, expectation is taken both over any randomness in the creation of the rewards matrix $X_{i,t}$ and any randomness in the selection of S_t. As such, this produces a measure of regret which requires the agent to be prescient to any purely random factors in the reward matrix in order to achieve zero regret. In other words, it is no longer sufficient to select the arm with the highest *expected* payoff (expected in what is possible to know) but rather one must select the arm with the highest *actual* payoff in order to achieve zero regret.

This way of thinking eliminates any stochastic nature from the forecasting discussion, as we measure against the ideal world state for a particular instance of the game (inside the expectation) and then average that over all possible world states (outside the expectation).

[3] This is in some sense an operationalization of a bandit process represented with true randomness, as in practice, the randomness is implemented with a pseudo-random number generator which amounts to a vector of fixed *a priori* numbers derived from some seed.

This perspective of evaluation is distinct from the techniques generally applied to perform forecasting in a stochastic bandit, where knowledge is expressed with uncertainty *and* the underlying world has some inherent (unexplainable) uncertainty.

8. **Pseudo-Regret ($\bar{R}_n$)** [30]. This measure is identical to the regret we refer to as *adversarial regret* ($\underline{R}$) with the exception that the definition of expectation differs. As in the previous definition, expectation here $\mathbb{E}$ is defined to be expectation both over the creation of the random rewards matrix and over any randomness in the selection process – that is, it produces a weighted average over all possible world states, while in *adversarial* regret expectation is pushed in to be over the *reward distribution.* Formally,

$$\bar{R}_n = \max_{i=1,2,\dots,K} \mathbb{E}\left[\sum_{t=1}^{n} X_{i,t} - \sum_{t=1}^{n} X_{S_t,t}\right]. \tag{2.8}$$

In this case, the optimal decision is defined only in expectation. In the stationary, context-free stochastic setting, this reduces to a simpler equation where μ^* is the mean payoff of the best arm, that is, using our earlier definition of expectation: $\mu^* = \max_{i=1,2,\dots,K} E_\theta[x_i]$. Formally,

$$\bar{R}_n = n\mu^* - \sum_{t=1}^{n} \mathbb{E}\mu_{S_t}. \tag{2.9}$$

Furthermore, Audibert and Bubeck [11] relate these two quantities and show that $\bar{R}_n \leq \mathbb{E}R_n$ via an argument reliant on Jensen's inequality and in the stationary, context-free stochastic case, further show the gap is bounded by $\mathbb{E}R_n - \bar{R}_n \leq \sqrt{\frac{n \log K}{2}}$.

In the rest of this work, we view the bandit problem as an exclusively stochastic problem (from the perspective of the agent) where arms are represented as unknown distributions and thus avoid utilizing these two measures of regret.

As we will show in the following sections, in experiments, the selection of regret measure is paramount to the understanding of policy decisions.

The variable definitions of regret are a serious issue in comparing bandit algorithms in many contexts. In much of the published research, it is difficult to identify which definition of regret is utilized, as it is quite easy to phrase a plain language definition of regret in a form that leaves the oracle criteria ambiguous[4].

Lai and Robbins [99] demonstrated that the lower bound of regret for policies for the general stochastic multi-armed bandit problem was $O(\log t)$. This lower bound does not hold for all variants

[4] As an example – though definitely not to single out these authors, as ambiguity of this sort is common – a sentence defining regret as "the difference between the sum of rewards expected after N successive arm pulls, and what would have been obtained by only pulling the optimal arm" appears in Granmo and Berg [70] and is not in itself clear whether "only pulling *the* optimal arm" means the singular mean optimal arm (as in pseudo-regret) or the variable per-play optimal arm (as in strong regret) or even the prescient form of expected regret.

of the problem, and indeed it is easy to construct synthetic variants of the problem with zero minimum regret.

It is important to note that regret is further complicated in a number of variants of the problem. Broadly, regret is only a simply defined concept in the case where there is a single best arm (or a set of indistinguishable best arms). In any variant of the problem where the best arm changes over time, plays or context or in any variant where the payoff choice is chosen adversarially or from a non-stationary distribution, there will be substantial complications to the analysis of regret, in particular with regard to what information the oracle has available to it.

Variance and Bounds of Regret

Similarly to the well-known bias/variance tradeoffs in the estimator selection literature, variance minimization is an important consideration for many applications. In the most extreme case, a high-variance algorithm will be completely unacceptable in a practical sense even if it had a minimal expectation regret. Certain choices in the measurement of regret are more appropriate if one chooses to measure variance than others, for instance, selecting expected-payoff regret allows one to measure the variance of an individual payoff (e.g., a next play payoff variance) while expected-expected regret will only allow measurement of variance across repeated plays of the full sequence.

Often, rather than measuring variance of a policy, a *high-probability bound* on the regret is proposed. High-probability bounds arise from the "probably approximately correct" learning (PAC learning) literature [149] which provides a framework for evaluating learning algorithms with traditional asymptotic measures from computational complexity theory. The language $PAC_{\epsilon,\delta}$-learnable provides that an algorithm exists such that it can learn the true value with an error rate less than ϵ with probability of at least $1 - \delta$ in the limit or within finite number of fixed iterates.

High-probability bounds introduce this PAC learning framework to regret analysis. The primary difference from the regret in expectation is that a high-probability bound considers the potential variation in return, which can be (and is, in many cases) large, a factor which is very important in medical and many financial contexts, where stakes are high. Intuitively, a high-probability bound provides a function that can be used to evaluate a bound at a given probability. This is a statement of the form $P[R(n) > b_{\text{high\_probability}}] \leq \delta$ where $R(n)$ is the regret after n plays, $b_{\text{high\_probability}}$ is the high-probability bound and δ is the probability with which we evaluate the "high" probability. The parameter δ is often picked to be a function of the form n^{-c} for a fixed constant c [9].

To provide an intuitive understanding of high-probability bounds compared to expectation regret, consider the slot-playing ϵ-first example: if we have two slot machines to pick between, and we explore 10 times (5 each) to measure our empirical estimates of each arm then exploit the best measured machine forever. In expectation the best machine will be picked, however, randomness and the small number of plays may result in a single win dominating the results and causing the estimate of the best machine to be incorrect. While the expectation of regret in this model is zero, the variance of regret is high: in many[5] plays, the regret will tend to infinity. The high-probability bound in general, will say that for $p\%$ or fewer of the repeated plays of the game, the regret will exceed b. In this specific example, it is likely the bound will be very weak for any reasonable δ, as in this particular strategy, a small number of lucky early explorations will result in suboptimal plays forever, accruing large regret.

Historically, prior to high-probability bounds being introduced for the multi-armed bandit problem, strict *bounds* were provided for certain algorithms. These bounds tend to be much weaker than the high-probability bounds found in more recent research.

Higher Moments and Risk Measures of Regret

In some contexts, other parameters of the shape of the regret distribution may prove relevant. There is very little work analyzing or applying moments higher than variance. Skewness and kurtosis may interest some applications, especially those where risk is being calculated upon the notion of regret. In general, while important, risk-aware or risk-averse bandits are conjectured by Audibert and Bubeck [11] and others as a significantly more complex problem and few major publications explore these problems. In one of the few, Carpentier and Valko [38] propose a measure of regret that is based on the extreme value of the arm distribution, producing an algorithm which minimizes tail events in a constrained environment. There is a substantial literature of risk measurement and mitigation in financial market research which the bandits literature may benefit from in the future.

Feedback Delay

In many problems, there is a simultaneity issue with training: new trials may run before the rewards for older problems have been observed. In other problems, there is a tradeoff between computation time (to recompute decision parameters for the model) and model accuracy: a longer computation time creates more feedback delay and more accurate answers, but the feedback delay effects the currency of the answers. For example, a user may be delivered an advertisement and still be reading or navigating through the process, or a computation process for that user may not have completed, when another user arrives, this delay is called feedback delay. Feedback delay is an important, often unexplored, consideration for multi-armed bandit policies with some policies being unviable in high delay environments producing an environment where fast suboptimal (in terms of theoretical regret) policies may outperform slower, theoretically-optimal policies.

Problem Difficulty

In the context of quantifying performance of algorithms in a distribution agnostic sense, a measure of “hardness” is important. Specifically, a problem where the difference between the optimal arms and non-optimal arms is small seems to have some “intrinsic difficulty” that applies across algorithms.

[5] Dependent on the gap between the two machines.

In order to capture this intrinsic difficulty, Audibert et al. [12] introduced two measures of hardness of identifying the best distribution in a set of K distributions, which they go on to prove are within a logarithmic factor of each other. The measures presented are,

$$H_1 = \sum_{i=1}^{K} \frac{1}{\Delta_i^2} \tag{2.10}$$

and

$$H_2 = \max_{i \in \{1,\ldots,K\}} \frac{i}{\Delta_i^2} \tag{2.11}$$

where Δ_i is the difference between the $i \in K$th arm's payoff and the optimal arm's payoff.

Further, there is existing work in the statistics literature aiming to quantify the difference between distributions. Many statistical tests are dependent on the idea of quantifying the distance between an assumed true distribution and the observed empirical distribution. One such measure, Kullback-Leibler (K-L) divergence, has been proposed [30, 99, 105] as an appropriate tool for the bandit problem. Broadly, Kullback-Leibler divergence is the logarithmic ratio between two distributions with regard to the first distribution. Formally,

$$D(P, Q) = \int_{-\infty}^{\infty} \ln\left(\frac{p(x)}{q(x)}\right) p(x)dx. \tag{2.12}$$

Where $p(x), q(x)$ is the probability density function of P, Q respectively. While this is not a true metric, in that it does not satisfy the triangle inequality nor is it symmetric, for our purpose, it can be used to quantify the distance between the optimal arm and all other arms. In the two arm case, this is a valid measure of the desired property. In the $k > 2$ arm case, an arm-iterated regret-weighted K-L divergence can be proposed of the form,

$$D_{arms} = \sum_{k=0}^{K} D(B_{best}, B_k) \cdot \mu_{B_{best}}. \tag{2.13}$$

.

Stationarity of the Problem

One of the strongest assumptions of many statistical models, including most variants of the multi-armed bandit problem, is that the underlying distributions and parameters are *stationary.* In many contexts, including the context studied here, this is not a reasonable assumption: the state of the world is changing around the learning process and in our context, the best arm in one time period may not be the best arm in another. Non-stationary problems are in general challenging for algorithms that make stationarity assumptions, whether explicit or implicit, as the real world performance of any such policy can continuously degrade in response to unconsidered changes in the distribution. In particular, rapid changes of the distribution and switching-type models (day, night; seasonal; or any other repeatedly changing, but unmodelled, confounding factor) have extremely poor performance on many fixed policies.

Some variants of the model, known generally as non-stationary bandit models have been proposed with drifting or step-function changing parameters. A simple solution to deal with non-stationarity is to allow the data to "decay" out of the model with a time-weighted component, however, this solution requires an accurate model of the appropriate rate of decay to be efficient. Non-stationarity is very important in the advertising context, as there is no reason to believe the underlying distributions are static. This is discussed at depth in the non-stationary bandits section (2.2.5).

Change-Point Detection For the step-function variant of the problem, change-point analysis (or step detection in signal theory) is a deeply researched statistical technique aimed at identifying the times when the underlying probability distribution of a stochastic process changes. Mellor and Shapiro [108] introduces an online Bayesian change-point detection algorithm appropriate for use in a variety of "switching environments" where the underlying arm distributions change in semi-structured ways.

Kalman Filters For the drifting or noisy version of the problem, a Kalman filter "state transition model" technique has been applied. Granmo and Berg [70] introduce a Bayesian technique that uses sibling Kalman filters to define the distributions for a Thompson sampling-type policy and demonstrate that it is empirically strong for both non-stationary and stationary variants of the problem, suggesting it could be a strong technique in cases of uncertainty. Later, we explore time series techniques for solving non-stationarity in the context of a changing world.

Ethical and Practical Constraints

In medical and research applications of multi-armed bandit models, exploratory phases may involve human subjects. These cases bring with them ethical constraints independent of the moments of the distribution. Specifically, in medical applications it may be desirable to bound the exploratory portion of a model without exception or to favour exploiting new knowledge as soon as it passes some statistical significance. In a resource exploration context, some locations may be environmentally sensitive or unexplorable for legal reasons; these might take the form of a hole in our exploration function in a contextual bandit framework or simply some bandits which are left unexplored in a pure bandit framework.

Practically, other constraints may exist. In some contexts, such as investment in research or private corporation stock, the exploration may be bounded by large minimum or insufficient maximum investments. In a financial portfolio or marketing context, due to model or stochastic error, it may never be desirable to expose a large portion of a firm's total capital to any one optimal result (or, in a contextual framework, any cluster of results). We consider the application of risk-based measures to attempt to quantify this effect and later discuss state of the art results with respect to satisficing and convex tinkering from risk management research on how to best deal with this within our marketing context.

Practical Significance

The multi-armed bandit specification discards one motivation of traditional experiment design: statistical significance or hypothesis testing[6]. Fortunately, the model maintains some sense of economic or practical significance: arms which have largely different rewards will still, to the extent possible, be identified as such in the multi-armed bandit, as a good, uncertainty-aware policy (such as UCB or Thompson sampling) must isolate the best arm from the others in a similar sense they must in a traditional experiment design in order to ensure they are appropriately balancing exploration and exploitation.

2.1.3 Formalization

To reiterate the nature of the problem we are considering, we provide the following summary formalization. A (finite-armed, stochastic) multi-armed bandit problem is a process where an agent (or forecaster) must choose repeatedly between K independent and unknown *reward distributions* (called *arms*) over a (known or unknown) time horizon H in order to maximize his total reward (or equivalently, minimize some total *regret*, compared to an oracle strategy). At each time step, t, the *strategy* or *policy* selects (*plays*) a single arm S_t and receives a reward of $x_{S_t,t}$ drawn from the ith arm distribution which the policy uses to inform further decisions. In our application, individual arms represent webpage modifications with the goal of maximizing desired user behavior (sales, time engaged, etc.).

2.2 Studied Problem Variants

For each of the variants of the model, we introduce the variant, describe the state of the research and then explore how different algorithms can be applied to this specific problem variant. In particular, we explore the common threads in terms of algorithm design, model assumptions and other factors that tie together the variants and known solutions to the problem. At the end of this section, the reader should have a strong intuition for both the state of the art and sufficient context to understand and approach new variants of the model.

[6] While achieving strict statistical significance in the test of distinguishing arms from each other is not a direct goal in the multi-armed bandit model, [87] shows that the model does not exclude traditional significance testing, especially when the goal is only to identify that a given arm performs strongly.

2.2.1 Traditional K-Armed Stochastic Bandit

The most studied variant of the model is the traditional model described earlier in this chapter, with a discrete, finite number of arms (K) for the agent to choose between at each time step t. There are a large number of techniques for solving this variant, many of which meet various notions of provably efficient.

A strategy or algorithm used to solve the multi-armed bandit problem is often called a *policy*. We discuss a set of policies, the ϵ policies [141], dependent on a parameter ϵ which determines how much exploration takes place, a set of policies called UCB-strategies (upper confidence bound strategies), based on the observation by Auer, Cesa-Bianchi, and Fischer [15] on utilizing upper confidence bounds, a variety of standalone policies and finally probability matching policies which rely on the idea of matching the probability of success with the probability of drawing that arm. Strategies like ϵ-based strategies that maintain an ongoing distinction between exploitation and exploration phases are called semi-uniform.

ϵ-greedy

The ϵ-greedy approach appears to be the most widely used simple strategy to solve the simple stochastic, i.i.d. form of the (discrete) multi-armed bandit model in practice. The strategy, in which the agent selects a random arm $0 \leq \epsilon \leq 1$ fraction of the time, and the arm with the best observed mean so far otherwise, was first presented by Watkins [156] as a solution to the equivalent one-state Markov decision process problem[7]. The choice of ϵ and strategy for estimating the mean is left to the application.

ϵ-based strategies have been well studied. Even-Dar et al. [57] show that after $O\left(\frac{K}{\alpha^2} \log \frac{K}{\delta}\right)$ random plays an α-optimal arm will be found with probability greater than $1 - \delta$, a result that applies to all major ϵ strategies.

Constant ϵ With a constant value of ϵ, a linear bound on regret can be achieved. Constant ϵ-greedy policies are necessarily suboptimal, as a constant ϵ prevents the strategy, in general, from asymptotically reaching the optimal arm [152]. That is, even after strong knowledge is acquired, the strategy will continue to behave randomly some ϵ fraction of the time.

Adaptive and ϵ-Decreasing One of the more salient variants of ϵ-greedy is the ϵ-decreasing strategy. In a stationary, finite horizon environment, it is logical to have a policy do more exploration early and more exploitation as it becomes more confident about its knowledge or as it gets closer to its horizon. This can be implemented with a variance weighted strategy or by simply decreasing ϵ according to some rule (time, observations, etc.). In known-horizon environments, ϵ-decreasing

[7] Watkins' motivation was in modelling learning processes in the real world, not for machine learning. The distinction does not appear to be important for the particular policy he devises.

policies can weight the rate exploration as a function of the remaining horizon available, though no known work has explicitly defined the correct functional form to do so.

A simple ϵ-decreasing strategy is natural and given by Vermorel and Mohri [152] which defines $\epsilon(t)$ as the value of ϵ after t plays as $\min(1, \frac{\epsilon_0}{t})$ where ϵ_0 is left as a choice to the user. A similar strategy is called GreedyMix and analyzed in Cesa-Bianchi and Fischer [40] where $\epsilon(t)$ (referred to as γ) is defined as $\min(1, \frac{5K}{d^2} \cdot \frac{\ln(t-1)}{t-1})$ where $0 < d < 1$ is a constant picked by the user[8]. GreedyMix is shown to have regret on the order of $\ln(H)^2$ for H trials for Bernoulli- and normally-distributed bandits. Selection of d is left to the reader, and performance degrades if a sub-optimal value of d is selected.

An interesting result regarding ϵ-decreasing policies is given by Auer, Cesa-Bianchi, and Fischer [15] with the simulations on a policy called ϵ_n-greedy. ϵ_n-greedy is the generalized form of ϵ greedy where the fraction of exploration is a function of the time step. At each time step t, we select the $\epsilon_t = \epsilon(t)$. By defining $\epsilon(t) \equiv \min\left\{1, \frac{cK}{d^2 t}\right\}$ and correctly selecting an unknown parameter $c > 0$ and a lower bound $0 < d < 1$ on the difference between the reward expectations of the best and second best arms, we get a policy that which has an expected regret of $O(\log H)$. Unfortunately, as noted in Auer et al. [15] this result is not of a lot of practical use, for the same reason GreedyMix lacks practicality: the selection of the constant factors c and d are dependent on the underlying distribution which we are trying to estimate and the performance degrades rapidly in the incorrectly tuned case. A theoretical, but not particularly practical, extension of this strategy is one where $\epsilon(t)$ is correctly chosen for each time step; this strategy is guaranteed to converge in an optimal number of trials in expectation.

ϵ-first

In the non-academic web optimization and testing literature, ϵ-first is used extensively, generally for 2-armed bandits and is widely known as "A/B testing"[9]. In ϵ-first, the horizon, H, must be known *a priori*. The first $\epsilon \cdot H$ plays are called the exploration phase, and the agent picks arms uniformly randomly, producing an estimate of each arm's payoff. In the remaining $(1-\epsilon) \cdot H$ plays, called the exploitation phase, the agent strictly picks the best empirically estimated arm.

An ϵ-first strategy is superior to an ϵ-greedy strategy when the horizon is fixed and stationarity

[8] Note that by letting $\epsilon_0 = \frac{5K}{d^2}$ GreedyMix is similar to the Vermorel and Mohri strategy, but not the same, as the rate of decrease is $\frac{\ln(t-1)}{t-1}$.

[9] The extensive toolsets available for automating this testing often perform "A/B testing" incorrectly. Specifically, they perform testing with repeated testing without an appropriate multiple testing significance adjustment. It is up to the user, who is generally not expected to be familiar with the statistics involved, to behave appropriately to maintain the assumptions of the model. Many researchers have addressed the multiple testing issue, for an overview of the problem see Jennison and Turnbull [82]; for an review of strategies for correcting multiple testing errors, see Hsu [78] or Westfall, Young, and Wright [159]. Indeed, the most fragile of these toolsets offer functionality to make decisions "upon reaching significance" (using an unadjusted measure of significance) which suggests a significance test after every trial: the worst form of the multiple-testing problem, resulting in a false positive rate which increases as the number of trials increases [82].

on the arms can be assumed as the estimates are expected to be better for a larger number of plays and thus fewer suboptimal plays will be likely. Compared to ϵ-greedy, ϵ-first is vulnerable to non-stationarity of the reward distribution, because all learning takes place "upfront". ϵ-first is also vulnerable to errors in estimating the *time horizon*, the number of trials remaining.

Multiple Epoch A variant of the ϵ algorithms is the multiple epoch approach. Multiple epoch approaches can be applied to many multi-armed bandit policies (e.g., Langford and Zhang [100]), but they are largely unstudied in non-ϵ approaches. They may show promise in non-stationary bandit cases where the epoch length (and data decaying) can be used to control for the maximum deviation. In the multiple epoch approach, we divide our total time horizon (known or unknown, finite or infinite) into epochs of an integer length. The respective policy is then applied *within* the epoch. For example, in the ϵ-first strategy, this eliminates some of the vulnerability to non-stationarity and horizon-unawareness by allowing learning to take place at spaced periods within the total time.

UCB1

Much of the research in regret bounds demonstrates regret that is logarithmic ("optimal") only asymptotically. Auer et al. [15] present an algorithm originating in Agrawal [5] called UCB1 which achieves expected logarithmic regret uniformly over time, for all reward distributions, with no prior knowledge of the reward distribution required. UCB1 is the first strategy we have discussed that is not a *semi-uniform* strategy, that is, it does not maintain a distinction between an exploratory phase and an exploitation phase, choosing instead to optimize how exploration happens at each individual iterate. UCB1 belongs to the general family of *upper confidence bound* (UCB) algorithms, first proposed in Lai and Robbins [99] but developed extensively in Auer et al. [15]. UCB algorithms take the form of picking the arm which maximizes a surrogate function, i.e., they pick,

$$i = \arg\max_i \mu_i + P_i, \tag{2.14}$$

where μ_i is the "average function" which estimates the mean payoff of arm i and P_i is a padding function which generally takes the form of an approximation of the uncertainty on μ_i. The primary contribution of variants of the UCB algorithms is the selection of P_i.

For convenience, let Δ_i be defined the same way as in Auer et al. [15]: $\Delta_i \equiv \mu^* - \mu_i$ where μ^* represents the mean reward expected from the optimal arm and μ_i represents the current reward expectation for arm i.

UCB1 begins by playing each arm once to create an initial estimate. Then, for each iterate t, arm i is selected to achieve the maximum value $\max_i \bar{x}_i + \sqrt{\frac{2 \ln t}{n_i}}$ where $\bar{x}_i$ is the average observed reward of arm i thus far (the empirical mean) and n_i is the number of times arm i has been played. The second term in this equation acts as an approximation for "optimism" by treating arms which have been played less as more uncertain (and thus plausibly better) than arms that have been played frequently. In UCB1's strict formulation, the bound is derived from the Chernoff-Hoeffding bound [22, 44, 76] on the right tail distributions for the estimation of Bernoulli random variables, but the confidence bound model applies equally well to any distribution where an appropriate bound can be defined.

The second term in the maximization criterion has been extended, as in the MOSS algorithm [10] (discussed in an upcoming section) to consider the remaining horizon to create an "exploratory value" that is declining in finite time or to improve the tightness of the bound on variance.

UCB1 as specified has a bounded regret at time t, for Bernoulli arms, given by the following formula, shown in the original paper,

$$8 \cdot \left[\sum_{i:\mu_i < \mu^*} \left(\frac{\ln t}{\Delta_i} \right) \right] + \left(1 + \frac{\pi^2}{3} \right) \left(\sum_{i=1}^{K} \Delta_i \right). \tag{2.15}$$

UCB2

UCB2, an iterative improvement over UCB1, reduces the constant term in the fraction of time a suboptimal arm will be selected, reducing the overall regret, at the cost of only a slightly more complicated algorithm.

In UCB2, iterates are broken into epochs of a varying size. In each epoch, arm i is selected to maximize $\bar{x}_i + \sqrt{\frac{(1+\alpha)(\ln(et/(1+\alpha)^{r_i}))}{2(1+\alpha)^{r_i}}}$ and then played exactly $\lceil (1+\alpha)^{r_i+1} - (1+\alpha)^{r_i} \rceil$ times before ending the epoch and selecting a new arm. r_i is a counter indicating how many epochs arm i has been selected in and $0 < \alpha < 1$ is a parameter that influences learning rate discussed below.

The bound of regret for UCB2 is known for times $t \geq \max_{i:\mu_i < \mu^*} \frac{1}{2\Delta_i^2}$ and is given by,

$$\sum_{i:\mu_i < \mu^*} \left(\frac{(1+\alpha)(1+4\alpha)\ln(2e\Delta_i^2 t)}{2\Delta_i} + \frac{c_\alpha}{\Delta_i} \right), \tag{2.16}$$

where e is Euler's constant and $c_\alpha = 1 + \frac{(1+\alpha)e}{\alpha^2} + (\frac{1+\alpha}{\alpha})^{(1+\alpha)}(1 + \frac{11(1+\alpha)}{5\alpha^2 \ln(1+\alpha)})$ as proven in Auer et al. [15]. The important property of c_α to notice is that $c_\alpha \to \infty$ as $\alpha \to 0$, forcing a trade-off between the selection of α to minimize the first term towards $1/(2\Delta_i^2)$ and the second term. The original paper suggests optimal results from setting α such that it is decreasing slowly in t but is not specific to the form of decrease; in practice, they also demonstrate, the choice of α does not seem to matter much as long as it is kept relatively small.

UCB-Tuned

A strict improvement over both UCB solutions can be made by tuning the upper bound parameter in UCB1's decision rule. Specifically, Auer et al. [15] further expands these solutions by replacing the second term $\sqrt{\frac{2\ln t}{n_i}}$ with the tuned term $\sqrt{\frac{\ln t}{n_i}\min(\frac{1}{4}, V_i(n_i))}$ where V_i is an estimate of the upper bound of the variance of arm i given by, for example,

$$V_i(n_i) \equiv \left(\frac{1}{n_i}\sum_{\tau=1}^{n_i} X_{i,\tau}^2\right) - \bar{X}_{i,n_i}^2 + \sqrt{\frac{2\ln t}{n_i}}, \tag{2.17}$$

where n_i is the number of times arm i has been played out of t total plays. UCB-Tuned empirically outperforms UCB1 and UCB2 in terms of frequency of picking the best arm. Further, Auer et al. [15] indicate that UCB-Tuned is “not very” sensitive to the variance of the arms. Simple experimentation shows that UCB-Tuned as defined above outperforms the earlier UCBs significantly in all tested worlds.

MOSS

MOSS [11], or the Minimax Optimal Strategy in the Stochastic case, produces a variant of UCB1 that is presented in a generalized context, such that it can apply to all known bandit variants or subproblems. In MOSS, the $\ln t$ component of the padding function in UCB1 for arm i is replaced with $\ln\frac{H}{Kn_i}$ where n_i is the number of times arm i has been played, H is the total number of iterates to be played (the horizon, at the beginning) and K is the number of arms available in a stochastic (non-adversarial) bandit problem. The work of Audibert and Bubeck [11] shows that expected regret for MOSS is bounded from above, by,

$$\mathbb{E}R \leq 25\sqrt{HK} \leq \frac{23K}{\Delta}\log\left(\max\left(\frac{140H\Delta^2}{K}, 10^4\right)\right), \tag{2.18}$$

where $\Delta = \min_{i:\Delta_i>0}\Delta_i$, the smallest gap between the optimal arm and the second best arm. Note that this calculation of regret applies continuously in the stochastic case, but we will see later in the adversarial discussion that it is marginally complicated in that environment due to non-unicity of the optimal arm.

Bayes-UCB

Combined with KL-UCB (covered in the next section), Bayes-UCB [88] — an explicitly Bayesian variant of UCB — represents the current state of the art of UCB algorithms. It is an asymptotically efficient advanced algorithm with promising empirical results. In the Bayesian approach to the multi-armed bandit problem, each arm is represented as an estimate of a distribution that is updated in the traditional Bayesian fashion. Kaufmann et al. [88] show that this Bayesian-derived UCB has a cumulative regret that empirically outperforms the strongest of the original UCB algorithms by a substantial margin in a handful of selected problems while having the advantage of being distribution agnostic and showing the early-iterate flexibility of a Bayesian approach to knowledge acquisition. A computational complexity challenge is acknowledged but not explored in depth.

Bayes-UCB is similar to the *probability matching* strategies to be discussed later: quantiles of a distribution are estimated to increasingly tight bounds and the probability of a given arm "being the best" is used to determine the next step. To perform Bayes-UCB, the algorithm requires a prior on the arms, Π^0 and a function to compute the quantiles of the expected distributions, $Q(\alpha, \rho)$ such that $P_\rho(X \leq Q(\alpha, \rho)) = \alpha$. At each time step t, Bayes-UCB draws the arm i to maximize the quantile selected as follows. It picks

$$i = \arg\max_i q_i(t) = Q(1 - \frac{1}{t}, \lambda_i^{t-1}), \tag{2.19}$$

where Q meets the property described above and $\lambda_i^{(t-1)}$ is the estimated posterior distribution of arm i at the previous time step. This is then updated according to the Bayesian updating rule and used as the prior for the next iteration.

In a theoretical analysis, Kaufmann et al. [88] show that Bayes-UCB achieves asymptotic optimality and a non-asymptotic finite-time regret in $O(H)$.

It is interesting to note that by treating the quantile function and underlying model appropriately, Bayes-UCB can, in theory, represent any distribution and most subproblems of the multi-armed bandit. As a simple but valuable example, by representing the underlying model as a Bayesian regression, one can include contextual information in the bandit process.

KL-UCB

KL-UCB [105] presents a modern approach to UCB for the standard stochastic bandits problem where the padding function is derived from the so-called Kullback-Leibler (K-L) divergence. KL-UCB demonstrates regret that improves the regret bounds from earlier UCB algorithms by considering the distance between the estimated distributions of each arm as a factor in the padding function. Specifically, define the Kullback-Leibler divergence[62, 98] (for Bernoulli distribution arms) as,

$$d(p, q) = p \log \frac{p}{q} + (1 - p) \log \frac{1 - p}{1 - q}, \tag{2.20}$$

with convention of $0 \log 0 = 0$, $0 \log \frac{0}{0} = 0$, and $x \log \frac{x}{0} = +\infty$ for $x > 0$. The Kullback-Leibler divergence $d(p, q)$ provides a probability-weighted measure of the difference between two distributions which does not rely on collapsing the distribution to a midpoint (e.g., expectation).

To pick an arm in each iteration of KL-UCB, we maximize

$$i = \arg\max_i n_i \cdot d(\mu_i, M) \leq \log t + c \log \log t \tag{2.21}$$

where M is picked from the set of all possible reward distributions. The K-L divergence of $d(x, M)$ is strictly convex and increasing in $[x, 1)$ [62] making this equation tractable.

POKER and price of knowledge

A non-UCB algorithm, POKER [152] or Price of Knowledge and Estimated Reward is a generalizable economic analysis inspired approach to the problem.

The intuition behind POKER is to assign a “value” of the information (the “exploration bonus”) gained while pulling a given arm. This value is estimated for each arm, and then the arm with the highest expected payoff *plus* expected value of information is played. Value of information is defined to maximize expected outcome over the horizon. To gain an intuition, first assume an oracle provides the best arm as arm i^* with payoff μ^*, that we have an estimate for each arm's payoff $\hat{\mu}_i$ and that we have an estimated best arm $\hat{i}$ with estimated payoff $\hat{\mu}^*$. Define the magnitude of the expected improvement as $\delta = E[\mu^* - \hat{\mu}^*]$, then the probability of an improvement for a given arm is $P[\mu_i - \hat{\mu}^* \geq \delta]$.

When there are $(H - t)$ plays left, any new knowledge found in this iterate can be exploited $(H - t)$ times. This means the expected improvement has a (non-discounted) value of $\delta \cdot (H - t)$.

A problem arises in computing δ, as if i^* and μ^* were known, there would be no need to explore. Instead, the ordered estimate of the means are used. Imagine, an ordered list of the mean rewards as $\hat{\mu}_{i_1} \geq \cdots \geq \hat{\mu}_{i_q}$. Vermorel and Mohri [152] choose, based on primarily empirical results, to approximate δ proportional to the gap between $\hat{\mu}_{i_1}$ and the $\hat{\mu}_{i_{\sqrt{K}}}$ arm. Specifically, they set $\delta = \frac{\hat{\mu}_{i_1} - \hat{\mu}_{i_{\sqrt{K}}}}{\sqrt{K}}$. That is, if there are K arms, the difference between the best and the $\sqrt{K}$th best current estimate is proportional to the plausible gain.

In the limit (as the number of arms approaches infinity), this approximation strategy ensures bias and variance minimization.

Additionally, one can observe that the whole probability $P[\mu_i - \hat{\mu}^* \geq \delta] = P[\mu_i \geq \hat{\mu}^* + \delta]$ is approximated (or identical, in the event of normally distributed means[10]) by the cumulative probability of the reward *higher* than the best empirically expected reward plus expected improvement $\hat{\mu}^* + \delta$,

$$P[\mu_i \geq \hat{\mu}^* + \delta] = \int_{\hat{\mu}^*+\delta}^{\infty} \mathrm{N}\left(x, \hat{\mu}_i, \frac{\hat{\sigma}_i}{\sqrt{n_i}}\right) dx, \tag{2.22}$$

where $N(x, \mu, \sigma)$ represents the normal distribution and n_i means the number of times arm i has been played and mean μ_i and variance σ_i take on their usual meaning.

This gives us sufficient information to define a decision criterion. Select the arm which maximizes the expected sum of total rewards over the horizon H. Formally, at each time step t, select arm i to play:

$$\arg\max_i \mu_i + \delta(H - t)P[\mu_i \geq \hat{\mu}^* + \delta] \tag{2.23}$$

[10] This is true in the limit by the central limit theorem, but as there may be a small number of arms and trials, it may be a poor approximation in some environments.

Note that $\delta(H-t)$ is the total expected gain over the remaining horizon. By multiplying by the probability this arm will actually exceed the best known arm, we achieve a sensible expectation to maximize. This value could be easily time-discounted by introducing a sum of discounted payoffs if the time horizon was measured at a scale where time-discounting were of value.

POKER uses knowledge of the length of the horizon or number of plays that remain, $(H-t)$, as a parameter that effectively determines how to weight exploration and exploitation. The authors make the claim that requiring the horizon explicitly is a more intuitive parameter than the parameters associated with many other algorithms. Additionally, the parameter can be set to a fixed value to simply use it to balance exploration and exploitation in the case horizon is unknown or infinite.

Vermorel and Mohri [152] introduce the POKER policy and use the term *zero-regret strategy* to describe it. In their context, *zero-regret* means guaranteed to converge on the optimal strategy, eventually: that is, a strategy which has average per-play regret tending to zero for any problem which has a horizon tending to infinity. The term *zero-regret* will not be used in the rest of our discussion, preferring instead “guaranteed to converge to zero.”

The authors compare POKER to ϵ strategies, Exp3 (discussed in a future section) and others on a real world redundant retrieval[11] routing problem and find that POKER outperforms ϵ strategies by a factor of approximately 3. As of this writing, there has been no known finite time analysis of regret for POKER.

2.2.2 K-Armed vs. Infinite-Armed Bandits

Expanding on the traditional model is a variant which treats the number of arms as an infinite or continuous range with some functional form defining the relationship *or* a sufficient mechanism for discretizing the infinite space. This variant allows for substantial variation in problem difficulty by varying how much the agent knows about the arm relationship.

As an example of the infinite-armed bandit case, consider the case of picking a color for a purchase button to optimize clicks. Each user that views the purchase button is (possibly) influenced by its color, and color is a (theoretically) continuous function. As it would be impossible to sample all the colors, in the infinite-armed case, for the analysis of an infinite-armed bandit to be tractable, there must exist an underlying well-behaved function defining the relationship between arms (colors) and the payoff function (clicks).

While the infinite-armed case is very interesting from a web optimization and online advertising perspective, it is not a major source of motivation for this work choosing to prefer a focus on discrete changes. As such, only a quick overview of the state of the art is provided here for completeness. One of the earliest works in this space was Agrawal [5].

[11] The problem is to identify the fastest source for a content delivery network with numerous redundant sources of requested data.

Recent work has studied such variants of the infinite armed problem as high-dimensional spaces [147, 148], non-smooth spaces [47] and multiple-objectives [151], although work on theoretical analysis of the existing algorithms is still ongoing. An approach to define a computationally and mathematically feasible regret in a generalized infinite-armed case is presented in Carpentier and Valko [39].

Bandit Algorithm for Smooth Trees (BAST)

[48] present an analysis of the popular UCT (upper confidence bound for trees) algorithm which combines Monte Carlo tree-based techniques from artificial intelligence[12] with the UCB1 algorithm discussed prior. UCT is popular in planning problems for game-playing artificial intelligence, but not itself appropriate for infinite-armed bandit problems as it applies in a space where the number of potential decisions extremely large[13], not infinite.

The Bandit Algorithm for Smooth Trees introduces a mechanism related to continuity in to the tree approach, choosing to, for example, represent a continuous space as a tree with repeated branches dividing that space. The fundamental assumption is that leaf nodes in the tree can be expected to have similar values in the payoff space. Coquelin and Munos [48] represent this assumption rigorously requiring that for any level in the tree, there exists a value $\delta_d > 0$ called the smoothness coefficient such that for at least one (optimal) node i in that level d the gap between the optimal leaf node (μ^*) and all other leaf nodes is bounded by δ_d. Formally,

$$\mu^* - \mu_j \leq \delta_d \quad \forall j \in \text{leaf(i)} \tag{2.24}$$

Assumptions of this style are the core tool with which infinite-armed bandits are generally represented. The tree is then generally assumed to take a coarse-to-fine hierarchical representation of the space down to some maximum tree height.

BAST performs selection at each level of the tree using a variant of the UCB algorithms which takes in to consideration the estimates of nearby nodes. Specifically, for a given smoothness coefficient δ_d for each level on the tree, a selection mechanism for any non-leaf node is given as maximizing

$$B_{i,n_i} = \min\left\{(\max B_{j,n_j}), X_{i,n_i} + \delta_d + c_{n_i}\right\} \tag{2.25}$$

And for any leaf node as the simple UCB criteria,

[12] A good survey of related algorithms for tree search is given by Browne et al. [29].

[13] Many games are well modelled as a tree structure where each decision reveals a new set of decisions until an end state (win or lose) is reached. These trees can grow very rapidly, but are not generally continuous. A number of techniques are proposed for dealing with them [120] and these techniques frequently overlap with some techniques proposed for infinite- or continuum-armed bandits.

$$B_{i,n_i} = X_{i,n_i} + c_{n_i} \tag{2.26}$$

Where c_{n_i} takes the role of the padding function from UCB and is defined as

$$c_n = \sqrt{\frac{\log(2Nn(n+1)\beta^{-1}}{2n}} \tag{2.27}$$

A technique described in [49] and [64] is also explored in [48] which allows the online production of the tree with or without the assumption of a maximum tree height. In the iterative growing variant of BAST, the algorithm starts with only the root node, then at each iterate, selects a leaf node via its selection mechanism described above and expands the node in to two new child nodes, immediately choosing to play each child node once. This iterative technique still requires $O(n)$ memory to maintain the tree, but results in only optimal branches being explored in depth, a desirable property.

Hierarchical Optimistic Optimization (HOO)

Similar to BAST, Hierarchical Optimistic Optimization [34, 35, 92] attempts to build an estimate of the functional form f by treating the problem as a hierarchical (coarse-to-fine) tree, with a particular focus on only maintaining a high-precision estimate of f near its maxima. HOO builds and maintains a binary tree where each level is an increasingly precise subspace of the total space of arms, X. Each node in the tree tracks its interval range (subspace), how many times the node has been traversed and the empirical estimate of the reward, which it uses to compute an optimistic upper bound estimate, B, for this leaf's reward in a similar fashion to the UCB algorithms. At each time step, the algorithm traverses the tree, picking the highest B node at each junction until it reaches a leaf. At a leaf, it splits the node and creates a new point which is evaluated and the upper bound estimate is updated up the tree accordingly.

This makes an assumption about the shape of f, but not one as strong as the BAST algorithm did. Rather than requiring continuity in a strongly defined sense as in the δ_d existence assumption before, HOO requires only that a dissimilarity function exists which puts a lower bound on the mean-payoff function over the arbitrary space the arms exist in.

In HOO, the selection strategy for each node requires a measure $\mu_{d,i}$ which represents the empirical mean of the payoff from each time the node been traversed and $N_{d,i}$, the number of times that node has been traversed. We use d to denote depth in the tree, as in the BAST exposition, and i to denote the specific node. Following again in the UCB strategy, the corresponding upper confidence bound criterion can be given as

$$\mu_{d,i} + \sqrt{\frac{2\ln n}{N_{d,i}}} + \upsilon_1\rho^d \tag{2.28}$$

where $0 < \rho < 1$ and $\upsilon_1 > 0$ are parameters selected by the implementer

Since their work, there has been work on an algorithm called Open-Loop Optimistic Planning (OLOP) [32, 150] and a combination work called hierarchical OLOP [157] which combines the methods. These are not explored in more detail here, as infinite-armed bandits do not arise significantly in the remainder of our work.

2.2.3 Adversarial Bandits

One of the strongest generalizations of the k-armed bandit problem is the adversarial bandits problem. In this problem, rather than rewards being picked from an *a priori* fixed distribution, rewards are selected, in the worst case per-play, by an adversary. The problem is transformed into an iterated three step process; in step 1, the adversary picks the reward distributions (generally with full availability of the list of prior choices, though the constraints on the distribution are discussed); in step 2, the agent picks an arm; in step 3, the rewards are assigned. This is a strong generalization because it removes the distribution dependence on the arms (and as such, stationarity and other distribution-dependent assumptions); an algorithm that satisfies the adversarial bandits problem will satisfy more specific[14] bandits problems, albeit, often sub-optimally.

As adversarial bandits are such a strong generalization, Audibert and Bubeck [11] provide a taxonomy of bandit problems that builds from the constraints on the adversary's selection process. Fundamentally, allowing the distribution to vary in each time, they let n represent the number of possible distributions available. They then provide five distinctions. (1) The purely *deterministic bandit problem*, where rewards are characterized as a matrix of nK rewards, where K represents the number of arms and n the number of time steps. In each time step, a single deterministic reward is set (fixed *a priori*) for each arm. (2) The *stochastic bandit problem* – the variant discussed in the majority of this work – in this taxonomy is characterized by a single distribution for each arm, stationary in time, independent and bounded on some range, say, $x_i \in [0, 1]$. (3) The *fully oblivious adversarial bandit problem*, in which there are n distributions for each arm, independent of each other (both through time and across arms) and independent of the actor's decisions, corresponding to changes selected by the adversary across time. (4) The *oblivious adversarial bandit problem*, in which the only constraint is that the distributions are selected independent of the actor's decisions. Finally, (5) the adversarial bandit, in their work referred to as the *non-oblivious bandit problem*, where the reward distributions can be chosen as a function of the actor's past decisions. For reference, we have presented the distinctions and how they map to the bandit taxonomy we have provided in Table 2.1.

In the majority of this work, we focus explicitly on the stochastic variants of the multi-armed bandit problem, choosing a lens by which deterministic or even simply non-oblivious bandits are not known to be deterministic or non-oblivious by the agent ahead of time. Our lens models the

[14] Especially, contextual bandits.

various forms of oblivious bandits as considerations to the stochastic nature of the problem, for example, treating contextual covariates and non-stationarity as a form of statistical misspecification, even when sometimes that misspecification will be impossible to resolve (as in the case of *Knightian uncertainty* [93], where the correct reward model has some immeasurable and incalculable component). This differs from the lens in Audibert and Bubeck [11], providing a prospective which applies more closely to the application area of interest in this work (one in which the true underlying model almost certainly consists of unknown or unknowable covariates, but is also partially approximated by variables we can observe), but comes at the cost of not generalizing to the pure non-oblivious adversarial problems.

In our slot machine example, this is a world where the casino selects which machines will have which payoff distributions after each play, in the worst case, in effort to *minimize* your reward. For example, if a player were consistently picking the same machine, the casino would move the worst payoff distribution to that machine. Indeed, the title of Auer and Cesa-Bianchi [14] paper introducing the problem, “Gambling in a rigged casino: the adversarial multi-armed bandit problem” captures this stylized example well. In the general sense, the adversarial case makes no assumption on the underlying payoff distribution, and the adversary does not need to necessarily be a true minimizer.

The major caveat of adversarial bandits, is that our definition of “performance” needs to be relaxed for any measures to be meaningful. Specifically, a strong performing algorithm must be defined using a measure of regret that compares our decisions solely to a fixed machine over time, that is, a strong adversarial bandit can still achieve logarithmic regret, but only if the “best arm” is defined at time $t = 0$ and does not vary across trials. To rephrase, that means that of our definitions of regret given earlier in this chapter, only the “weak-regret” notions can be meaningful within an adversarial context.

In the online advertising context in this work, the general form of adversarial bandits have little direct application other than as motivation for contextual (and distribution-free) bandits, as such only a cursory overview of their evolution follows. Further work may include competition with other advertisers as an adversarial case, though the specific implementation of the model is uncertain.

The majority of the efficient solutions to adversarial problems are variants of the Exp3 algorithm presented in Auer, Cesa-Bianchi, Freund, and Schapire [16] for the general, no statistical assumptions adversarial bandits case. Beygelzimer, Langford, Li, Reyzin, and Schapire [26] extend the work of Auer et al. [16] and McMahan and Streeter [107] to transform Exp4 to produce a high-probability bounded version called Exp4.P.

Hedge and Exp3

Auer and Cesa-Bianchi [14] present the first look at the adversarial bandit problem and include an algorithm with high-probability bounded regret called Exp3: the **exp**onential-weight algorithm for **exp**loration and **exp**loitation based on an algorithm called Hedge for the full information problem. Exp3 [16] presents a readily understandable, simple algorithm for adversarial bandits. Given a "pure exploration" parameter $\epsilon \in [0,1]$, which measures the fraction of time the algorithm selects a purely random decision, the algorithm then spends $(1-\epsilon)$ of the time doing a weighted exploration/exploitation based on the estimated actual reward.

The estimation process for Exp3 is an exponentially updating probability weighted sample. The arm weight is updated immediately after pulling a given arm and being delivered the reward ρ_i with the formula

$$w_{i,t} = w_{i,t-1} \cdot e^{\epsilon \cdot \frac{\rho_i}{p_{i,t} \cdot K}}, \tag{2.29}$$

where $w_{i,t}$ is the arm i specific weight at time t and p is our selection criteria. The probability of each specific arm to play in each iteration is selected according to p, which considers the arm weighting and ϵ semi-uniformity, namely,

$$p_{i,t} = (1-\epsilon)\frac{w_{i,t}}{\sum_{j=1}^{K} w_{j,t}} + \epsilon \cdot \frac{1}{K}. \tag{2.30}$$

In some sense, Exp3 combines the semi-uniformity in the parameterization of ϵ strategies with the "probability of best" weighted exploration/exploitation strategies of probability matching methods.

A computationally efficient version of Exp3 called Exp3.S is presented in Cesa-Bianchi and Lugosi [41].

Exp4

Exp3 does not include any concept of contextual variables or "expert advice". Auer et al. [16] develop an extension of Exp3, called Exp4 (Exp3 with **exp**ert advice). Exp4 is identical to Exp3, except the probability of play is selected with the addition of a set of N context vectors ξ per time and the weight function is similarly replaced to consider the context vectors. One should note that the weights w are now computed per context vector, where a context vector can be viewed as an "expert" advising of a selection coefficient for each arm; we now use j to indicate the index of the expert and continue to use i to indicate the index of the arm, for clarity,

$$w_{j,t} = w_{j,t-1} \cdot e^{\epsilon \cdot \frac{\rho_j \cdot \xi_{j,t}}{p_{j,t} \cdot K}}. \tag{2.31}$$

For the selection probability p, interpret $\xi_{j,t}(i)$ as the advice coefficient expert j gives at time t about arm i,

$$p_{i,t} = (1-\epsilon)\sum_{i=1}^{N} \frac{w_{j,t}\xi_{j,t}(j)}{\sum_{k=1}^{K} w_{k,t}} + \epsilon \cdot \frac{1}{K}. \tag{2.32}$$

Note that k represents an iterator over all arms in the second term. With a minor abuse of notation, this is equivalent to Exp3 where we update our weight vector with the context ξ, reward ρ, and selection probability p according to $\xi \cdot \rho / p$ for the arm played at each time step except that the weight vector is now the summed contextual weight vector.

Exp4.P Exp4.P is a variant of the Exp4 algorithm presented in Beygelzimer, Langford, Li, Reyzin, and Schapire [26] with bounded[15] regret in the high-probability case of $\tilde{O}(\sqrt{KH \log N})$. The bound does not hold in high-probability in the original Exp4 presentation as the variance of importance-weighted numerator term is too high [26]. Exp4.P modifies Exp4 such that the bound holds with high-probability. The change in Exp4.P is only in how the weight vector is updated at time t. Rather than using Equation (2.31), Exp4.P uses an updating function,

$$w_{j,t} = w_{j,t-1} \cdot e^{\frac{\epsilon}{2} \cdot \left(\rho_j \cdot \xi_{j,t} + \hat{v}_{j,t} \sqrt{\ln{(N/\delta)}/KH}\right)}, \tag{2.33}$$

where $\delta > 0$ is a parameter that defines the desired probability bound of the regret $(1 - \delta)$ and $v_{j,t}$ is defined as

$$v_{j,t} = \sum_{1,\ldots,K} \frac{\xi_{i,t}(j)}{p_{i,t}}. \tag{2.34}$$

This modification allows Beygelzimer, Langford, Li, Reyzin, and Schapire [26] to bound regret of the new algorithm, Exp4.P, with probability of at least $1 - \delta$ to $-6\sqrt{KH \log(N/\delta)}$.

Stochastic and Adversarial Optimal (SAO)

Bubeck and Slivkins [33] introduce a testing technique that is capable of handling both the stochastic (non-adversarial) problem and the adversarial problem with near-optimal regret results. Stochastic problems generally use a different definition of regret than adversarial problems, so the analysis provided in this work takes place in two parts assuming the model is *either* stochastic *or* adversarial showing asymptotically regret of $O(\text{polylog}(n))$ in the stochastic case[16] and the $O(\sqrt{n})$ pseudo-regret from Exp3 in the adversarial case.

SAO proceeds in three phases, making it a semi-uniform strategy: exploration, exploitation and the adversarial phase. The exploration and exploitation phases are largely as expected, interleaved to operate pairwise (arm 1 vs. arm 2) and rule out "suboptimal" arms as it progresses. For the remainder of this exposition, assume there are only two arms and arm 1 is strictly superior to arm 2. Further, let $C \in \Omega(\log n)$ be an arbitrary parameter which enforces consistency, selected specifically for the application area, for example $C = 12\log(n)$, let $\tilde{H}_{i,t}$ be the average observed

[15] The notation $\tilde{O}(n)$ is read "soft-O of n" and is equivalent to $O(n \log^k n)$, i.e., the big-O notation where logarithmic factors are ignored.

[16] The notation $O(\text{polylog}(n))$ means $O((\log n)^k)$ for some k. This is similar to the use of $\tilde{O}$ to indicate the insignificance logarithmic terms often bring to the analysis of algorithms.

reward for arm i, t represent time (number of iterates so far) and τ_* represent the point we switch from exploration to exploitation.

We start in a state of exploration, where we pick an arm with equal probability for a minimum of C^2 rounds and until we find a "sufficiently superior" arm according to the condition,

$$|\tilde{H}_{1,t} - \tilde{H}_{2,t}| < \frac{24C}{\sqrt{t}}. \tag{2.35}$$

During the exploitation phase, the arms are drawn according to the probabilities $p_t(2) = \frac{\tau_*}{2t}$ and $p_t(1) = 1 - p_t(2)$, that is, the probability of drawing the suboptimal arm is decreasing asymptotically in time. A set of conditions is checked to see if the observed rewards still fit within the expected stochastic model. The conditions checked are referred to as *consistency conditions* and are as follows.

The first consistency condition, which checks if the observed rewards in exploitation are congruent with the findings of the exploration phase, that is, whether the rewards are bounded in a range consistent with our observation that arm 1 is better than arm 2 by approximately the observed amount. Concretely, the first consistency condition is

$$\frac{8C}{\sqrt{\tau_*}} \leq \tilde{H}_{1,t} - \tilde{H}_{2,t} \leq \frac{40C}{\sqrt{\tau_*}}. \tag{2.36}$$

.

The second consistency condition, which checks that arm i's estimate is still within bounds of the expected estimate, consistent with the fact that during exploitation the suboptimal arm is drawn with low probability. Consider $\hat{H}_{i,t}$ to be the expected reward from arm i at time t given that the world is stochastic and the arm can be appropriately modelled, so that $\tilde{H}_{i,t} - \hat{H}_{i,t}$ represents the difference in the expected reward and the observed reward. Concretely, the second consistency conditions are,

$$|\tilde{H}_{1,t} - \hat{H}_{1,t}| \leq \frac{6C}{\sqrt{t}}, \tag{2.37}$$

$$|\tilde{H}_{2,t} - \hat{H}_{2,t}| \leq \frac{6C}{\sqrt{\tau_*}}. \tag{2.38}$$

All the *magic numbers* in these conditions are derived from the high-probability Chernoff bounds for the stochastic case. The different denominators on the right hand side of the equation account for the low probability of drawing the inferior arm (arm 2) during exploitation.

In the event any of the consistency conditions fail, we assume the model is non-stochastic and switch from the explore-exploit algorithm to that of Exp3. The work explores and proves properties of the conditions. Selection of the consistency parameters is important, as they would allow a carefully crafted adversary to maintain the conditions. Such conditions cannot allow the adversary to create a high level of regret for the application yet must hold in high probability in the non-adversarial case.

This algorithm as described combines the asymptotic regret bounds of both UCB1 and Exp3 in a near-optimal (asymptotic) fashion for both stochastic and the most general form of adversarial bandits. There is no analysis of the finite time regret.

Table 2.1: A hierarchy of bandit problems, categorized by the adversarial bandits generalization in Audibert and Bubeck [11].

Fully Oblivious

Each arm's distribution may change through time, but are fully independent of each other both across time and across arms, e.g., each arm changes to a randomly drawn new payoff every iterate.

– Deterministic

Arm payoffs are not random variables and are fully selected ahead of time, independently of both each other, the player's decisions and time.

– Stochastic

Arm distributions are random variables drawn from a distribution for each arm which is fixed in time and independent of each other. In an equivalent alternative presentation, arm distributions are drawn from a single K-dimensional distribution which is i.i.d.

Oblivious

Each arm's distribution may change through time and may be correlated with itself, other arms, or observed or unobserved covarying variables, but not with the player's past decisions. This appears as an "adversary" who is not informed of the player's choices.

– Nonstationary Stochastic

Arm distributions may have their parameters drifting in time or have changepoints in time at which they suddenly change in parameters.

– Contextual Stochastic

Arm distributions have observed or unobserved covariates which partially determine their payoff. E.g., the web optimization environment described where a user may have demographic or individual variables which make one arm better or worse for that specific context than another or the case of a casino with slot machines half red and half blue, where red arms pay at a different rate than blue arms (but the player does not know that ex-ante). Covariates can be received from the environment (in the described case of an individual observer having them) or from the arms themselves (as in the colored machines case).

Non-Oblivious

Arm distributions can be chosen as any function of the actor's past decisions, including one that meets the intuitive definition of "adversary," e.g., the arms may be "designed to trick" the actor, a long string of wins followed by a large loss.

2.2.4 Contextual Bandits

The simple k-armed bandit problem performs sub-optimally by its design in the advertising context. In general, the contextual bandits framework is more applicable than the non-contextual variants of the problem, as it is rare that no context is available [100][17]. Specifically, the simplest form of the model selects from k advertisements then discovers the payout associated with that particular play.

The contextual bandit setting has taken many names including bandits with context, bandit problems with covariates [121, 131, 160], generalized linear bandits, associative bandits [139] and bandit problems with expert advice [15]. The contextual bandit problem is closely related to work in machine learning on supervised learning and reinforcement learning; indeed, some authors [53] have referred to it as "the half-way point" between those fields because of the ability to construct algorithms of a reinforcing nature with convergence guarantees while considering relatively general models.

In this work, we further divide the context into both the advertisement (arm-context) and the user or world-state (weather) being selected for (world-context), where arm-context can be used to learn shared properties across arms in the same way as in infinite armed bandits, while world-context interacts with arm context and is declared on a per step basis. In a more complicated hierarchical model, we have even more information - we know at each step all the factors of the preceding step (hierarchy and world context). This allows a much more rich learning process where the hidden vector of contextual variables can be used to guide learning, even if they are incomplete.

Broadly, the expected reward can be approximated by a model of the form

$$Y_i = \alpha + \beta A_i + \gamma W_t + \xi A_i W_t + \epsilon \tag{2.39}$$

where Y_i indicates the expected payoff of a given arm conditional on the context, β indicates the coefficient vector as a function of the arm context, and γ a coefficient vector of the world context. In the web search context, the world context vector might be the words included in the search query, in which case we would expect our agent, in the limit, to learn a model that suggests ideal advertisements related to the query for any given search.

A slightly more general form of contextual or side-information bandits is referred to as associative reinforcement learning [139] in some statistical and machine learning literature.

Early research for the contextual bandit problem includes Wang et al. [155] and Pandey et al. [118] and makes additional assumptions about the player's knowledge of the distribution or relationship between arms. One of the first practical algorithms to be discussed in the context of horizon-unaware side-information bandits was Epoch-Greedy presented by Langford and Zhang

[17] While it is rare that no context is available, it is **not** rare that the value of the context is entirely unknown – in the stylized example of a slot machine, the arms may be different colors, whether that is a determining factor in the payoff probabilities or not may be *a priori* completely uncertain.

[100]. One of the most salient works in this space is that of Dudik et al. [53] which brings contextual learning to a practical light by producing an online learning algorithm with a running time in *polylog*(N) and regret that is additive in feedback delay. Additionally, the work of Chu, Li, Reyzin, and Schapire [46] produces an analysis of an intuitive linear model-type upper confidence bound solution called LinUCB [1, 50, 128] derived from the UCB solutions for non-contextual bandits which provides good real world performance.

Importantly, Exp4 [16] makes no statistical assumptions about the state of the world or arms and therefore can be applied to the contextual problem, however, the majority research thus far derived from Exp-type algorithms has been focused on the adversarial problem discussed prior. One exception is the application of Exp4.P to the strictly contextual problem found in Beygelzimer et al. [26].

Returning briefly to the slot machine example, contextual bandits model the situation where the machines have properties (arm-context) we believe may effect their payoff: perhaps some machines are red, some machines are blue (categorical context); perhaps machines closer to the front casino seem to pay better (linear, continuous context). This could also represent the situation where payoffs vary by day of week, time of day or another (world-context): perhaps slot machines in general are set by the casino to pay more on weekdays than on weekends, in effort to increase the number of plays during the week.

LinUCB

LinUCB is a strong, intuitive polynomial time approach to the contextual bandits problem. Largely, LinUCB builds on the upper confidence bound work of the non-contextual bandits solution by synthesizing concepts captured by the associative reinforcement learning algorithm LinRel[13]. LinUCB introduces a feature vector to the UCB estimate which is maintained with a ridge regression.

In general form, LinUCB observes a set of d features per arm (i) $x_{t,i}$ at each time step (t) and then selects an arm by a maximization of the regularized upper confidence bound estimate,

$$i = \arg\max_i \theta_t^{'} x_{t,i} + \alpha\sqrt{x_{t,i}^{'} A^{-1} x_{t,i}}, \tag{2.40}$$

where α is a positive regularization parameter and θ_t is the coefficient estimate for the arm's features. ($\theta_t = A^{-1}b$ where A and b are maintained via the ridge regression updating process after observing the reward[18])

LinUCB achieves regret in polylog(H). Specifically, the regret bound shown by Chu, Li, Reyzin, and Schapire [46] is $O(\sqrt{Td\ln^3(KH\ln(H)/\delta)})$ for d dimensional feature vectors up to a probability of $1-\delta$. The algorithm's sensitivity to non-stationarity and feedback delay has not yet been investigated in depth though it may perform adequately on feedback delayed situations as the

[18] Recall that in the regression minimization problem, $\hat{\theta} = (X^{'}X)^{-1}X^{'}y$ and let $A = X^{'}X$ and $b = X^{'}y$ where y is the observed reward

effect (or "pull") of each additional observation should decrease in increasing trials.

We present a technique in Chapter 3 which extends LinUCB to the Thompson Sampling paradigm for improved performance in handling variance in early pulls and in generalizing the distribution underlying the regression coefficients.

CoFineUCB

An interesting approach to the contextual bandits problem is to treat the exploratory contexts as a hierarchy. When this works, it could achieve logarithmic treatment of the features by treating them as a tree. Generalizing LinUCB, CoFineUCB approaches the estimation in a coarse-to-fine approach that allows increasing accuracy by drilling into a particular variable subspace. CoFineUCB extends LinUCB to fit a model strictly within a selected "coarse" subspace with a regularization parameter for the "fine" regression. The intuition provided is one of user's preferences – if preferences can be embedded in a coarse-fine hierarchy (e.g., movies (coarse), action movies (fine); or ice cream (coarse), vanilla ice cream (fine)), then an initial model on the coarse levels can be supplemented by a stronger model on only those within the class to predict the fine levels.

In practice, CoFineUCB has been used in a recommender system context and shows good performance on experimental measures of regret when the coarse subspace accurately reduces the prediction variation for most users.

Our technique presented in Chapter 3 for creating LinTS is applicable to the CoFineUCB results to produce CoFineTS, an information-theoretically improved variant of the coarse-fine strategy.

Banditron and NeuralBandit

The contextual bandits solutions explored so far require the effect of context be linear in the parameters within the interval being estimated. While some flexibility exists in terms of acceptable error rate and interval estimation, the linear regression techniques are all subject to similar constraints. Banditron [86] and NeuralBandit [7] are recent algorithms for the non-linear contextual bandit which utilize the insights from the multi-layer perceptron [126]. At a high-level, these algorithms replace the (generally back-propagation based) updating process in the perceptron algorithm, with a partial information technique using only the bandit feedback. The specific update process differs in each algorithm.

As of the date of this work, neural network-based techniques lack much theoretical analysis and show significantly suboptimal regret in stationary and linear applications, however they are robust to both non-stationarity and non-linearity (and do not require a convex cost function whatsoever) where they show superior results.

2.2.5 Non-stationary Bandits

Non-stationary bandit problems is currently a very active research area. Slowly changing environments have been explored in depth in the Markov decision process literature [52, 140]. In their paper [63], Garivier and Moulines prove an $O(\sqrt{n})$ lower-bound of regret for generalized non-stationary bandits.

Discounted UCB(-T)

Discounted UCB and Discounted UCB-Tuned [63, 94] build on the work of UCB1 and UCB-Tuned for the original stochastic bandit problem, modifying the uncertainty padding estimate (the second term in the maximizing condition) and using a "local empirical average" instead of the traditional average considering older data in a discounted fashion. Effectively, discounted UCB creates an exponentially decayed version of UCB parameterized by some discount factor $\gamma \in (0, 1)$.

In the same fashion as the original UCB, in Discounted UCB, at time t we select the arm i that maximizes the form $\bar{x}_{i,t} + c_{i,t}$ where $c_{i,t}$ is a measure that "shifts" the estimate upward (often selected as a variance-adjusted estimator of the "exploratory value" of arm i). In Discounted UCB, however, we parameterize both terms of that equation with a discount factor γ. We use an indicator function $\mathbb{1}_\sigma$ defined as 1 if the condition σ is true and 0 otherwise and a list A_t that indicates the arm selected at time t. Specifically,

$$\bar{x}_{i,t}(\gamma) = \frac{1}{N_t(\gamma, i)} \sum_{s=1}^{t} \gamma^{t-s} x_s(i) \mathbb{1}_{A_s=i}, \tag{2.41}$$

where $N_t(\gamma, i) = \sum_{s=1}^{t} \gamma^{t-s} \mathbb{1}_{A_s=i}$ is the discounted average denominator and $x_s(j)$ is the payoff received from arm i so far. This equation serves to capture the mean *discounted* payoff estimate for arm i at time t. And $c_{i,t}$ is,

$$c_{i,t}(\gamma) = 2B\sqrt{\frac{\xi \log \sum_{j=1}^{K} N_t(\gamma, j)}{N_t(\gamma, i)}}, \tag{2.42}$$

where B is an upper bound on the reward, as in the general formulation of UCB1 and ξ is a parameter selected as $\frac{1}{2}$ in their paper, but with little further exploration.

Garivier and Moulines [63] shows that Discounted UCB achieves optimum non-stationary regret up to logarithmic factors, $\tilde{O}(\sqrt{n})$. By replacing the $c_{i,t}$ term with the tuned term from UCB-Tuned with an additional time discounting in the same γ^{t-s} fashion, we get a variant of Discounted UCB, Discounted UCB-Tuned that is expected to have the same empirical improvements as in the non-discounted case.

Sliding-Window UCB(-T)

Sliding-Window UCB (SW-UCB) [63] is an extension of Discounted UCB to use a sliding window rather than a continuous discount factor. A sliding window can be modelled as a discount factor of 100% for all data points older than some parameter τ representing the size of the window. To define the UCB functions for SW-UCB, [63] extends the same UCB1-type maximization process $\bar{x}_{i,t} + c_{i,t}$ for a τ period window as,

$$\bar{x}_{i,t}(\tau) = \frac{1}{N_t(\tau)} \sum_{s=t-\tau+1}^{t} x_s(i), \tag{2.43}$$

where $N_t(\tau) = \min(t, \tau)$ is the total length of the set being summed over, eliminating the discounting consideration from above[19]. The padding or optimism function, $c_{i,t}$, then, is,

$$c_{i,t}(\tau) = B\sqrt{\frac{\xi \log(\min(t, \tau))}{N_t(\tau, i)}}, \tag{2.44}$$

where $N_t(\tau, i)$ indicates the number of times arm i was played in the window of length τ. SW-UCB performs slightly better in their experimentation than the pure discounted approach and has the benefit of not requiring maintenance of data older than τ records. Both algorithms are strongly superior in regret to the Exp3 type algorithms and UCB1 with no non-stationarity modifications for the non-stationary problems tested.

Adapt-EvE

Hartland, Gelly, Baskiotis, Teytaud, and Sebag [73] present an extension to the UCB-Tuned algorithm [15] to deal with abrupt changes in the distribution associated with each arm. Adapt-EvE is considered a *meta-bandit* algorithm in that it uses a bandit algorithm at a higher level of abstraction to determine which bandit algorithm parameterization to use at each time. In particular, Adapt-EvE works by running the UCB-Tuned policy until a change-point in the underlying distribution is detected using one of many change-point detection algorithms (in their paper they use the Page-Hinckley test [18, 74, 116] with "discounted inertia" to only trigger in the change-point case, not the drifting case[20]). Upon detecting a change-point, a meta-bandit is initialized with two arms: one, which continues using the trained version of UCB-Tuned, and the other which resets all parameters and instantiates a new instance of Adapt-EvE. Training continues at the meta-bandit level (learning whether to continue using the trained data or learn again) and at the selected sub-level.

[19] In their paper, N_t is erroneously provided as the same discounted version provided for discounted UCB. This cannot be correct, as γ is no longer provided and the average would be incorrect.

[20] The rationale presented in Hartland et al. [73] for discounting the change-point statistic is that UCB-Tuned is capable of handling slowly drifting reward distributions within itself. We show in Section 3.4 that for certain forms of non-stationarity, UCB-Tuned is out-of-the-box insufficient to perform optimally and an informed detrending technique is preferred.

Kalman Filtered Bandit

Kalman filtered bandits [23, 70, 71] have been investigated in which the estimate of the mean payout of an arm is maintained by a recursive sibling Kalman filter parameterized by two *a priori noise* estimates σ^2_{ob} for *observation noise* or measurement error and σ^2_{tr} for *transition noise* (the non-stationarity error). Results are somewhat sensitive to these noise estimates. At each time step t, an estimate of the mean and variance for the arm played (with reward received $x_{i,t}$) is updated,

$$\mu_{i,t} = \frac{(\sigma^2_{i,t-1} + \sigma^2_{tr}) \cdot x_{i,t} + \sigma^2_{ob} \cdot \mu_{i,t-1}}{\sigma^2_{i,t-1} + \sigma^2_{tr} + \sigma^2_{ob}}, \tag{2.45}$$

$$\sigma^2_{i,t} = \frac{(\sigma^2_{i,t-1} + \sigma^2_{tr}) \cdot \sigma^2_{ob}}{\sigma^2_{i,t-1} + \sigma^2_{tr} + \sigma^2_{ob}}. \tag{2.46}$$

The non-played arms all have σ^2_{tr} added to their variance estimate for each time step, indicating how their uncertainty increases as time progresses. These equations and the general form of this model arise from the well-studied Kalman filter. The numerous published extensions to the Kalman filter for varying confounding factors can likely be applied in this space.

This approach performs very well in drifting and change-point cases, however is outperformed by Adapt-EvE in the well-defined change-point case. The resilience to form of non-stationary make this a valuable approach in the event the parameters can be well predicted. This has not been explored within a contextual context, with Thompson sampling or probability matching techniques or with an optimistic approach.

2.2.6 Probability Matching and Thompson Sampling

W. R. Thompson (1933), "*On the likelihood that one unknown probability exceeds another in view of the evidence of two samples*" produced the first paper on an equivalent problem to the multi-armed bandit in which a solution to the Bernoulli distribution bandit problem now referred to as *Thompson sampling* is presented.

The stochastic solution presented by Thompson [144] involves *matching* the probability of playing a particular arm with the arm's inherent "probability of being the best" given the data observed by sampling from each distribution precisely once and selecting the maximum sample. The language *probability matching* arises from this intuition and seems to originate from Morin [112]. *Probability matching* is extensively used in the experimental psychology literature to describe the behavior matching action probabilities to the probability of an outcome. This concept is distinct from the actual implementation of sampling precisely once from the posterior estimate to simulate the optimality pseudo-distribution, which we refer to as Thompson sampling. A factor motivating this interplay of nomenclature is the increasingly common use of multi-armed bandit processes in the modelling of animal and human psychology and behavior [e.g., 123, 137].

Scott [132] applies a strictly Bayesian framework to presenting Thompson sampling and specifically calls it *randomized probability matching*. We will simply use the language of *Thompson sampling* through the rest of this discussion.

Thus far, we have not discussed any probability matching solutions. Probability matching is a technique that draws on the Bayesian literature and theoretically applies directly to any variant of the bandit problem where a probability of an arm being the best can be expressed as a distribution. Recently, it has been shown in many analyses that probability matching techniques are competitive with the state of the art in a variety of bandit and other learning contexts. We show later that a nonparametric form of probability matching is available and compare it to existing similar solutions (such as the quasi-parametric binomial probability matching technique of Agrawal and Goyal [6] and the bootstrap-derived approach of Eckles and Kaptein [55]).

Recent research in Thompson sampling has provided an information-theoretic analysis [130], various proofs and demonstrations of regret minimization [68, 72], a technique to apply Thompson sampling via the online bootstrap [55], exploration of the cold-start problem[21] in recommender systems [115] and numerous applications of the technique [27, 68, 114, 135].

Strict bounds on regret were a hindrance to theoretical adaption of generalized Thompson sampling, however, recently, bounds for a single specific model (traditional K-armed bandits with beta prior distributions) have been discovered by Agrawal and Goyal [6]. For $K = 2$, their bound on regret is given as $O(\frac{\ln H}{\Delta})$; for $K > 2$ the bound is significantly worse, as $O\left(\frac{\ln H}{\sum_{i=2}^{K} ({\Delta_i}^2)^2}\right)$. Significantly, the information-theoretic work of Russo and Van Roy [130] proves efficient ($O(\log H)$) regret bounds for Thompson sampling and show convincingly that Thompson sampling performs comparably to a correctly-tuned UCB-type algorithm in general. This is a result which had been expected, however is significant as Thompson sampling is a more general solution than any particular implementation of a UCB-type algorithm.

Empirical research show this strategy has both excellent results in traditional constrained variants and in variants with less strongly maintained assumptions.

Thompson Sampling extends well to cases with generalized distributions for the arms. For example, a Bayesian approach for Thompson Sampling for the common case of Bernoulli arms is made computationally efficient and simple to implement by sampling from the appropriate product of conjugate beta distributions parameterized in such a way that only tracking the number of successes and number of failures is necessary.

In order to formalize the matching of our play probabilities with the probability of a given play being the best play, we adopt a Bayesian framework and, in general, a parametric distribution over a parameter set θ. We can compute the probability at time t of a given arm providing optimal reward as,

[21] The cold-start problem is a particularly apt use of bandit modelling. Specifically, the problem models the issue of being unable to draw any recommendations for new users until sufficient data has been collected for said user to fit an appropriate model or prediction to his or her preferences. The multi-armed bandit model provides exploratory guidance in some contexts to help address this problem.

$$\int \mathbb{1}\left[S_t = \operatorname*{arg\,max}_{i=1,\ldots,K} E_\theta[x_{i,t}]\right] P(\theta|x)d\theta. \tag{2.47}$$

Rather than computing the integral, Thompson [144] and others show that it suffices to simply sample from the estimated payoff distribution at each round and select the highest sampled estimated reward. That is, the repeated selection of the maximum of a single draw from each distribution, produces an estimate (and thus selection behavior) of the *optimality distribution*. This result, while long known, is surprising and valuable, turning an intractable problem into a computationally simple one.

Optimism in Probability Matching

A recurring perspective on the efficient use of uncertainty within such a multi-armed bandit (and exploratory learning in general) has been that of "optimism in the face of uncertainty" [15, 99, 113, 142]. The idea is presented as a method for treating uncertainties and balancing exploration: when a statistical uncertainty is present, a small but consistent gain in outcome [42] can be achieved by simply remaining optimistic and assuming the value is in the "more desirable" portion of the distribution under uncertainty. We show in a later section that "small but consistent" may not be the right language to describe optimism, indeed the gains to optimism are inconsistent and can be large.

This idea has been seen already in many of the static (non-probability matching) algorithms presented prior. For example, any UCB-type algorithm derives its action estimates from an "optimistic" surrogate about the state of the empirical estimate. This form of static optimism is the basis of most algorithms for multi-armed bandits, though the mechanism for defining optimism is variable.

In probability matching, *optimism* is usually defined as *mean-optimism*, that is, sampling only from the distributions which are above the mean. Often we will refer to the distribution of values only above the mean as the *optimistic surrogate distribution*. Later, we will show how to efficiently sample from this distribution for any tractable distribution, and then show how empirical sampling and resampling can be used to improve the results when the true underlying distribution is unknown.

The Bernoulli Approach to Nonparametric Thompson Sampling

In the case of bounded bandits (that is, arms which have a reward guaranteed to be in some range), Agrawal and Goyal [6] propose an algorithm which transforms any bounded distribution to look like a Bernoulli distribution for the sake of sampling. First, the Bernoulli implementation of the simplest form of Thompson Sampling is presented in Figure 2.3.

This implementation as described relies on the way the Beta($W+1$, $L+1$) distribution behaves accurately as an estimator of the mean and variance of the binomial (repeated Bernoulli trials)

```
Data:
$i$: the set of available arms ($i = 1, 2, ..., K$)
$\Omega$: some stopping rule (e.g., exit when the horizon $H$ is reached)
Result: Thompson Sampling for Bernoulli Bandits
// Set counters of wins and losses for each arm to their prior (or one, for
   ignorance).
initialize $\forall i \ \ W_i \leftarrow 1$ ;
initialize $\forall i \ \ L_i \leftarrow 1$ ;
while ($\Omega$) do
    $\hat{r}_* \leftarrow -\infty$;
    $S_t \leftarrow nil$;
    for each arm $i$ do
        // Draw once from the beta distribution representing each arm.
        $\hat{r}_{\text{test}} \leftarrow$ draw from Beta($W_i, L_i$);
        // Record the best draw.
        if $\hat{r}_{\text{test}} > \hat{r}_*$ then
            $\hat{r}_* \leftarrow \hat{r}_{\text{test}}$;
            $S_t \leftarrow i$;
        end
    end
    reward $\leftarrow$ play($S_t$);
    if reward == 1 then
        $W_{S_t} \leftarrow W_{S_t} + 1$;
    else if reward == 0 then
        $L_{S_t} \leftarrow L_{S_t} + 1$;
    end
end
```

Figure 2.3: Thompson Sampling for Bernoulli Bandits

distribution and as such is only appropriate for arms drawn as a Bernoulli trial. When the rewards are generated from an arbitrary *bounded* distribution, Agrawal and Goyal [6] suggests a simple modification: assume the rewards are drawn from a distribution bounded between zero and one, then, when a reward is observed, it performs a Bernoulli trial (i.e., a weighted coin flip) and updates W_i or L_i according to the payoff. The modified algorithm only augments the play function in the pseudocode above to become one which plays and then flips a coin based on the value of the result.

This technique only works if the distribution is bounded in a range that can be treated (via scaling) as [0,1]. In Chapter 3, we present both an empirical sampling technique which allows the implementation of an *optimistic* variant of this strategy and a novel technique which produces a distribution-free sampler without the boundedness requirement.

Bootstrap Thompson Sampling

Eckles and Kaptein [55] present an implementation of Bootstrap Thompson Sampling (BTS) which produces a scalable, robust approach to Thompson sampling. We show in this section that their implementation can be modified to support an arbitrarily scaled form of optimism and a completely nonparametric approach at minimal cost, at least for some subset of the problem space. We then show that the sampling strategies shown in the Sampling Uncertainty section can be applied here to avoid a number of poor edge-case performances and essentially control the *risk* of such a policy. Finally, we expand the implementation to a categorical-contextual policy which subsets the sampling space into categories to implement a refinement of the model.

The initial implementation of BTS involves selecting a parameter J which simultaneously controls the computational complexity and relative greediness of the model. Upon receipt of each reward r_i, BTS trains J parametric models on the assumed underlying distribution by considering the reward in each model with probability one-half. To select the next arm to play, BTS, on a per arm basis, chooses one of the J replicates and uses the expected value of that model to predict an empirical mean payoff. In the Thompson sampling style, the highest of those estimated payoffs is selected as the action for this iterate. As the empirical mean is deterministic across the replicates[22], if J is too small, the decision becomes fundamentally greedy, choosing to only play the best empirical arm prematurely. This technique works well, showing performance competitive with the traditional model and even exceeding it in the case of heteroskedasticity.

Change-Point Thompson Sampling (CTS)

Mellor and Shapiro [108] present an algorithm based on a two-stage Thompson Sampling technique combined with a particle filter based learning tool [3, 59, 60] for a specific form of non-stationarity in the distributions. The non-stationarity utilized is a constant hazard rate change-point problem, where the payoff distributions have sudden and complete changes at a constant (but unknown) rate.

They model the problem as a non-contextual bandit simulation, where, because of the complete and independent nature of the change, after a change is found all prior data has no value. In order to utilize the randomized probability matching technique within this dual-unknown, they sample first the posterior (produced from a particle filter technique) for the hazard rate, and then use their drawn sample to sample from the bandit arms. This idea that Thompson sampling can be applied at a recurring process inspires the basis for one of our linear bandits Thompson sampling implementation experiments in Chapter 3.

A big contribution of the Mellor and Shapiro [108] paper is their simulation technique. They amalgamate the (non-stationary) PASCAL EvE challenge [79], the Yahoo! [102] real world news

[22] Computing the expectation, rather than sampling from the subdistributions, gives us an efficient sampler in the bootstrap sampling distribution θ, but does not maintain the “uncertainty awareness” property in the individual distributions. A sufficiently large J returns the uncertainty awareness property of Thompson sampling.

click-through data and a foreign exchange simulation [109]. Their algorithm shows great real world performance, but underperforms on the simulation data.

2.3 Application Area

The multi-armed bandit represents the advertising problem explored in this work well. This much can be seen simply by the size of the substantial related literature. Much of the literature in place considers the inverse problem: selecting advertisements to display or purchase for a given vendor to maximize its outcomes [117, 129], rather than the problem of maximizing an outcome given a set of advertisements. These problems are related, but the experimental design on the sales copy and web design side of the problem provides a substantially larger distribution of marketer-selected variables, as well as a significantly different sampling distribution for experimentation over.

One of the most significant related works in the application area is that of Li et al. [101], where the algorithm known as LinUCB is presented. LinUCB presents the first work to provide a simple regression-based framework for multi-armed bandits and applies it explicitly in selecting the optimal (in terms of revenue, time on site, or other objective criteria) news story to display on the homepage of Yahoo.com. Their result is successful, applies the set of readily available contextual variables in a clear and easy to understand manner and shows promise in extending to more complicated variants of the same problem.

Scott [132] and Scott [133] present perspectives of Thompson sampling as applied to the online advertising problem. Scott [133] in particular, provides a number of designs for experiments intimately related to the problem under consideration in this work: explicit experiments in web design or content selection. For example, an experiment for selecting button color for a *call to action* button is presented where the experimental configuration is as a logistic regression with a boolean variable in place for each of the available options. Scott [133] also draws attention to the mixed modality of the objective function: showing that it is not always simply sales (or immediate profits) that is the correct variable to optimize, but rather many intermediaries such as measures of quality and measures of satisfaction are necessary to produce a long-term sustainably profitable business.

A work by Tang et al. [143] produces a system for using the multi-armed bandit problem to select the appropriate advertisement *layout* in order to increase effectiveness of an advertisement (measured either in click-through rate or total revenue in an online advertisement auction context) within an existing webpage. Their work relies both on a large corpus of work related to predicting click-through rates [4, 19, 43, 69, 124] and a large corpus of available contextual variables from the context of users being signed in to their LinkedIn accounts when displayed the advertising in question. Importantly, their work, as in all of this work, finds that Thompson sampling is a deceptively competitive tool for optimizing bandit problems of this nature.

Chapelle, Manavoglu, and Rosales [43] provide a comprehensive look at both the online advertising industry itself and the utilization of a logistic regression to predict the response to display advertising via a number of contextual variables related to both the advertisement itself and the context within which it is to be displayed. Such a prediction can be applied within a multi-armed bandit context (in a technique related to that of Li et al. [101] or of our own LinTS model presented in Section 3) to produce a revenue (or response) maximizing online exploration-exploitation aware system.

As a quick review of the necessary background, the economic model under which our problem is operating looks similar to many online eCommerce businesses: a product is for sale and there is a user acquisition process which directs users to a pre-sales environment, where they are presented the marketing information and opportunity to purchase the product. Three common forms of user acquisition exist for businesses of this sort, each having two parties, the advertiser, who wishes to acquire users to his marketing technique and the publisher, who operates an existing medium with captive users to be advertised to. The three common user acquisition processes are: (1) a long-term approach called search engine optimization (SEO), where the goal is to increase relevancy in the search engines (e.g., Google.com) in order to acquire traffic from users who were searching for target keywords; (2) a medium-term branding-oriented approach where advertisers pay *cost per mille* (CPM) for views (priced per thousand), but not necessarily clicks, of their advertisements on a selected other website; (3) the short-term approach of purchasing clicks directly, paid for per-click on the advertisement. There exists a fourth case where advertisers pay publishers of their advertisements per sale (CPA) of their product (or other *action*), but it is not directly considered in our work, however, our primary goal (increasing "actions" or sales) could then be seen from the perspective of the publisher.

In the first case, there is little short-term control over the number or cost of new users. In the second case, the advertiser must select an *advertisement creative* in order to have the maximum desired effect. Often, CPM advertising will be used for *branding* campaigns, or advertising campaigns where the immediate goal is not necessarily increased sales, but rather a longer term relationship with the public. Our work does not consider the branding case. In the immediately increased sales variant, the advertisement creative is selected to maximize clicks through to their website, at which point the marketer must select an appropriate website design and sales text in order to maximize sales or profit. In the third case, the advertiser pays only for clicks through to their website, in which case only the last step, selecting the appropriate website design and sales text is paramount.

Chapter 3

Towards the Use of Multi-Armed Bandits in Advertisement Testing

3.1 An Extensible Platform for Simulating Bandit Problems

We constructed a highly-extensible, multi-paradigm simulation and experimentation platform for quantifying and comparing bandit strategies across a variety of contexts. The platform provides an efficient way of running comparable controlled experiments for the purpose of learning about policies or parameters in a risk-free environment. Fundamentally, the structure of the platform is made up of three components: simulators, arms and policies. The platform is highly parallel in both world variants, replicates and policy or parameter choices and is set up to immediately spawn separate threads to maximally utilize available CPU hardware.

3.1.1 Implementation

Simulator

A simulator is a container that holds the current state of the world in terms of available arms and maintains detailed statistics on the results throughout the simulation. The simulator is initialized with the set of information the experimenter wishes to retain and a pointer to the instruction files for the problem set (a collection of different world states, each made up of a number of arms) and the policy set being experimented upon (a collection of policies and their configurations). In the set of problems and policies, *tags* or shared identifiers are used to categorize worlds and policies in arbitrary dimensions in an easily comparable way.

Arms

An arm is an object which contains its own state information (usually distributional parameters such as mean and variance, but also state variables related to non-stationarity or contextual confounders), knows how to calculate its own expected value (if possible) and can produce a draw from its result distribution. Arms within this context can have context (provided by the simulator at each time step) at both an arm and world level, can drift or have change-points, and can generally introduce any arm-level deviations of the problem desired via further extension of the prototypes.

Policies

A policy or "strategy" is a self-contained unit that interacts with the simulator to pull arms and receive rewards. Policies are implemented by the experimenter, in their own class which receives only the information necessary to inform the policy. We have included an implementation of many of the policies explored in Chapter 2. This is important to produce reproducible real world applicable understanding of policies and their parameters. In certain contexts, modifications to the simulator are available (with appropriate selection at initialization) to allow policies the ability to observe information that they would not normally be able to access. This is used in particular later when we explore the prescient measures to elevate our understanding of how particular modelling errors can effect the result.

A policy generally comes with an object which specifies its configuration. This allows different parameters for a policy (such as ϵ in the ϵ-strategies, weight functions in the case of our own WLS implementation or discount rates in strategies like POKER) to be explored in a way that allows easy comparison in the simulation output.

Measurements

The simulator is capable of reporting a wide variety of desirable measurements including all the computable definitions of regret given in Chapter 2 (weak and strong regret; stochastic and expected value; non-optimal play count) and a number of measures of "divergence" between the arms (updated per pull, to handle cases of non-stationarity) including Kullback-Leibler (K-L) divergence [98] (symmetrized [83]), resistor average [83] and the J-measure [81] computed distribution-wide (across all arms), and in the maximizing, minimizing and closest-arm subcases.

Included with the simulator is a small set of visualization tools intended to transform the iterate- and replicate- level variable outputs of the simulator into appropriate graphical representations for understanding, analysis and publication. These tools allow categorization over any classification variable available for the underlying problems or worlds examined, including assumption violations, distribution choice or size/scale of the problem, as well as categorization over policy selection and policy configuration.

3.1.2 Problems

While many problems can be simulated, we argue that the multi-armed bandit problem is extremely difficult to simulate in a way that generalizes to applied problems effectively. As such, one of the simulators we have provided is the "Yahoo! News Click-Through" simulator, which uses the technique of Li, Chu, Langford, Moon, and Wang [103] to allow the reconstruction of bandit state from the wealth of real Yahoo! data and a simulator which uses live currency-pair data [109] in a similar method to simulate forex trading with bandit algorithms. These produce real world environments with which simulation-driven error can be detected by a cautious experimenter.

Additionally, we provide re-implementations of the PASCAL Exploration vs. Exploitation (EvE) 2006 environments. PASCAL EvE was a competition to identify new and experimentally optimal non-stationary bandit algorithms. The challenge provided six artificial simulations: frequent swap, where the best arm switches rapidly; long gaussians, where the payoff drifts in a gaussian fashion over a long time period; weekly variation, in which two sinusoidal components vary the payoff probability with the longer being dominant; daily variation, in which two sinusoidal components vary the payoff probability with the shorter being dominant; weekly close variation, in which response rates are very close together and constant, in which there is no non-stationarity.

Finally, we have a set of our own synthetic simulations, covering features including: drift (options including: linear, exponential (varying size), sinusoidal, random walk, none); change-points (constant, poisson, none); arm class (binomial, Gaussian, contextual); mean and variance; number of arms. This set of base simulations can be extended to introduce further "world problems" (deviations from the assumptions) by future authors to improve the quality of simulation results.

This platform drives most of the experimentation in the following sections. Where appropriate, assumption violations and modifications to the problem have been implemented as novel problem sets (and even novel arm classes) in the simulation framework to produce research which is both reproducible and extensible to new, but related, questions.

3.2 Linear Model Thompson Sampling: LinTS

LinUCB [101] as discussed in Section 2.2.4 and most other regression-based UCB models can be transformed to use Thompson sampling if one can get estimates of the higher (non-mean) moments of the distribution. Indeed, one almost always has sufficient information to do this, as most UCB mechanisms rely on a formulaic representation of variation to capture the definition of the upper confidence bound. We extend the model only to the linear regression case similar to that of LinUCB.

At a high level, we extend the model of LinUCB, to fit a model of the form,

$$Y = \alpha + \beta_0 C + \beta_1 A + \gamma C \cdot A + \epsilon \tag{3.1}$$

,

where Y is the expected reward for the given set of parameters, $C = [C_n \in \mathbb{R}^K]_{n=\{0,\ldots,N\}}$ is the set of contextual variables and $A \in \{0,1\}^K$ is a set of arm dummy variables. The fitting technique used is left to the implementer, however, it must provide an adequate estimate of the moments of the (true) distribution. In the frequentist domain penalized least squares techniques such as ridge regression show promise, especially in the case of uncertainty where the relevance of the contextual variables is uncertain and some level of overspecification is likely.

Without loss of generality[1], we assume the normally-distributed linear regression fit with a coefficient (mean estimate) and standard error (variance estimate) provided for each contextual variable, arm and interaction effect. This scheme is inherently Bayesian in nature, however, to improve computability in a practical sense, we utilize a traditional linear regression model with penalized coefficients (ridge regression). Ridge regression can be represented as a Bayesian regression technique with a fixed prior on the βs making some of the result interpretation more tractable. The algorithm used is roughly as follows:

1. **Fit the model**. Using the fitting technique given, produce coefficients (μ) and standard errors (σ^2), generating normally distributed random variables $\xi \sim N(\mu, \sigma^2)$.

2. **Calculate a summed distribution $\tilde{Y}$ for each arm**. For each arm i, set $A_i = 1, A_j = 0$ where $(j : \{j \neq i\})$, sum the random variables multiplied by their observed context vector C_n to get an estimate of $\tilde{Y}_{i,C} \sim N(\tilde{\mu}, \tilde{\sigma^2})$ for arm i and context C.

3. **Apply Thompson Sampling**. With each $\tilde{Y}_i$ in our given context, we now have a model of the (current) distribution of expected reward for each arm given our current knowledge. Using this, we sample according to the probability that reward is the maximizing reward in the tradition of Thompson Sampling. We explore a few methods of doing this in a computationally tractable and affordable way.

This provides a result that is similar to that of LinUCB, however utilizes the Thompson Sampling strategy rather than the constrained upper confidence bound approach. In order to optimize the outcome, we test a number of uncertain implementation details within our simulation architecture, expecting that each question has a "correct" answer that can be generalized to all problems.

3.2.1 Optimistic Thompson Sampling in LinTS

We propose a technique that exploits the assumptions of the linear model and the probability matching technique of Thompson sampling. Based on the assumption of normality, the regression coefficients, $\hat{\beta}$, are normal and hence the predictions $\hat{y}_t$ are normal. We then optimistically sample (drawing only values above the mean) from a normal distribution with mean $\sum_i(\hat{\beta}_i \cdot x_{i,t})$ and variance $\sum_i (\widehat{\text{Var}}(\hat{\beta}_i) \cdot x_{i,t}^2)$ to approximate $\hat{y}_t$. A more general form of this fundamentally Bayesian

[1] As necessary, replace distributional assumptions in our algorithm with those of your fitting technique.

algorithm can be constructed utilizing the techniques of Bayesian regression [111] at the cost of generally higher computational complexity.

In this section, we have presented a computationally efficient, flexible model, linear regression-based Thompson Sampling implementation, LinTS, which is easily extensible to any contextual model, simple to implement and provides low regret and easily interpretable results in practical experiments. Further research on LinTS is necessary to prove (theoretical) asymptotic bounds on regret and to better understand how results from the linear regression model can be interpreted for non-prediction type tasks[2].

In the next section, we extend this model and explore the space of non-stationary regression models and show how existing time series techniques can be extended to LinTS in order to efficiently handle both slowly drifting *and* rapidly changing true world states.

3.3 Experiments in Thompson Sampling

In this section, we explore some of the considerations and questions that arise when implementing an optimistic sampler in a parametric (specifically, model-fitting) context. These questions are experimented within the context of the LinTS algorithm, but the experiments have been generalized to other forms of Thompson sampling via the simulator. Each of these questions is explored in more detail in the coming sections, but as a quick introduction and reference a summary is provided here.

- Q_1: **How does the measure of centrality effect optimism?** When computing an optimistic sampler, we must choose a form of optimism. Indeed, optimism can take many forms: one could drop the least favorable observations, sample only from the top 10% or otherwise be optimistic. In general, optimism with regard to Thompson sampling is tested in the context of mean-optimism, where the results from the portion of the distribution above the mean are used. We explore using the median in certain contexts to modify the impact of skewness and outliers and find that there is no significant effect here.

- Q_2: **How does the concept of *estimative uncertainty* differ from the standard error for sampling?** We test this question by using our within-simulator prescience to the true underlying variance of an arm. We find two surprising results: (1) the effect size of removing true (or even computed) underlying variation from the estimate under sampling is large and (2) this effect appears to be shared with the effect of optimism – that is, the benefit to removing estimative uncertainty is substantially smaller in optimistic sampling than it is in

[2] Specifically, to interpret linear regression coefficients as causal in the econometric sense or to derive many proofs of unbiasedness and efficiency, one must assume the regression samples are drawn randomly. In the linear bandit case, regression samples are drawn according to the Thompson Sampling process or UCB maximizing process on the previous time world state. How this affects the model is an open research question which requires further exploration.

non-optimistic sampling. This result provides a piece of evidence towards the understanding of why optimism is beneficial to the exploratory process under uncertainty.

- Q_3: **How does k-sampling from $\tilde{Y}_i$ effect the distribution of regret?** To compute the likelihood of a given arm being the best, we can simply draw k times from each distribution and combine the estimates in a variety of ways. We *ex ante* expect that higher k's learn faster, but have a marginally higher computation cost and are more likely to get stuck on incorrect solutions.

  We explore different values of k and different combination strategies and find some particularly interesting results about the true nature of optimism: specifically, excessively optimistic sampling (taking the most optimistic of the k samples per arm) seems to perform asymptotically better in k suggesting that the ratio of prospective best values is a determinant in bandit performance. This is a surprising result as the surrogate (sampling) distribution is no longer influenced strongly by the true underlying mean, but only the most optimistic forecast. A thought experiment with $k = \infty$ provides an interesting but poorly understood look at this result, showing that the result only holds if the ratio of two arms' most optimistic forecasts is well defined.

3.3.1 Measure of Centrality

The traditional implementation of optimistic Thompson sampling implements optimism as a surrogate distribution bounded at the mean or expectation of the distribution. This is intuitive, as our goal is to maximize total reward, however, in smaller samples drawn from distributions with sufficient skew it is plausible that a different measure of centrality such as the median may be preferred. In general, in our model of optimism, there are an infinite set of *degrees of optimism*, each representing a portion of the distribution to be discarded. That is, a median optimism can be treated as dropping exactly the lower $\frac{1}{2}$ of the distribution.

We compared mean-optimism and median-optimism using the beta distribution as the estimator for the binomial distribution. This gives us the cases of positively skewed, negatively skewed and non-skewed parameterizations. We selected the beta distribution approximation solely for the ease of varying the relevant shape parameters[3]. The results shown appear to generalize well to other distributions along their relevant skewness.

After extensive experimentation, including experimentation with artificially introduced error, we find the measure of centrality is not a significant factor in the performance of optimism, even in highly skewed distributions, when measured by expectation regret, total reward or suboptimal

[3] Recall, the beta distribution is parameterized on α and β, shape parameters. Kerman [90]'s approximation to the median is used and distributional parameters are selected such that the numerically computed relative error of the approximation is $< 10^{-6}$.

plays. We do reproduce the small benefit accrued to optimism seen in prior work and explore that in more depth going forward.

3.3.2 Estimative Uncertainty

When sampling with the goal of maximizing the expected return, it is (generally) functional but not optimal to sample from the distribution denoted by the mean and the standard error of the model's estimate (for example, a LinTS [36] regression-based model or the similar Bayesian regression technique described in Eckles and Kaptein [55]). The standard deviation of the estimate can be decomposed into two components, *estimative uncertainty*, caused by sampling error and *underlying variance*, which exists absolutely in the underlying distribution. The goal of Thompson sampling is to exploit the *estimative uncertainty* where it could plausibly lead to higher rewards in future iterates, but it is wasted effort to play suboptimal arms if the variance is provided solely by *underlying variance*.

We attempt to remove the *underlying variance* in two methods. One method, where arms are of known distribution with known standard deviation, simply subtracts the known standard deviation from the estimated standard error, giving an accurate estimate of our estimative uncertainty[4] and another where we empirically maintain an online estimate of the variance and subtract that from the standard error of the model to produce our sampling distribution. Not surprisingly, in non-optimistic (unbiased) sampling, the prescient technique outperforms the empirical technique, however they both outperform using the raw standard error by a large margin. Table 3.1 presents results which are an average across all our representative normal distribution worlds in each of 50, 100 and 1000 iterates each with 2,000 replicates demonstrate the benefit in the non-optimistic sampling case.

The gains did not maintain their magnitude when (symmetric) optimism was invoked, as optimism accrues a fairly large benefit itself, but does not appear to accrue much additional benefit for the uncertainty correction as later iterates' variance goes to zero. This may suggest the optimism result is from systematic early underexploration in the Thompson sampling strategy. Further research is necessary to explain how optimism and the uncertainty correction interact to produce similar scale results.

A further set of tests were conducted scaling the variance of the sampling process in arbitrary ways (squaring, dividing by constants) with no lucrative benefits. The unscaled variance always outperformed on average with or without estimative uncertainty adjustments.

[4] Note, we must floor this value at zero, as a negative variance does not have meaning, but could arise in this model due to the statistical properties of the distribution.

Table 3.1: Results from eliminating estimative uncertainty in the unbiased sampling case. $***$ denotes $p < 0.01$ (against the pairwise null hypothesis of no effect of changing from raw standard error).

	Average $\bar{R}^E$ (SD)	**Improvement**
Raw Standard Error	534 (23.47)	–
Prescient Uncertainty Correction	371 (31.69)	43.9% ***
Empirical Uncertainty Correction	419 (27.92)	27.4% ***

Table 3.2: Results from eliminating estimative uncertainty in the optimistic sampling case.

	Average $\bar{R}^E$ (SD)	**Improvement**
Raw Standard Error	404 (27.23)	–
Prescient Uncertainty Correction	377 (31.99)	7.1%
Empirical Uncertainty Correction	395 (27.49)	2.3%

3.3.3 Sampling Uncertainty

We can change the expected variance of the (optimistic or non-optimistic) sampler by sampling more than once. The Thompson [144] strategy of drawing once from each positively biased distribution is distinct from, say, drawing twice and using the mean of the samples. Here we experiment with the variance of our sampler and its interaction with optimism. An initial intuition is to simply compare optimistic sampling under the single sample case to a number of k (say, 2, 5, 10) sample cases under different strategies to produce an action sample value. We experiment with a number of replication strategies:

1. **Sample means**. In this strategy, the k samples are averaged to produce our action value. This reduces the variance of the sampling strategy in both cases, around the mean in the non-optimistic case and around the biased-mean in the optimistic case.

2. **Most optimistic.** In this strategy, the most optimistic element of the k samples is taken. In the optimistic case, this serves to create a more optimistic sampler than pure 1-sample optimism. In the non-optimistic case, this creates an optimistic sampler without the hard constraint at the measure of centrality.

3. **Least optimistic.** In this strategy, the least optimistic element of the k samples is taken. In the optimistic case, this serves to dampen optimism.

4. **Least deviation.** In this strategy, the sample which is closest to the measure of centrality is taken. This is equivalent to least optimistic in the optimistic case, but in the non-optimistic case, it is simply a conservative sampler.

Our results are presented in Table 3.3. We see convincing evidence in favor of optimism, even *excessive* optimism, showing both the least regret and the lowest standard deviation of that regret. When $k = 1$, we should be indifferent between replication strategies, as they will all simply return their first sample. One can think of the Optimistic-Most Optimistic sampler as encouraging *excessive* optimism by sampling optimistically, then taking the most optimistic of those samples. We get a distribution that is on average further in the tail of the distribution (proportional to k). Similarly, one can think of the Unbiased-Most Optimistic sampler as a simulation implementation of optimistic sampling and would expect that as $k \to \infty$ that sampler will perform more like the Optimistic-Most Optimistic sampler (with more computation cost). It is interesting to note that setting k (finitely) higher appears to improve the performance of the Most Optimistic replication strategy asymptotically, suggesting that the increase in relative tail probabilities (across arms) continues to improve the result. The Sampled Mean and Least Deviation strategies act to *reduce* optimism (or possible pessimism, in the case of the unbiased sampler) by reverting samples towards the mean.

Table 3.3: Results of a selection of replication strategies.

#	**Sampling Policy**	**Replication Strategy**	k	**Average** $\bar{R}^E$	**SD**	$\sum_t x_{S_t,t}$	**SD**
1	Optimistic	Least Deviation	1	1.787940	2.812723	28.25736	6.919474
2	Optimistic	Least Deviation	2	2.038760	3.147973	27.99885	7.006590
3	Optimistic	Least Deviation	3	2.131280	3.288711	27.87352	7.077301
4	Optimistic	Least Deviation	4	2.239480	3.419299	27.80768	7.107457
5	Optimistic	Least Deviation	5	2.299420	3.503120	27.64148	7.135077
6	Optimistic	Least Deviation	10	2.433740	3.657933	27.56664	7.260929
7	Optimistic	Sampled Mean	1	1.794856	2.814387	28.22871	6.892612
8	Optimistic	Sampled Mean	2	1.797440	2.831595	28.14065	6.899314
9	Optimistic	Sampled Mean	3	1.774540	2.815696	28.20008	6.900778
10	Optimistic	Sampled Mean	4	1.742480	2.808865	28.29840	6.892055
11	Optimistic	Sampled Mean	5	1.782160	2.845401	28.22016	6.888765
12	Optimistic	Sampled Mean	10	1.772120	2.842582	28.21684	6.874437
13	Optimistic	Most Optimistic	1	1.782860	2.795673	28.23848	6.932394
14	Optimistic	Most Optimistic	2	1.627420	2.594976	28.39532	6.841655
15	Optimistic	Most Optimistic	3	1.539440	2.467145	28.42102	6.798019
16	Optimistic	Most Optimistic	4	1.488580	2.410582	28.52400	6.769490
17	Optimistic	Most Optimistic	5	1.454160	2.358807	28.52707	6.768880
18	**Optimistic**	**Most Optimistic**	**10**	**1.373060**	**2.249344**	**28.61777**	**6.743186**
19	Optimistic	Least Optimistic	1	1.801640	2.820853	28.19821	6.926717
20	Optimistic	Least Optimistic	2	2.017980	3.129366	27.98708	6.994345
21	Optimistic	Least Optimistic	3	2.133160	3.293516	27.91489	7.065552
22	Optimistic	Least Optimistic	4	2.237860	3.415851	27.76209	7.112057
23	Optimistic	Least Optimistic	5	2.303500	3.496363	27.65757	7.141841
24	Optimistic	Least Optimistic	10	2.389940	3.633834	27.60138	7.230841
25	Unbiased	Least Deviation	1	2.414180	3.317339	27.52917	7.156890
26	Unbiased	Least Deviation	2	2.433260	3.483726	27.51896	7.196959
27	Unbiased	Least Deviation	3	2.457000	3.575881	27.59020	7.233475
28	Unbiased	Least Deviation	4	2.488600	3.635268	27.54201	7.226739
29	Unbiased	Least Deviation	5	2.483780	3.664905	27.52593	7.264307
30	Unbiased	Least Deviation	10	2.531200	3.758365	27.47851	7.351306
31	Unbiased	Sampled Mean	1	2.419500	3.311818	27.62926	7.188822
32	Unbiased	Sampled Mean	2	2.439500	3.445440	27.52135	7.195143
33	Unbiased	Sampled Mean	3	2.428440	3.488071	27.59352	7.189119
34	Unbiased	Sampled Mean	4	2.461000	3.550799	27.56638	7.213879
35	Unbiased	Sampled Mean	5	2.444140	3.556658	27.54016	7.220508
36	Unbiased	Sampled Mean	10	2.489640	3.653163	27.49565	7.206094
37	Unbiased	Most Optimistic	1	2.408760	3.310960	27.61688	7.143456
38	Unbiased	Most Optimistic	2	1.948180	2.920867	28.07839	6.966973
39	Unbiased	Most Optimistic	3	1.797240	2.756331	28.16101	6.846418
40	Unbiased	Most Optimistic	4	1.675020	2.627932	28.30523	6.862079
41	Unbiased	Most Optimistic	5	1.596760	2.556773	28.41517	6.812572
42	Unbiased	Most Optimistic	10	1.482580	2.393168	28.49253	6.744063
43	Unbiased	Least Optimistic	1	2.429660	3.322913	27.54089	7.149666
44	Unbiased	Least Optimistic	2	2.993740	3.845496	26.99108	7.440987
45	Unbiased	Least Optimistic	3	3.332900	4.118677	26.63976	7.647496
46	Unbiased	Least Optimistic	4	3.592980	4.287377	26.41372	7.746072
47	Unbiased	Least Optimistic	5	3.745220	4.392518	26.23426	7.864180
48	Unbiased	Least Optimistic	10	4.173680	4.586818	25.83976	8.000412

3.4 Non-Stationary Time Series Techniques

Change-point analysis, also known as change detection or *structural breakpoints* modelling, is a well-studied problem in the applied stochastic process literature. Intuitively, a change-point is a sudden, discrete or "drastic" (non-continuous) change in the shape of the underlying distribution. In an offline fashion, change-points may be detected efficiently and with an adequate set of tuneable parameters with clustering algorithms. For bandits, the problem is necessarily an online problem and offline algorithms for change point detection are not feasible. The basic idea of online change-point bandits is to use a mechanism to detect change-points, generally parameterized for some acceptable *false alarm rate*, and then utilizing some mechanism to "forget" learned information after each change-point as necessary.

Hartland, Gelly, Baskiotis, Teytaud, and Sebag [73] propose an algorithm called Adapt-EvE based on the UCB-Tuned algorithm [15]. Adapt-EvE uses the frequentist Page-Hinckley test to identify change-points. Upon detection of a change-point, Adapt-EvE treats the problem as a meta-bandit problem. That is, a second layer of bandit optimization is instituted with two arms: (1) continues using the learned data and (2) restarts the UCB-Tuned algorithm from scratch. This meta-bandit forms a hierarchical strategy that can be expected to efficiently evaluate the *cost* in regret of each detected change. This technique was the winning technique in the PASCAL Exploration vs. Exploitation challenge in 2006 [79] demonstrating its ability to handle both drifting and change-point type bandits.

Kocsis and Szepesvári [95] present a variant of UCB-Tuned called DiscountedUCB which applies a continuous discount factor to the estimates in time. Garivier and Moulines [63] introduce Sliding Window UCB (SW-UCB) parameterized by a window length and show it performs similarly to DiscountedUCB contingent on appropriately selected parameterizations.

Mellor and Shapiro [108] present an online Bayesian change-point detection process for *switching* (discrete change) bandits with constant *switching rate* – the frequency with which the distributions change – in the contexts where switching occurs globally or per-arm and when switching rates are known or must be inferred. Their algorithm is probability matching based, but, as presented does not support contextual variables. Further, their technique addresses a bandit with switching behavior, rather than drifting behavior as explored in this work.

3.4.1 A Short Review of Stochastic Drift

In time-series analysis, *stochastic drift* is used to refer to two broad classes of non-stationarity in the population parameter being estimated: (1) cyclical or model-able drift that arise because of model misspecification and (2) the random component. Often it is possible to *detrend* non-stationary data by fitting a model that includes time as a parameter. Where the function of time is well-formed and appropriate for statistical modelling, a *trend stationary* model can be found with this detrending process. For some models, detrending is not sufficient to make a process stationary, but, sometimes *difference stationary* models can be fit, where the differences between values in time Y_t and Y_{t-n} can be represented as a well-formed function appropriate for statistical modelling.

Difference stationary models are represented with autoregressive models. The generalized representation of the simple autoregressive model is referred to as $AR(n)$ where n is the number of time steps back the current value maintains a dependency upon,

$$AR(n): \;\; Y_t = \alpha_0 + \alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} + \cdots + \alpha_n Y_{t-n} + \epsilon_t, \tag{3.2}$$

where ϵ_t is the error term with the normal characteristics of zero mean ($E[\epsilon_t] = 0$), variance σ^2 and independence across times ($E[\epsilon_t \epsilon_s] = 0, \forall t \in \{t{\neq}s\}$) after fitting the autoregressive correlations. If these two detrending strategies are not sufficient to make a given process stationary, more complex filters such as a band-pass or Hodrick-Prescott filter may be applied.

Generalized Linear Bandits

Filippi, Cappe, Garivier, and Szepesvári [61] use generalized linear models (GLMs) for bandit analysis, extending the work of Dani et al. [51] and Rusmevichientong and Tsitsiklis [128] to utilize the UCB strategy of Auer, Cesa-Bianchi, and Fischer [15] and proving (high-probability) pseudo-regret bounds under certain assumptions about the link function and reward distributions. In some sense, our work extends the Filippi et al. result to an experimental analysis within the non-stationary case, as well as introducing a Thompson sampling based strategy for integrating GLMs, rather than the UCB technique.

3.4.2 Overview of the Approach

The general technique we experiment with is to fit a regression model of varying form to the data and then to utilize the technique of optimistic Thompson sampling to predict arm payoffs in the next iteration of the algorithm. We explore and compare two primary models, the autoregressive, time-detrended approach and the weighted least squares approach for handling non-stationarities with a regression framework.

Autoregression and Detrending

Formally, we fit a model

$$Y_{t,i} = \alpha_t + \mathrm{AR}_i(p) + \mathrm{Trend}_i(t) + A_{t,i} + \epsilon_{t,i}, \tag{3.3}$$

where $\mathrm{Trend}(t)$ is a function representing the expected time trend, $\mathrm{AR}(p)$ is the autoregressive term of order p and $Y_{t,i}$ is the expected reward for arm i at time t. In general, trends and AR terms must be considered on a per-arm basis (equivalently, as an interaction effect on the A matrix) as it is the *change between arms* that determines a meaningful nonstationarity for the sake of decision making. In practice, this model is generally fit as a model of Y_t with binary ("dummy") variables

$A_{t,i}$ and relevant interaction terms indicating which arm is detected. In our experimental results, we explore how variations (especially *overspecification* of the functional form) in the "correctness" of the selection of Trend(t) affect the overall results. This model, fit with the ordinary least squares technique, the ridge regression technique [145] or the Bayesian conjugate prior technique, returns an estimated set of time-detrended, plausibly stationary[5] coefficients $\hat{\beta}$ and estimates of their standard errors $\widehat{SE}(\hat{\beta})$. This model can be readily extended to contain any contextual variables, such as demographic information about the user (in the web optimization context) or grouping criteria on the arms to improve the learning rate.

Combined, we follow in standard experiment design terminology and call the terms in our model α, $\text{AR}_i(p)$, $\text{Trend}_i(t)$, and $A_{t,i}$ the *design matrix* and refer to it as X.

Penalized Weighted Least Squares

The *weighted least squares* (WLS) process introduces a multiplicative weighting of "reliability" for each observation, resulting in a technique which minimizes the reliability-adjusted squared errors. In the multi-armed bandit context with drifting arms (without any *a priori* knowledge of the functional form of the drift), the weights are set to the inverse of their recency, indicating that at each time step t, older data provides a less reliable estimate of the current state.

Intuitively, weighted least squares provides a simple, well-explored, highly tractable technique to discount the confidence of old data, increasing predictive uncertainty as time progresses. This is a desirable quality within the context of restless bandits as it appropriately accounts for the growing predictive uncertainty of old observations.

Formally, the weighted least squares procedure picks $\hat{\beta}$, coefficients on a set of variables, X, called the independent variables (or regressors), according to the equation $\hat{\beta} = (X^T W X)^{-1}(X^T W y)$ where W is the matrix of weights and y is the rewards as observed (or, in general, the regressand). Standard errors of the coefficients are also computed, producing an estimate of the standard deviation of our estimators.

To apply the weighted least squares procedure, we extend LinTS (presented in the prior section), following in the work of Pavlidis, Tasoulis, and Hand [119] which uses a standard linear regression to compute the estimates of each arm and the work of the LinUCB algorithm [101] which applies a non-weighted penalized linear regression to compute estimates of the payoff for each arm. As we are *a priori* uncertain about the functional form of the non-stationarity in our bandit arms, we experiment with a variety of time weighting techniques – logarithmic, with varying scale and base; linear, with varying polynomials; exponential, with varying coefficients; and sinusoidal – demonstrating the generality of this technique. In all cases we strictly decrease the weight of a sample as it becomes further in time from our current prediction time. When additional information about the form of non-stationarity is available, weights can be specified appropriately to reduce the

[5] As long as the detrending process successfully removed the non-stationarity.

predictive uncertainty.

3.4.3 Simulation Environment

To test our combined strategies and produce objective comparisons, we utilize the synthetic simulator described in Section 3.1 with a wide variety of "true worlds" (unobserved to the agent) including arm distribution type and parameters, arm count, and drift type from a set of functional forms including random walk, exponential random walk, logarithmic, linear (in varying degree), exponential and periodic drift (sinusoidal over varying periods). Each form of drift is parameterized by a randomly drawn real number constrained to be within the same order of magnitude as the arm payoffs in its simulation world which determines the scale of the parameterization.

We present the combined algorithm, parameterized in degrees of autoregression, detrending and functional form of our weighted least squares discounting process in pseudocode in Figure 3.1. The first n, the number of model terms, iterations must be performed using another method (uniformly at random, in our case) to provide enough degrees of freedom to fit the regression model.

3.4.4 Experimental Results

Specific details of the optimistic sampling procedure are shown in the next section when we discuss techniques for sampling from an arbitrary distribution. In this case, as normality is assumed in the model, we can sample very easily.

In the results presented, we omit ϵ-greedy, UCB1, DiscountedUCB and others as they were strictly outperformed by UCB-Tuned or SW-UCB for all parameter choices tested. Across all true worlds, we find in general that a detrending term congruent with the *true drift* form (e.g. *linear detrend* in the linear drift quadrant of Figure **??**) outperforms all other strategies in the long run, producing a *zero-regret strategy* [152] for restless bandits where the functional form of restlessness is known. Similarly, we find that utilizing a weighting function which closely approximates the true drift performs well in most cases. Surprisingly, we find that linear detrending is an effective technique for handling the *random walk*, a result that is robust to variations in the step type and scale of the random walk. Unintuitively, WLS techniques also perform strongly even in the case when there is no drift.

In these experiments, we find no convincing evidence for a general application for detrending in polynomial degree greater than one or autoregression of any level in our model. Both autoregression and higher degree polynomials strictly reduce regret *if* the true world trend is autoregressive or determined, even partially, by the chosen form. We find the linear weighted least squares technique (weights set to the inverse of t) to be the most robust technique over all experiments, suggesting it is the strongest technique in the case of no *a priori* information on the form of drift: having the lowest mean total regret (20.8), lowest standard deviation across all drift types (11.8) and the lowest 75th (worst-) percentile regret (26.6).

Data:
λ: the penalty/regularization factor for the ridge regression $w(t)$: a function which defines the weighting strategy
Ω: a function which determines whether we should play another round

Result: A discounted (in time) bandit algorithm proportional to the weighting strategy.

```
$X \leftarrow y \leftarrow W \leftarrow []$;
$t \leftarrow 0$;
while $\Omega$ do
    // Generate the weighting matrix according to our function $w(t)$.
    $W[t] \leftarrow w(t)$;

    // Get the model from the penalized weighted least squares (WLS)
       subroutine.
    $\hat{\beta} \leftarrow (X^T W X + \lambda \mathbb{I})^{-1}(X^T W y)$;
    $s^2 \leftarrow (y - \hat{\beta} X)^2 / n$ ;
    // Compute the errors (estimated variance) necessary to perform the
       sampling procedure.
    $\widehat{\mathrm{Var}}(\hat{\beta}) \leftarrow \mathrm{diag}[s^2(X^T W X + \lambda \mathbb{I})^{-1}]$;
    // End penalized WLS subroutine.

    $\hat{r}_* \leftarrow -\infty$;
    $S_t \leftarrow nil$;
    for each arm $i$ do
        // Draw from the estimated distribution for this arm.
        $\hat{r}_{\text{test}} \leftarrow$ draw optimistic$(\mathrm{N}(\sum_i(\hat{\beta}_i \cdot X_{i,t}), \sum_i (\widehat{\mathrm{Var}}(\hat{\beta}_i) \cdot X_{i,t}^2)))$;
        // Maintain a Thompson-like estimate of the best arm.
        if $\hat{r}_{\text{test}} > \hat{r}_*$ then
            $\hat{r}_* \leftarrow \hat{r}_{\text{test}}$;
            $S_t \leftarrow i$;
        end
    end
    reward $\leftarrow$ play$(S_t)$;
    extend $X$, the design (history) matrix;
    append reward to rewards history $y$;
    $t \leftarrow t + 1$;
end
```

Figure 3.1: Pseudocode of combined algorithm.

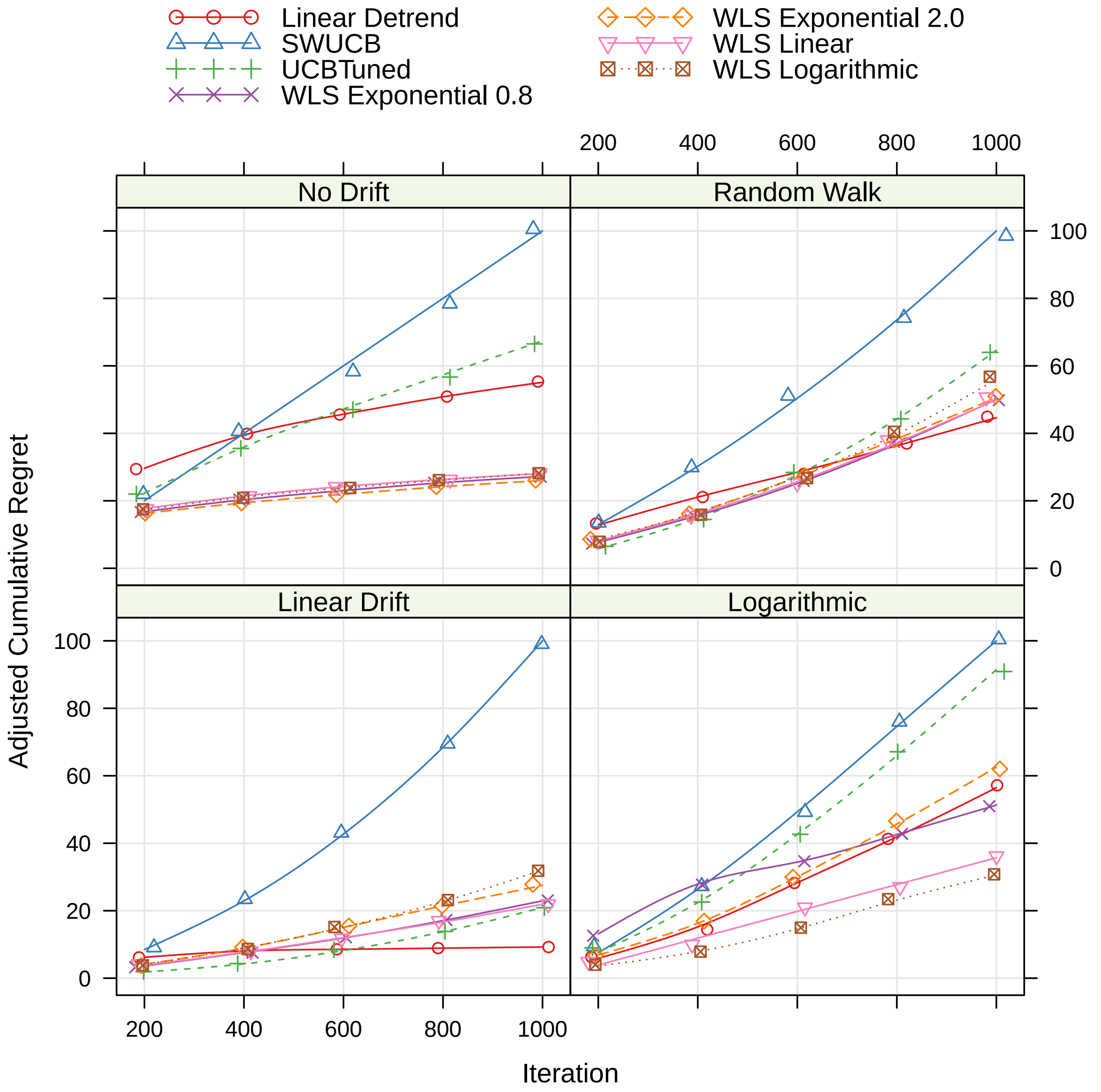

Figure 3.2: Adjusted average cumulative regret of selected algorithms over 1,000 replicates of all worlds and true drift forms.

3.5 Statistical Regret for Applications

All definitions of regret given in Chapter 2 are dependent on access to an *oracle* which *ex-ante* both knows the parameters of the underlying distributions (e.g., at a minimum, know the best mean) and fully enumerates any non-stationarities or contextual factors. We propose an oracle-free regret interval which can be used in real world applications to quantify differences in bandit policies *a posteriori*.

Statistical regret is an important contribution in the case of real world computation. In applied bandits, no oracle is available[6], but there is still a need to compare policies. If an oracle were available, a nontrivial bandit policy would be unnecessary, as the ideal arm would be fully known. In the event all arms' rewards are available *after* each play, the problem is better represented as general reinforcement learning than a multi-armed bandit. A simpler technique where samples are divided (equally, in a traditional experiment design) across multiple bandit algorithms is available under certain distributional assumptions, however in the case of non-stationarity in time *and* per-play regret reduction, this may not accurately capture the difference in policies.

Our measure, presented in the next subsection, is derived from intuitions of the γ-confidence interval and the expected-expected and expected-payoff regrets described in the prior section, and provides a meaningful applied measure to compare algorithms under certain assumptions. We further suggest a plausible extension to statistical regret using the bootstrap [56] to compute confidence bounds in effort to tighten the bounds for a given confidence level and produce a distribution-free estimator.

3.5.1 Traditional Parametric Statistical Regret

To compute parametric statistical regret, begin by enumerating your set of *predicted* arm mean distributions from observations up to time t, $\tilde{X}_i$. Then, assume a confidence interval exists of the form $(L_{t,\gamma}(\tilde{X}_i), U_{t,\gamma}(\tilde{X}_i))$ such that,

$$\Pr(L_{t,\gamma}(\tilde{X}_i) < \tilde{X}_i < U_{t,\gamma}(\tilde{X}_i)) = \gamma.$$

A confidence interval of the form above exists if and only if there were sufficient observations to accurately fit the assumed parametric distribution. From these values, we can compute the statistical best- and worst-case scenarios. Intuitively, for each play in our recorded history, we look at the best upper and best lower bound and accrue regret accordingly. Formally, the total stochastic statistical regret up to time t is an interval,

[6] It is conceivable that some non-simulation applications exist in which only regret is revealed after each round, in which case traditional regret measures are well-defined without an out-of-band oracle.

$$\tilde{R}^P_\gamma = \left(\sum_{j=0}^{t}(\max_i L_{t,\gamma}(\tilde{X}_i) - x_{S_j}), \sum_{j=0}^{t}(\max_i U_{t,\gamma}(\tilde{X}_i) - x_{S_j})\right). \tag{3.4}$$

This produces a regret interval with properties similar to expected-payoff regret; it is possible to accrue negative regret and the resulting output is a random variable in both S_j and the X distributions' parameters θ. We can eliminate the stochasticity on the parameters θ by eliminating the actual received payoff from our consideration and instead using the empirical mean (computed from all data available at the time we compute, t) on the chosen arm in iteration j, ($\bar{X}_{S_j}$) with

$$\tilde{R}^E_\gamma = \left(\sum_{j=0}^{t}(\max_i L_{t,\gamma}(\tilde{X}_i) - \bar{X}_{S_j})), \sum_{j=0}^{t}(\max_i U_{t,\gamma}(\tilde{X}_i) - \bar{X}_{S_j})\right). \tag{3.5}$$

By utilizing the estimates at time t to evaluate the plays at times $j = 0, ..., t$ we utilize the *best information available* to judge actions chosen by the policy prior to that information being available; by utilizing the *bounds* per-arm we acknowledge that there is an uncertainty (for many bandit policies, a *large* uncertainty) to the knowledge we have. When considering statistical regret, it is important to note that unlike in a traditional experiment, in a bandit experiment a tighter bound on statistical regret (somewhat equivalently, a tighter bound on the confidence intervals) is *not* the objective to be optimized.

We experiment extensively on varying world formats, horizon lengths and confidence bound widths and find that stochastic statistical regret accurately tracks the true expected-payoff regret with sufficiently tight bounds to enforce the γ confidence level. As $\gamma \to 1.0$, the required sample size per arm tends to ∞.

Statistical regret combines properties of the traditional experiment with the multi-armed bandit to produce a powerful real world diagnostic and evaluation tool. Post-hoc analyses in an oracle-free, multi-armed bandit context are made possible by such a tool. In particular, we conjecture that statistical regret will be a useful diagnostic for aiding in the applied calibration of common bandit policy parameters such as discount rate or egalitarianism.

3.6 Simple Efficient Sampling

When performing Thompson sampling, we always need a method to actually produce a sample from a distribution. In this section, we deal with how to perform this sampling in a way that is efficient in two dimensions: computationally efficient, as in, to not take more computational instructions (or time) than necessary and statistically accurate, as in, to accurately capture the intended sampling distribution.

There are two interacting rationales for this work observed in the literature. A number of papers produce work which samples from an optimistic surrogate distribution in an iterated Monte Carlo-like strategy, sampling from the initial (non-optimistic) distribution until the intended results are found, or sampling from a black box to produce an estimate of the underlying distribution. This strategy is computationally inefficient, and we show that it can be done in a strictly more efficient way for most distributions. The other reason is the statistical inaccuracy – optimism in (empirical) sampling research is often left undefined or as an exercise to the implementer. This has resulted in work which produces definitions of optimism with likely unintended behaviors, such as placing half of the samples at exactly the mean.

3.6.1 Simple Efficient Symmetric Sampling (SESS)

For symmetric distributions, finding a technique for sampling from the optimistic distribution efficiently (both in the sense of *computational* efficiency and *statistical* accuracy) is simple. The process is: (1) center the distribution at zero, (2) get a sample, s, from the centered distribution, (3) add the measure of centrality (*e.g.,* mean or median) of the uncentered distribution to $|s|$. For example, in the normal distribution, parameterized on μ and σ^2, we can draw each sample according to

$$s_i^A = \mu_i + |\mathrm{N}(0, \sigma_i^2)|. \tag{3.6}$$

3.6.2 Efficient Non-Symmetric Sampling (ENSS)

For a general distribution, P_i with expectation μ_i, (including non-symmetric distributions), at least one paper presents a statistically *inefficient* technique (in the sense of not completely achieving the optimistic sampling process we intended) which produces a sample,

$$s_i^B = \max\{\mu_i, P_i(\cdot)\}. \tag{3.7}$$

As an example of the inefficiency of this technique, we take the case of a symmetric distribution in which this produces a sample that is 50% biased to the mean (every $P(\cdot) < \mu$ is selected at exactly μ). To perform (both statistically and computationally) efficient sampling in the non-symmetric case, we require a quantile function $Q(p) = \inf\{x \in \mathbb{R} : p \leq F(x)\}$ where F is the cumulative distribution function. When this is available, we can sample from the optimistic distribution in three steps.

1. Find the quantile, v, of the measure used to bound optimism (for example, for *median-optimism*, set $v = 0.5$).

2. Draw a random value from a uniform distribution $t \sim \mathrm{UNIF}(v, 1)$ indicating the quantile of the desired value.

3. Find the value in our original distribution at that point, $s_i^C = Q(t)$, this is our sample.

For the most common distributions used in multi-armed bandits, the quantile function required to determine v and cumulative distribution function are readily available. For example, the normal, Student, beta, and gamma distributions' quantile functions are explored in Steinbrecher and Shaw [138] and available to programmers in the R programming language as `qnorm`, `qt`, `qbeta` and `qgamma` respectively.

3.6.3 A Short Background in Nonparametric Sampling

Bootstrap Thompson Sampling

Eckles and Kaptein [55] present an implementation of Bootstrap Thompson Sampling (BTS) which produces a scalable, robust approach to Thompson sampling. We show in this section that their implementation can be modified to support an arbitrarily scaled form of optimism and a completely nonparametric approach at minimal cost, at least for some subset of the problem space. We then show that the sampling strategies shown in the Sampling Uncertainty section can be applied here to avoid a number of poor edge-case performances and essentially control the *risk* of such a policy. Finally, we expand the implementation to a categorical-contextual policy which subsets the sampling space into categories to implement a refinement of the model.

The Eckles and Kaptein (2014) Model

The initial implementation of BTS involves selecting a parameter J which simultaneously controls the computational complexity and relative greediness of the model. Upon receipt of each reward r_i, BTS trains J parametric models on the assumed underlying distribution by considering the reward in each model with probability one-half. To select the next arm to play, BTS, on a per arm basis, chooses one of the J replicates and uses the expected value of that model to predict an empirical mean payoff. In the Thompson sampling style, the highest of those estimated payoffs is selected as the action for this iterate. As the empirical mean is deterministic across the replicates[7], if J is too small, the decision becomes fundamentally greedy, choosing to only play the best empirical arm prematurely. This technique works well, showing performance competitive with the traditional model and even exceeding it in the case of heteroskedastic errors.

3.6.4 A Simple Efficient Nonparametric Sampler

Our goal sets out to produce a nonparametric sampler and simultaneously maintain the benefit we have seen thus far with respect to optimism in the face of uncertainty. To do so, we first propose a simple non-optimistic sampler and then augment that sampler by adding optimism, producing

[7] Computing the expectation, rather than sampling from the subdistributions, gives us an efficient sampler in the bootstrap sampling distribution θ, but does not maintain the "uncertainty awareness" property in the individual distributions. A sufficiently large J returns the uncertainty awareness property of Thompson sampling.

the simple optimistic sampler we call SOS. In the section that follows, we will present a simple discretization observation that allows this model to be extended to a relatively small number of categorical contextual variables easily.

Simple Sampler

In the simple sampler, we maintain a history of observed rewards per arm $\tilde{x}_{i,t}$ and sample one value ($\hat{r}_{test} \in \mathbb{R}$) uniformly from the history for each arm i at each iterate t as presented in Figure 3.3. This $\hat{r}_{test}$ value represents a single payoff from the history of the arm; of these samples, the maximum value seen (from the K values, each representing a single arm) determines the arm to be played in the next iterate with ties broken randomly. As the distribution of our observed history limits to the true population distribution, this technique captures the true underlying distribution of the arms rapidly at the cost of some low sample size misbehavior. In experimentation, if the assumed parametric distribution is *correct* this model performs similarly to the standard unbiased Thompson sampler fitted with either an assumed beta-binomial (true binomial) or the least squares procedure (true normal). When modelling error is introduced to simulate real world misspecifications, we find that this model greatly outperforms the standard Thompson sampler even in the early process. Further, if the early sample size behavior is deemed high risk in the application, the replication strategies described earlier can be applied to change the nature of the sampled distribution.

This technique draws from the same assumption that Thompson sampling itself draws from: the concept that sampling a single time from each distribution is enough to approximate the *probability of optimality* distribution. As such, it is expected to have similar considerations. We consider first optimism, and then the replication strategy questions answered in the previous section for traditional Thompson sampling.

Introducing Optimism to the Simple Sampler

Introducing optimism in the simple sampler requires maintaining a *payoff-ordered* list of historical rewards. In order to produce a sampler that is both computationally reasonable and robust to the binomial distribution (or any similar distribution where one distribution may be a clear winner (or tied) in the top half but not in expectation), we propose 3 additional optional parameters: a maximum sample size ($B = 100$), the optimism cutoff ($\gamma = 0.5$), indicating the percentile (fraction of the observed results) that should be discarded to get our optimistic surrogate distribution and a minimum sample size ($\kappa = 20$). Despite this larger parameterization, we show that even eliminating these parameters (that is, setting $B = \infty, \kappa = 0$ and using the traditional mean cutoff for $\gamma = Q(F(\mu))$ where Q is the quantile function and F is the empirical CDF) we still have a sampler that performs well in all but the most degenerate distributions, *even* in the large sample binomial case.

From the sorted list of historical payoffs, we sample uniformly from the top $n = \min(\kappa, N \cdot \gamma)$ observations. Upon receiving a reward, we append it to our list of rewards. If the total length of

```
Data:
$S_i$:   the result of playing $n > 1$ purely random seed rounds for each arm $i$
$\Omega$:   a function which determines whether we should play another round
Result: A simple distribution-free bandit sampling strategy.
initialize $\forall i \;\; \tilde{x}_i \leftarrow S_i$ ;
while ($\Omega$) do
    $\hat{r}_* \leftarrow -\infty$;
    $S_t \leftarrow []$;
    // Draw once from each arm history and play the best draw, $S_t$.
    for each arm $i$ do
        $\hat{r}_{\text{test}} \leftarrow$ draw uniformly($\tilde{x}_i$);
        if $\hat{r}_{\text{test}} > \hat{r}_*$ then
            $\hat{r}_* \leftarrow \hat{r}_{\text{test}}$;
            $S_t \leftarrow [i]$;
        else if $\hat{r}_{\text{test}} == \hat{r}_*$ then
            $S_t$.append($i$);
        end
    end
    // Break ties randomly and play the selected arm.
    $S_t^* \leftarrow$ draw uniformly($S_t$);
    reward $\leftarrow$ play($S_t^*$);
    ($\tilde{x}_{S_t^*}$).append(reward);
end
```

Figure 3.3: The Simple Nonparametric Sampler

the list of rewards N is more than our maximum size B, we uniformly randomly select one to drop from the sample. This can be trivially extended to the J-replicate system of the original Eckles and Kaptein (2014) sampler but we find no benefit in doing so when B is sufficiently large.

This technique has a major advantage in that it is distribution-free. The standard optimistic/nonoptimistic sampler performs very poorly in the case the model is misspecified. As an example, we have computed comparisons for SOS, Simple Sampler and the traditional parametric model computed with the normal distribution sampler and computed with a beta-binomial sampler as described in Chapelle and Li [42]. As expected, in the event the rewards are congruent with the selected sampler, the performance is not much different (although the implementation is arguably simpler under SOS) but in the event the rewards are incongruent with the selected sampler (that is, a beta-binomial to approximate a normal reward, or a normal to predict a beta-binomial) we find that the SOS and Simple Sampler performance is significantly improved. This is essential for applications where the true payoff distribution may not be generated from a known distribution.

Data:
S_i: the result of playing $n > 1$ purely random seed rounds for each arm i
Ω: a function which determines whether we should play another round

B: the maximum size of our sample history (positive integer or infinity)
γ: the degree of optimism in $[0, 1)$, $\gamma = 0$ is non-optimistic, $\gamma = 0.5$ provides an approximation to *median-optimism*
κ: the minimum size of our sample below which we retain the whole sample

Result: A simple distribution-free *optimistic* bandit sampling strategy.

```
initialize $\forall i$ $\tilde{x}_i \leftarrow$ sort($S_i$) ;
initialize $\forall i$ $N_i \leftarrow$ count($S_i$) ;
while ($\Omega$) do
    $\hat{r}_* \leftarrow -\infty$;
    $S_t \leftarrow []$;
    for each arm $i$ do
        // Sample from a $\gamma$-optimistic surrogate distribution that is at least $\kappa$ in size (if $\kappa$
           samples exist).
        if $(N_i - \gamma N_i) < \kappa$ then
            $\tilde{x}_i^{\text{optimistic}} \leftarrow$ subset($\tilde{x}_i$, $N_i - \kappa$, $N_i$);
        else
            $\tilde{x}_i^{\text{optimistic}} \leftarrow$ subset($\tilde{x}_i$, $\gamma N_i$, $N_i$) ;
        end
        $\hat{r}_{\text{test}} \leftarrow$ draw uniformly($\tilde{x}_i^{\text{optimistic}}$) ;
        if $\hat{r}_{\text{test}} > \hat{r}_*$ then
            $\hat{r}_* \leftarrow \hat{r}_{\text{test}}$;
            $S_t \leftarrow [i]$;
        else if $\hat{r}_{\text{test}} == \hat{r}_*$ then
            $S_t$.append($i$);
        end
    end
    $S_t^* =$ draw uniformly($S_t$);
    reward $\leftarrow$ play($S_t^*$);
    ($\tilde{x}_{S_t^*}$).insert sorted(reward);
    $N_{S_t^*} \leftarrow N_{S_t^*} + 1$;
    if $N_{S_t^*} > B$ then
        // Randomly remove a reward above $B$ from the samples.
        $\tilde{x}_{S_t^*} \leftarrow$ delete(draw uniformly($\tilde{x}_{S_t^*}$));
        $N_{S_t^*} \leftarrow N_{S_t^*} - 1$;
    end
end
```

Figure 3.4: The Fully Parameterized Simple Optimistic Sampler (SOS)

Table 3.4: The robustness and performance of the distribution-free sampler compared to the traditional parametric sampler.

Sampling Policy	**Distribution**	**Average $\bar{R}^E$**
Traditional Unbiased	Incorrect	0.500
Traditional Unbiased	Correct	0.391
Traditional Optimistic	Incorrect	0.498
Traditional Optimistic	Correct	0.341
Simple Sampler	–	0.368
Simple Optimistic Sampler	–	**0.323**

We see in Table 3.4 the simple samplers produce results comparable to the traditional parametric sampler in the case of correct specification *without any knowledge of the underlying distribution*, producing a performant real-world sampling technique for arbitrary applications. Further, we see that the traditional technique can perform very poorly (approximately 30% worse) under misspecification providing more evidence for using the robust sampling technique.

Experiments in Replication Strategies with SOS

We reproduce the multiple-sampling replication experiments from Section 3.3 here in the context of SOS. This is significant, as the small sample performance of unreplicated SOS is not obviously performant and there are plausible concerns of managing the distribution around optimism in certain degenerate cases. We find that neither of these concerns are a problem in expected regret or cumulative reward. For comparison, we provide the results from the traditional and traditional optimistic Thompson sampler in both matching (correct) and non-matching (incorrect) parametric models. We present the results of the replication strategy experiments in Table 3.5.

3.6.5 Using Categorical Contextual Variables in SOS

In many contexts, contextual variables are fundamental to the application of bandit methods. Historically, methods for computing contextual bandits have been computationally expensive. The independence of the sampling process in SOS allows us to produce a contextual bandit method for cases of a small number of independent categorical variables by subsetting the data to those records which belong to the appropriate subset. This method has the advantage of being easily and efficiently implemented within a relational database context but suffers from the curse of dimensionality; there are two distinct considerations in handling that concern: (1) for a small number of classes (even misspecified classes), it performs well while remaining easily scalable and (2) we can introduce a new parameter $\varkappa$ which sets a minimum size within a class. If the minimum size is not reached, a random class is dropped from the subset until the size is met. Our fully parameterized algorithm for categorical contextual bandits with SOS is detailed in Figure 3.5.

Table 3.5: Replication strategy experiments for SOS and Simple Sampler

#	**Sampling Policy**	**Replication Strategy**	k	**Average** $\bar{R}^E$	**SD**	$\sum_t x_{S_t,t}$	**SD**
49	Correct OTS	None	-	0.341200	0.474160	2.683923	2.051690
50	Incorrect OTS	None	-	0.492600	0.499995	2.512447	2.060076
51	**SOS**	**None**	**-**	**0.326600**	**0.469016**	**2.650144**	**2.042715**
52	SOS	Least Deviation	2	0.337000	0.472732	2.678071	2.049964
53	SOS	Least Deviation	3	0.338000	0.473076	2.629016	2.069614
54	SOS	Least Deviation	4	0.347000	0.476063	2.662008	2.058679
55	SOS	Least Deviation	5	0.328800	0.469824	2.662791	2.034599
56	SOS	Least Deviation	10	0.331200	0.470692	2.636481	2.027811
57	SOS	Least Deviation	30	0.331000	0.470620	2.667166	2.028424
58	SOS	Sampled Mean	2	0.330600	0.470476	2.711309	2.067525
59	SOS	Sampled Mean	3	0.336600	0.472594	2.699488	2.047436
60	SOS	Sampled Mean	4	0.324600	0.468272	2.656813	2.039116
61	SOS	Sampled Mean	5	0.335800	0.472317	2.702142	2.101192
62	SOS	Sampled Mean	10	0.332200	0.471049	2.711977	2.042467
63	SOS	Sampled Mean	30	0.330600	0.470476	2.685058	2.043417
64	SOS	Most Optimistic	2	0.341800	0.474360	2.653896	2.089214
65	SOS	Most Optimistic	3	0.331200	0.470692	2.676421	2.095322
66	SOS	Most Optimistic	4	0.327800	0.469458	2.646591	2.080160
67	SOS	Most Optimistic	5	0.330200	0.470332	2.698665	2.042973
68	SOS	Most Optimistic	10	0.336400	0.472525	2.695991	2.025342
69	SOS	Most Optimistic	30	0.336800	0.472663	2.715561	2.063079
70	SOS	Least Optimistic	2	0.343400	0.474891	2.617706	2.048963
71	SOS	Least Optimistic	3	0.332800	0.471263	2.722110	2.055616
72	SOS	Least Optimistic	4	0.338800	0.473349	2.665208	2.036815
73	SOS	Least Optimistic	5	0.337800	0.473007	2.672261	2.055658
74	SOS	Least Optimistic	10	0.347800	0.476320	2.644284	2.037160
75	SOS	Least Optimistic	30	0.327800	0.469458	2.699044	2.043100
76	Correct TS	None	-	0.391200	0.488068	2.640098	2.100612
77	Incorrect TS	None	-	0.496200	0.500036	2.497952	2.074799
78	Simple Sampler	None	-	0.368600	0.482473	2.643983	2.046037
79	Simple Sampler	Least Deviation	2	0.362800	0.480856	2.644576	2.042220
80	Simple Sampler	Least Deviation	3	0.358000	0.479460	2.622352	2.047739
81	Simple Sampler	Least Deviation	4	0.361600	0.480512	2.675990	2.048984
82	Simple Sampler	Least Deviation	5	0.362000	0.480627	2.634314	2.061458
83	Simple Sampler	Least Deviation	10	0.365600	0.481646	2.601818	2.054081
84	Simple Sampler	Least Deviation	30	0.367200	0.482090	2.620877	2.029750
85	Simple Sampler	Sampled Mean	2	0.345800	0.475676	2.696116	2.055796
86	Simple Sampler	Sampled Mean	3	0.327400	0.469312	2.675551	2.019245
87	Simple Sampler	Sampled Mean	4	0.334800	0.471968	2.647391	2.050078
88	Simple Sampler	Sampled Mean	5	0.324800	0.468347	2.671250	2.076719
89	Simple Sampler	Sampled Mean	10	0.320600	0.466754	2.680171	2.044166
90	**Simple Sampler**	**Sampled Mean**	**30**	**0.314200**	**0.464243**	**2.663312**	**2.066480**
91	Simple Sampler	Most Optimistic	2	0.358600	0.479637	2.649196	2.061342
92	Simple Sampler	Most Optimistic	3	0.355400	0.478682	2.637505	2.011785
93	Simple Sampler	Most Optimistic	4	0.340000	0.473756	2.664240	2.059272
94	Simple Sampler	Most Optimistic	5	0.330400	0.470404	2.619605	2.030020
95	Simple Sampler	Most Optimistic	10	0.343200	0.474825	2.652753	2.070380
96	Simple Sampler	Most Optimistic	30	0.325400	0.468571	2.657045	2.025362
97	Simple Sampler	Least Optimistic	2	0.363400	0.481027	2.684557	2.068898
98	Simple Sampler	Least Optimistic	3	0.344000	0.475089	2.619288	2.052010
99	Simple Sampler	Least Optimistic	4	0.337200	0.472801	2.683669	2.053468
100	Simple Sampler	Least Optimistic	5	0.334000	0.471687	2.681712	2.062158
101	Simple Sampler	Least Optimistic	10	0.337600	0.472939	2.672240	2.037189
102	Simple Sampler	Least Optimistic	30	0.341600	0.474294	2.672088	2.063609

```
Data:
$S_i$:  the result of playing $n \geq \varkappa > 1$ purely random seed rounds for each arm $i$
$\Omega$:  a function which determines whether we should play another round
$B$:  the maximum size of our sample history
$\gamma$:  the degree of optimism in $[0, 1)$
$\kappa$:  the minimum size of our sample below which we retain the whole sample
$\varkappa$:  the minimum size within a class below which we will collapse the class
Result: A distribution-free categorical-contextual optimistic bandit sampling strategy.

initialize $\forall i \;\; \tilde{x}_i \leftarrow$ sort($S_i$) ;
initialize $\forall i \;\; N_i \leftarrow$ count($S_i$) ;
while ($\Omega$) do
    $\hat{r}_* \leftarrow -\infty$;
    $S_t \leftarrow []$;
    $C_t^0 \leftarrow C_t \leftarrow$ get full context vector;
    for each arm $i$ do
        repeat
            // Subset observed rewards to only the matching context category.  If insufficient
               values are available, remove a context category randomly and repeat (minimum
               $\varkappa$).

            $\tilde{x}_i^{\text{match}} \leftarrow$ subset($\tilde{x}_i$, $C_t \in$ context($\tilde{x}_i$));
            $C_t \leftarrow$ delete(draw uniformly($C_t$)) ;
        until count($\tilde{x}_i^{match}$) $> \varkappa$;

        $N_i^{\text{match}} \leftarrow$ count($\tilde{x}_i^{\text{match}}$);
        if ($N_i^{\text{match}} - \gamma N_i^{\text{match}}$) $< \kappa$ then
            $\tilde{x}_i^{\text{matching optimistic}} \leftarrow$ subset($\tilde{x}_i^{\text{match}}$, $N_i^{\text{match}} - \kappa$, $N_i^{\text{match}}$);
        else
            $\tilde{x}_i^{\text{matching optimistic}} \leftarrow$ subset($\tilde{x}_i^{\text{match}}$, $\gamma N_i^{\text{match}}$, $N_i^{\text{match}}$) ;
        end

        $\hat{r}_{\text{test}} \leftarrow$ draw uniformly($\tilde{x}_i^{\text{matching optimistic}}$) ;
        if  $\hat{r}_{\text{test}} > \hat{r}_*$ then
            $\hat{r}_* \leftarrow \hat{r}_{\text{test}}$;
            $S_t \leftarrow [i]$;
        end
        else if $\hat{r}_{\text{test}} == \hat{r}_*$ then
            $S_t$.append($i$);
        end
    end
    $S_t^* \leftarrow$ draw uniformly($S_t$);
    reward $\leftarrow$ play($S_t^*$, $C_t^0$);
    ($\tilde{x}_{S_t^*}$).insert sorted by reward(reward, $C_t^0$);
    $N_{S_t^*} \leftarrow N_{S_t^*} + 1$;
    if $N_{S_t^*} > B$ then
        $\tilde{x}_{S_t^*} \leftarrow$ delete(draw uniformly($\tilde{x}_{S_t^*}$));
        $N_{S_t^*} \leftarrow N_{S_t^*} - 1$;
    end
end
```

Figure 3.5: The Categorical-Contextual Simple Optimistic Bootstrap Sampler (SOS)

3.7 Summary

Our research presented in this section draws a unifying story on the nature of optimism and contextual sampling strategies – answering a set of questions which arise in the implementation of an advertising system driven by a simple linear model Thompson sampler. In particular, we have presented a simulation platform which provides the infrastructure we use to run tests, compare algorithms across a wealth of dimensions and assumptions in a repeatable and tractable way and answer questions as they arise. Then, we provided a definition of a new technique called LinTS which provides a linear model-driven Thompson sampler, then we both answer implementation questions and show how it can be extended to the non-stationary (drifting) case in a reliable way. In this process, we identify a number of interesting observations on the nature of optimism itself - providing evidence for an open question regarding the reason for the effectiveness and choice of specific type of *optimism in the face of uncertainty* in exploratory problems. Finally, we present a set of considerations related to the nature of performing the sampling: efficient implementations and a nonparametric, distribution-free model of a Thompson sampler which can support categorical covariates and performs similarly to the traditional parametric sampler while being robust to misspecification across model selection (in fact, requiring no model selection) in a way the traditional sampler is not.

Chapter 4

Conclusions

4.1 Summary

In this work we have explored some of the factors necessary to produce an efficient toolkit for online experiment design, especially with regard to application to the web optimization problem presented in the introduction. Our work takes a statistically-motivated perspective, relying on the fundamental tools of statistical analysis, from confidence intervals to regression analysis, while both explicitly and implicitly acknowledging the differences of the exploratory tradeoff inherent in bandits from traditional experiments. This perspective deviates slightly from some of the other perspectives presented in the literature, in a way which motivates the novelty of many of our results, especially with regard to the taxonomy of regret presented in the background.

We first presented an exploration of the type of problems and confounding factors typically considered in multi-armed bandits, an exploratory look at the work that has preceeded this and the first in-depth discussion of the considerations, especially the many definitions of regret, necessary to be understood when analyzing bandit problems. Our background presentation focused heavily on breadth and understanding for the sake of implementation, while still providing knowledge of the existing bounds and understanding the positional "efficiency," in the considerations for which efficiency applies, for each algorithm within the literature.

We have considered a number of theoretical and implementation questions and provided evidence to the interpretation of concepts such as regret and optimism. More so, we presented efficient algorithms for both parametric and nonparametric sampling and the simultaneous handling of the contextual and non-stationarity problems which are so fundamental to the problem of advertising and marketing decision theory.

Further, the presentation of the new metric of *statistical regret* creates the possibility of comparing algorithms *ex-post* in a real-world applied environment where there is no *a priori* oracle with which the experimenter can determine the correct answer. By treating the learning process as producing its own interval of correctness and confidence, we are able to compute measures in all environments which compare to the traditional measures that were only available in simulation. This allows a diagnostic and evaluative view of policy behavior in the real world to be taken, in a way not previously available, with an eye on tweaking and calibrating parameters for better results.

Especially interesting in our results are the outcomes with respect to the nature of empirically effective optimism, both with regard to how the prescient elimination of underlying variation in the arm distribution interacts with optimism to produce a similar benefit, and with regard to the *excessively optimistic* nature of optimism. This is likely to be an area where further research will provide a more in-depth understanding, especially with regard to the differences and relative merits between probability matching strategies and upper-confidence bound strategies in general. The surprising result with regard to the exponential discounting strategy for non-stationary bandits itself is interesting even for applications outside of the multi-armed bandit: anywhere where the tracking of an uncertain estimate of a moving variable is a requirement, this result should be considered and tested for application, indeed even if the nature of the path or functional form of the drift is unknown, it seems the exponential discounting process works well and fits cleanly within a regression-oriented framework.

4.2 Future Work

4.2.1 Theoretical Bounds in Low Sample Size Scenarios

In terms of the immediate application in a marketing and web optimization context, it is often the case that researchers will want to apply an optimization technique where the search space greatly exceeds the available trials in terms of power. One such example is the case of headline analysis for advertisements, where a researcher may wish to test hundreds of distinct headlines in a diverse set of locations where the total traffic that will observe each headline is relatively small. Techniques from the natural language processing literature and other pre-heuristics may be applied at this level to attempt to provide context (side-information) to accelerate the learning process, however, the theoretical work in low sample size cases currently leaves much to be desired.

4.2.2 Prior Elicitation

In most models, some concept of *prior knowledge* is available. This is, in fact, nearly the whole purpose of the multi-armed bandit: at each step, utilizing (and balancing) the knowledge which is accrued with the knowledge which is available to be accrued for future benefit. Capturing early prior knowledge effectively is a clear path to improving the results of these methods.

...from Experts

It is often the case that *a priori* ignorance is not the correct assumption for the type of variables we are optimizing. The enormous literature existent on the topic of eliciting priors from expert opinions and the inclusion of domain-expert knowledge in the described techniques remains an open problem.

...from Prior Experiments

While the contextual variable method of shared parameters across experiments may allow much of this prior elicitation to take place, a simple and coherent system for modifying and extending prior experiments to new variants, while retaining the valuable component of the prior data is itself an area ripe for contribution. Such a result may allow a crowd-motivated transformation of how these experiments are conducted, with shared quantified knowledge across all experiment types (especially those from the medical or academic domain) becoming the norm.

4.2.3 Risk-Aware Analysis

Risk aware methods for bandit optimization are essential to use in financial and medical domains. This comes in a multitude of forms from model and specification error (especially in the likelihood of tail events, as seen extensively in financial analysis) to the ability for the process to absorb negative results earned through exploration. In the medical domain, a negative result may represent the unsuccessful treatment or even death of a patient and depending on the stakes, these results may be entirely unacceptable. Risk aware bandits are conjectured [11] to be a substantially harder problem than the traditional models explored here, but some recent work has shown promise in this space.

The application of models from financial analysis, especially with regard to maximal drawdown and spectral measures of risk may prove promising sources of interdisciplinary integration of knowledge for the multi-armed bandit.

4.2.4 Feedback Delay

When a user accesses a website, traditionally, his request is logged in an access log for the purpose of compiling statistics. These statistics could be cross-referenced with sales or other action data to inform the multi-armed bandit process. A model of this variety is how we expect many web optimization implementations to be implemented. Unfortunately, the time gap between the first access and the point where the action such as a signup or purchase is recorded could be on the order of minutes, hours or even days, during which time more users (and as such, more experimental subjects) are processed through the system.

This *feedback delay*, the gap between when the user is presented the arm and the reward is observed, can be a source of many errors, as well as can be expected to increase regret. In the worst case, where the stream of rewards occurs after *all* available users have passed through the system, there is no learning taking place whatsoever. The interaction of feedback delay in reasonable cases and the policy selected can be large – for instance, we conjecture that a sampling-based technique will outperform any static technique (such as UCB variants) in the case of high feedback delay as they are able to consider the current uncertainty in a way that does not overtrain an individual arm. This is an area that necessitates further exploration both in quantifying the effects of feedback delay and prescribing appropriate treatments for the problem.

4.2.5 Contextual Variables

While we have presented two solutions (the simple categorical sampler and the LinTS techniques) which support contextual covariates, the contextual problem is so fundamental to rapid learning in the exploratory problem that it warrants additional attention.

Costs of Misspecification

In particular, because of small sample size effects in both the categorical sampler and the LinTS technique, it can be very expensive (in regret) to perform a *kitchen sink regression*, where all plausibly relevant variables are treated within the model. Quantifying the cost of different specifications to provide guidance to practitioners in how to select appropriate models for balancing forecasting error and model-associated fixed costs remains an important open question.

Clustering and PCA

Along the same line, where it is solely the fixed cost of early regression fits and the inability to appropriately group samples together that creates a resistance to adding model variables, it may be of value to explore clustering and the use of dimensionality reduction techniques like principle component analysis. These techniques will earlier sample fitting on small sample trials via the reduction of the feature space in a way that reduces the regret accrued in the small trial context and in the way of reducing the feedback delay associated with model computation time. Further, it is suspected that at least in some contexts, it may be possible to outperform the unclustered or high dimensional model in general, even without consideration of the fixed cost improvements.

4.2.6 Speed and Computational Complexity

In the online advertising context, millisecond-scale response times are essential for user satisfaction. Research from Google [106] finds that total load time cannot exceed 400 milliseconds before having an effect on users' well-being and positive perception of the website. Google further found that a half second increase in load time could result in a loss of up to 20% of their users for a particular request. Research from Amazon finds that every 100 milliseconds additional load time reduces sales by 1% [96]. Hundreds of other research projects both in-house and public have found many related results: response speed is an essential factor of user perception and behavior on the web; the CEO of Yahoo, Marissa Mayer is quoted as reaffirming this point "Users really respond to speed."

It is generally the case where a bandit model can be "frozen" in time for some number of iterations, and updated only when computational resources are available. The problem with this strategy is that it interacts tightly with the problems of feedback delay: in the early (smaller) sample sizes, where the model is still highly uncertain, regret may accrue very rapidly from a non-updated model. Further complicating the issue, the nature of the Internet is such that many service providers receive non-uniform flows of users, where spikes (often referred to as the *Slashdot effect* or a *flash crowd*) occur when a larger site links to the vendor, and the nature of these users is likely to be much different than the prior users. These factors complicate the adjustment in terms of the non-stationarity we have considered and require computationally efficient techniques for updating the model.

In particular, the utilization of an early-exit policy and continuous tracking type models for fitting the regression coefficients in LinTS or similar will prove invaluable in controlling the ratio between recomputation cost and the negative effects of feedback/update delay.

Bibliography

[1] Y. Abbasi-Yadkori and C. Szepesvári. Regret bounds for the adaptive control of linear quadratic systems. In *Conference on Learning Theory*, pages 1–26. Association for Computational Learning, 2011. → pages 43

[2] I. Abraham, O. Alonso, V. Kandylas, and A. Slivkins. Adaptive crowdsourcing algorithms for the bandit survey problem. In *Conference on Learning Theory*, volume 26, pages 1–29. Association for Computational Learning, 2013. → pages 8

[3] R. P. Adams and D. J. MacKay. Bayesian online changepoint detection, 2007. University of Cambridge. Technical Report. → pages 51

[4] D. Agarwal, B.-C. Chen, and P. Elango. Spatio-temporal models for estimating click-through rate. In *International Conference on World Wide Web*, pages 21–30. Association for Computing Machinery, 2009. doi:10.1145/1526709.1526713. → pages 52

[5] R. Agrawal. Sample mean based index policies with o(log n) regret for the multi-armed bandit problem. *Advances in Applied Probability*, 27(4):pp. 1054–1078, 1995. ISSN 00018678. doi:10.2307/1427934. URL http://www.jstor.org/stable/1427934. → pages 27, 32

[6] S. Agrawal and N. Goyal. Analysis of Thompson sampling for the multi-armed bandit problem. *Journal of Machine Learning Research*, pages 1–26, 2012. → pages 12, 48, 49, 50

[7] R. Allesiardo, R. Féraud, and D. Bouneffouf. A neural networks committee for the contextual bandit problem. In *Advances in Neural Information Processing Systems*, pages 374–381. Springer, 2014. doi:10.1007/978-3-319-12637-1_47. → pages 44

[8] K. J. Arrow, D. Blackwell, and M. A. Girshick. Bayes and minimax solutions of sequential decision problems. *Econometrica: Journal of the Econometric Society*, pages 213–244, 1949. → pages 12

[9] J. Aspnes. Notes on randomized algorithms. Course Notes, 2014. → pages 20

[10] J.-Y. Audibert and S. Bubeck. Minimax policies for adversarial and stochastic bandits. In *Conference on Learning Theory*, pages 773–818. Association for Computational Learning, 2009. → pages 28

[11] J.-Y. Audibert and S. Bubeck. Regret bounds and minimax policies under partial monitoring. *Journal of Machine Learning Research*, 11:2785–2836, 2010. → pages viii, 19, 21, 29, 35, 36, 41, 83

[12] J.-Y. Audibert, S. Bubeck, and R. Munos. Best arm identification in multi-armed bandits. In *Conference on Learning Theory*. Association for Computational Learning, June 2010. → pages 22

[13] P. Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3:397–422, 2003. → pages 43

[14] P. Auer and N. Cesa-Bianchi. On-line learning with malicious noise and the closure algorithm. *Annals of Mathematics and Artificial Intelligence*, 23(1-2):83–99, 1998. doi:10.1023/A:1018960107028. → pages 36

[15] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002. doi:10.1023/A:1013689704352. → pages 25, 26, 27, 28, 29, 42, 46, 49, 64, 65

[16] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *Journal on Computing*, 32(1):48–77, 2002. doi:10.1137/S0097539701398375. → pages 12, 36, 37, 43

[17] J. S. Banks and R. K. Sundaram. Switching costs and the Gittins index. *Econometrica: Journal of the Econometric Society*, pages 687–694, 1994. doi:10.2307/2951664. → pages 7

[18] M. Basseville. Detecting changes in signals and systems: a survey. *Automatica*, 24(3): 309–326, 1988. doi:10.1016/0005-1098(88)90073-8. → pages 46

[19] K. Bauman, A. Kornetova, V. Topinskii, and D. Khakimova. Optimization of click-through rate prediction in the Yandex search engine. *Automatic Documentation and Mathematical Linguistics*, 47(2):52–58, 2013. doi:10.3103/S0005105513020040. → pages 52

[20] L. Benkherouf and J. Bather. Oil exploration: sequential decisions in the face of uncertainty. *Journal of Applied Probability*, pages 529–543, 1988. → pages 7

[21] L. Benkherouf, K. Glazebrook, and R. Owen. Gittins indices and oil exploration. *Journal of the Royal Statistical Society: Series B (Methodological)*, pages 229–241, 1992. → pages 7

[22] V. Bentkus. On hoeffding's inequalities. *Annals of Probability*, 32(2):1650–1673, 2004. doi:10.1214/009117904000000360. → pages 28

[23] S. Berg. Solving dynamic bandit problems and decentralized games using the Kalman Bayesian learning automaton. Master's thesis, University of Agder, 2010. → pages 47

[24] D. Bergemann and U. Hege. Venture capital financing, moral hazard, and learning. *Journal of Banking and Finance*, 22(6):703–735, 1998. doi:10.1016/S0378-4266(98)00017-X. → pages 6

[25] D. Bergemann and U. Hege. The financing of innovation: learning and stopping. *RAND Journal of Economics*, pages 719–752, 2005. → pages 6

[26] A. Beygelzimer, J. Langford, L. Li, L. Reyzin, and R. E. Schapire. Contextual bandit algorithms with supervised learning guarantees. In *International Conference on Artificial Intelligence and Statistics*, volume 15, pages 19–26, 2011. → pages 36, 38, 43

[27] D. Bouneffouf, R. Laroche, T. Urvoy, R. Féraud, and R. Allesiardo. Contextual bandit for active learning: Active Thompson sampling. In *Advances in Neural Information Processing Systems*, pages 405–412. Springer International Publishing, 2014. doi:10.1007/978-3-319-12637-1_51. → pages 48

[28] D. B. Brown and J. E. Smith. Optimal sequential exploration: Bandits, clairvoyants, and wildcats. *Operations Research*, 61(3):644–665, 2013. doi:10.2307/23474009. → pages 7

[29] C. Browne, E. Powley, D. Whitehouse, S. Lucas, P. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton. A survey of monte carlo tree search methods. *Computational Intelligence and AI in Games, IEEE Transactions on*, 4(1):1 –43, March 2012. URL papers/ieeetcaigmctssurvey2012.pdf. A massive Survey of MCTS related papers. → pages 33

[30] S. Bubeck and N. Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1–122, 2012. doi:10.1561/2200000024. → pages 18, 19, 22

[31] S. Bubeck and C.-Y. Liu. Prior-free and prior-dependent regret bounds for Thompson sampling. In *Advances in Neural Information Processing Systems*, pages 638–646, 2013. → pages 17

[32] S. Bubeck and R. Munos. Open loop optimistic planning. In *Conference on Learning Theory*. Association for Computational Learning, June 2010. → pages 35

[33] S. Bubeck and A. Slivkins. The best of both worlds: Stochastic and adversarial bandits. In *Conference on Learning Theory*, pages 1–23. Association for Computational Learning, 2012. → pages 38

[34] S. Bubeck, R. Munos, G. Stoltz, and C. Szepesvári. Online optimization in X-armed bandits. In *Advances in Neural Information Processing Systems*, pages 201–208, 2008. → pages 34

[35] S. Bubeck, R. Munos, G. Stoltz, and C. Szepesvári. X-armed bandits. *Journal of Machine Learning Research*, 12:1655–1695, 2011. → pages 34

[36] G. Burtini, J. Loeppky, and R. Lawrence. Improving online marketing experiments with drifting multi-armed bandits. In *International Conference on Enterprise Information Systems*, 2015. → pages iii, 60

[37] G. Burtini, J. Loeppky, and R. Lawrence. A survey of online experiment design with the stochastic multi-armed bandit. *Statistics Surveys*, submitted 2015. → pages iii

[38] A. Carpentier and M. Valko. Extreme bandits. In *Advances in Neural Information Processing Systems*, volume 27, pages 1089–1097, 2014. → pages 21

[39] A. Carpentier and M. Valko. Simple regret for infinitely many armed bandits. In *International Conference on Machine Learning*. Association for Computing Machinery, 2015. → pages 16, 33

[40] N. Cesa-Bianchi and P. Fischer. Finite-time regret bounds for the multiarmed bandit problem. In *International Conference on Machine Learning*, pages 100–108. Association for Computing Machinery, 1998. doi:10.1023/A:1013689704352. → pages 26

[41] N. Cesa-Bianchi and G. Lugosi. *Prediction, learning, and games.* Cambridge University Press, 2006. → pages 37

[42] O. Chapelle and L. Li. An empirical evaluation of Thompson sampling. In *Advances in Neural Information Processing Systems*, pages 2249–2257, 2011. → pages 49, 75

[43] O. Chapelle, E. Manavoglu, and R. Rosales. Simple and scalable response prediction for display advertising. *Transactions on Intelligent Systems and Technology*, 5(4):61–95, 2014. doi:10.1145/2532128. → pages 52

[44] H. Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *Annals of Mathematical Statistics*, 23(4):493–507, 1952. doi:10.1214/aoms/1177729330. → pages 28

[45] P. D. Childs and A. J. Triantis. Dynamic R&D investment policies. *Management Science*, 45(10):1359–1377, 1999. doi:10.1287/mnsc.45.10.1359. → pages 7

[46] W. Chu, L. Li, L. Reyzin, and R. E. Schapire. Contextual bandits with linear payoff functions. In *Conference on Artificial Intelligence and Statistics*, volume 14, pages 208–214, 2011. → pages 43

[47] R. Combes and A. Proutiere. Unimodal bandits: Regret lower bounds and optimal algorithms. *International Conference on Machine Learning*, pages 521–529, 2014. → pages 33

[48] P.-A. Coquelin and R. Munos. Bandit algorithms for tree search. *Conference on Uncertainty in Artificial Intelligence*, pages 67–74, 2007. → pages 33, 34

[49] R. Coulom. Efficient selectivity and backup operators in monte-carlo tree search. In *Computers and games*, pages 72–83. Springer, 2007. → pages 34

[50] V. Dani, S. M. Kakade, and T. P. Hayes. The price of bandit information for online optimization. In *Advances in Neural Information Processing Systems*, pages 345–352, 2007. → pages 43

[51] V. Dani, T. P. Hayes, and S. M. Kakade. Stochastic linear optimization under bandit feedback. In *Conference on Learning Theory*, pages 355–366. Association for Computational Learning, 2008. → pages 65

[52] R. W. Dearden. *Learning and planning in structured worlds.* PhD thesis, The University of British Columbia, 2000. → pages 45

[53] M. Dudik, D. Hsu, S. Kale, N. Karampatziakis, J. Langford, L. Reyzin, and T. Zhang. Efficient optimal learning for contextual bandits. In *Conference on Uncertainty in Artificial Intelligence*, volume 28, pages 169–178. AUAI Press, 2011. → pages 42, 43

[54] I. Dumitriu, P. Tetali, and P. Winkler. On playing golf with two balls. *Journal on Discrete Mathematics*, 16(4):604–615, 2003. doi:10.1137/S0895480102408341. → pages 8

[55] D. Eckles and M. Kaptein. Thompson sampling with the online bootstrap. *arXiv preprint arXiv:1410.4009*, 2014. → pages 14, 48, 51, 60, 73

[56] B. Efron. Bootstrap methods: another look at the Jackknife. *Annals of Statistics*, pages 1–26, 1979. doi:10.1214/aos/1176344552. → pages 70

[57] E. Even-Dar, S. Mannor, and Y. Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7:1079–1105, 2006. → pages 25

[58] V. F. Farias and R. Madan. The irrevocable multiarmed bandit problem. *Operations Research*, 59(2):383–399, 2011. doi:10.1287/opre.1100.0891. → pages 7

[59] P. Fearnhead and P. Clifford. On-line inference for hidden Markov models via particle filters. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(4): 887–899, 2003. doi:10.1111/1467-9868.00421. → pages 51

[60] P. Fearnhead and Z. Liu. On-line inference for multiple changepoint problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(4):589–605, 2007. doi:10.1111/j.1467-9868.2007.00601.x. → pages 51

[61] S. Filippi, O. Cappe, A. Garivier, and C. Szepesvári. Parametric bandits: The generalized linear case. In *Advances in Neural Information Processing Systems*, pages 586–594, 2010. → pages 65

[62] A. Garivier and O. Cappé. The KL-UCB algorithm for bounded stochastic bandits and beyond. In *Conference on Learning Theory*, volume 24, pages 359–376. Association for Computational Learning, 2011. → pages 30

[63] A. Garivier and E. Moulines. On upper-confidence bound policies for non-stationary bandit problems. *arXiv preprint arXiv:0805.3415*, 2008. → pages 45, 46, 64

[64] S. Gelly, Y. Wang, R. Munos, and O. Teytaud. Modification of uct with patterns in monte-carlo go. 2006. → pages 34

[65] J. Gittins and D. Jones. A dynamic allocation index for the sequential design of experiments. In *Progress in Statistics*, pages 241–266. North-Holland, Amsterdam, NL, 1974. → pages 2

[66] J. Gittins, K. Glazebrook, and R. Weber. *Multi-armed bandit allocation indices*. John Wiley & Sons, 1989-2011. → pages 2

[67] J. C. Gittins. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society: Series B (Methodological)*, 41:148–164, 1979. → pages 2

[68] A. Gopalan, S. Mannor, and Y. Mansour. Thompson sampling for complex online problems. In *International Conference on Machine Learning*, pages 100–108. Association for Computing Machinery, 2014. → pages 48

[69] T. Graepel, J. Q. Candela, T. Borchert, and R. Herbrich. Web-scale Bayesian click-through rate prediction for sponsored search advertising in Microsoft's Bing search engine. In *International Conference on Machine Learning*, pages 13–20. Association for Computing Machinery, 2010. → pages 52

[70] O.-C. Granmo and S. Berg. Solving non-stationary bandit problems by random sampling from sibling Kalman filters. In *Trends in Applied Intelligent Systems*, pages 199–208. Springer Berlin Heidelberg, 2010. doi:10.1007/978-3-642-13033-5_21. → pages 19, 23, 47

[71] O.-C. Granmo and S. Glimsdal. A two-armed bandit based scheme for accelerated decentralized learning. In *Modern Approaches in Applied Intelligence*, pages 532–541. Springer Berlin Heidelberg, 2011. doi:10.1007/978-3-642-21827-9_54. → pages 47

[72] S. Guha and K. Munagala. Stochastic regret minimization via Thompson sampling. In *Conference on Learning Theory*, pages 317–338. Association for Computational Learning, 2014. → pages 48

[73] C. Hartland, S. Gelly, N. Baskiotis, O. Teytaud, and M. Sebag. Multi-armed bandit, dynamic environments and meta-bandits. In *Advances in Neural Information Processing Systems*, 2006. → pages 46, 64

[74] D. V. Hinkley. Inference about the change-point in a sequence of random variables. *Biometrika*, 57(1):1–17, 1970. doi:10.1093/biomet/57.1.1. → pages 46

[75] C.-J. Ho and J. W. Vaughan. Online task assignment in crowdsourcing markets. In *Conference on Artificial Intelligence*. AAAI, 2012. → pages 8

[76] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. doi:10.1002/0471667196.ess2066. → pages 28

[77] M. D. Hoffman, E. Brochu, and N. de Freitas. Portfolio allocation for Bayesian optimization. In *Uncertainty in Artificial Intelligence*, pages 327–336, 2011. → pages 7

[78] J. Hsu. *Multiple Comparisons: Theory and Methods*. CRC Press, 1996. → pages 26

[79] Z. Hussain, P. Auer, N. Cesa-Bianchi, L. Newnham, and J. Shawe-Taylor. Exploration vs. exploitation PASCAL challenge. http://www.pascalnetwork.org/Challenges/EEC, 2006. → pages 9, 51, 64

[80] S. Jain, S. Gujar, S. Bhat, O. Zoeter, and Y. Narahari. An incentive compatible multi-armed-bandit crowdsourcing mechanism with quality assurance. In *Symposium on Learning, Algorithms and Complexity*, 2015. → pages 8

[81] H. Jeffreys. An invariant form for the prior probability in estimation problems. *Journal of the Royal Society of London: Series A (Mathematical and Physical Sciences)*, 186(1007): 453–461, 1946. doi:10.1098/rspa.1946.0056. → pages 55

[82] C. Jennison and B. W. Turnbull. Statistical approaches to interim monitoring of medical trials: A review and commentary. *Statistical Science*, pages 299–317, 1990. doi:10.1214/ss/1177012099. → pages 6, 26

[83] D. H. Johnson and S. Sinanovic. Symmetrizing the Kullback-Leibler distance. 2001. → pages 55

[84] B. Jovanovic. Job matching and the theory of turnover. *Journal of Political Economy*, pages 972–990, 1979. doi:10.1086/260808. → pages 6

[85] D. Kahneman and G. Klein. Conditions for intuitive expertise: a failure to disagree. *American Psychologist*, 64(6):515–526, 2009. doi:10.1037/a0016755. → pages 8

[86] S. M. Kakade, S. Shalev-Shwartz, and A. Tewari. Efficient bandit algorithms for online multiclass prediction. In *International Conference on Machine Learning*, volume 25, pages 440–447. Association for Computing Machinery, 2008. doi:10.1145/1390156.1390212. → pages 44

[87] M. Kaptein. The use of thompson sampling to increase estimation precision. *Behavior research methods*, 47(2):409–423, 2015. → pages 24

[88] E. Kaufmann, O. Cappé, and A. Garivier. On Bayesian upper confidence bounds for bandit problems. In *International Conference on Artificial Intelligence and Statistics*, pages 592–600, 2012. → pages 29, 30

[89] E. Kaufmann, O. Cappé, and A. Garivier. On the complexity of best arm identification in multi-armed bandit models. *arXiv preprint arXiv:1407.4443*, 2014. → pages 17

[90] J. Kerman. A closed-form approximation for the median of the beta distribution. *arXiv preprint arXiv:1111.0433*, 2011. → pages 59

[91] S.-J. Kim, M. Aono, and M. Hara. Tug-of-war model for the two-bandit problem: Nonlocally-correlated parallel exploration via resource conservation. *BioSystems*, 101(1): 29–36, 2010. doi:10.1016/j.biosystems.2010.04.002. → pages 6

[92] R. Kleinberg, A. Slivkins, and E. Upfal. Multi-armed bandits in metric spaces. In *Symposium on Theory of Computing*, volume 40, pages 681–690. Association for Computing Machinery, 2008. doi:10.1145/1374376.1374475. → pages 34

[93] F. H. Knight. *Risk, uncertainty and profit.* New York: Hart, Schaffner and Marx, 1921. → pages 36

[94] L. Kocsis and C. Szepesvári. Bandit based monte-carlo planning. In *European Conference on Machine Learning*, pages 282–293. Springer, 2006. doi:10.1007/11871842_29. → pages 45

[95] L. Kocsis and C. Szepesvári. Discounted UCB. *EvE PASCAL Challenges Workshop*, 2006. → pages 64

[96] R. Kohavi and R. Longbotham. Online experiments: Lessons learned. *Computer*, 40(9): 103–105, 2007. doi:10.1109/MC.2007.328. → pages 84

[97] A. Krizhevsky. One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997*, 2014. → pages iii

[98] S. Kullback and R. A. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, pages 79–86, 1951. → pages 30, 55

[99] T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985. doi:10.1016/0196-8858(85)90002-8. → pages 12, 19, 22, 27, 49

[100] J. Langford and T. Zhang. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in Neural Information Processing Systems*, pages 817–824, 2007. → pages 27, 42, 43

[101] L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *International Conference on World Wide Web*, volume 19, pages 661–670. Association for Computing Machinery, 2010. doi:10.1145/1772690.1772758. → pages 9, 52, 53, 56, 66

[102] L. Li, W. Chu, J. Langford, and X. Wang. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *International Conference on Web Search and Data Mining*, pages 297–306. Association for Computing Machinery, 2011. doi:10.1145/1935826.1935878. → pages 51

[103] L. Li, W. Chu, J. Langford, T. Moon, and X. Wang. An unbiased offline evaluation of contextual bandit algorithms with generalized linear models. In *International Conference on Web Search and Data Mining*, 2012. doi:10.1145/1935826.1935878. → pages 9, 56

[104] O.-A. Maillard. Robust risk-averse stochastic multi-armed bandits. In *Algorithmic Learning Theory*, pages 218–233. Springer, 2013. doi:10.1007/978-3-642-40935-6_16. → pages 14

[105] O.-A. Maillard, R. Munos, and G. Stoltz. A finite-time analysis of multi-armed bandits problems with Kullback-Leibler divergences. In *Conference on Learning Theory*, volume 24, pages 497–514. Association for Computational Learning, 2011. → pages 22, 30

[106] M. Mayer. A glimpse under the hood at Google, 2008. → pages 84

[107] H. B. McMahan and M. J. Streeter. Tighter bounds for multi-armed bandits with expert advice. In *Conference on Learning Theory*. Association for Computational Learning, 2009. → pages 36

[108] J. Mellor and J. Shapiro. Thompson sampling in switching environments with Bayesian online change detection. In *International Conference on Artificial Intelligence and Statistics*, pages 442–450, 2013. → pages 23, 51, 64

[109] J. C. Mellor. *Decision making using Thompson Sampling*. PhD thesis, University of Manchester, 2014. → pages 52, 56

[110] R. A. Miller. Job matching and occupational choice. *Journal of Political Economy*, pages 1086–1120, 1984. → pages 6

[111] T. P. Minka. *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology, 2001. → pages 58

[112] R. E. Morin. Factors influencing rate and extent of learning in the presence of misinformative feedback. *Journal of Experimental Psychology*, 49(5):343–351, 1955. doi:10.1037/h0042813. → pages 47

[113] R. Munos. From bandits to Monte-Carlo tree search: The optimistic principle applied to optimization and planning. *Foundations and Trends in Machine Learning*, 7:1–129, 2014. doi:10.1561/2200000038. → pages 49

[114] J. Neufeld, A. György, D. Schuurmans, and C. Szepesvári. Adaptive Monte Carlo via bandit allocation. In *International Conference on Machine Learning*, volume 31, pages 1944–1952. Association for Computing Machinery, 2014. → pages 48

[115] H. T. Nguyen, J. Mary, and P. Preux. Cold-start problems in recommendation systems via contextual-bandit algorithms. *arXiv preprint arXiv:1405.7544*, 2014. → pages 48

[116] E. Page. Continuous inspection schemes. *Biometrika*, pages 100–115, 1954. doi:10.1093/biomet/41.1-2.100. → pages 46

[117] S. Pandey and C. Olston. Handling advertisements of unknown quality in search advertising. In *Advances in Neural Information Processing Systems*, volume 19, pages 1065–1072, 2006. → pages 52

[118] S. Pandey, D. Agarwal, D. Chakrabarti, and V. Josifovski. Bandits for taxonomies: A model-based approach. In *Statistical Analysis and Data Mining*, volume 7, pages 216–227. SIAM, 2007. doi:10.1137/1.9781611972771.20. → pages 42

[119] N. G. Pavlidis, D. K. Tasoulis, and D. J. Hand. Simulation studies of multi-armed bandits with covariates. In *UKSim*, pages 493–498. IEEE, 2008. → pages 66

[120] J. Pearl. Heuristics: intelligent search strategies for computer problem solving. 1984. → pages 33

[121] V. Perchet, P. Rigollet, et al. The multi-armed bandit problem with covariates. *Annals of Statistics*, 41(2):693–721, 2013. doi:10.1214/13-AOS1101. → pages 42

[122] R. S. Pindyck. A note on competitive investment under uncertainty. *American Economic Review*, pages 273–277, 1993. → pages 7

[123] P. Reverdy, V. Srivastava, and N. E. Leonard. Modeling human decision-making in multi-armed bandits. In *Proceedings of the IEEE*, volume 102, page 544, 2014. doi:10.1109/JPROC.2014.2307024. → pages 47

[124] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: estimating the click-through rate for new ads. In *International Conference on World Wide Web*, pages 521–530. Association for Computing Machinery, 2007. doi:10.1145/1242572.1242643. → pages 52

[125] H. Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952. doi:10.1007/978-1-4612-5110-1_13. → pages 12

[126] F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958. doi:10.1037/h0042519. → pages 44

[127] M. Rothschild. A two-armed bandit theory of market pricing. *Journal of Economic Theory*, 9(2):185–202, 1974. doi:10.1016/0022-0531(74)90066-0. → pages 6

[128] P. Rusmevichientong and J. N. Tsitsiklis. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411, 2010. → pages 43, 65

[129] P. Rusmevichientong and D. P. Williamson. An adaptive algorithm for selecting profitable keywords for search-based advertising services. In *Conference on Electronic Commerce*, pages 260–269. Association for Computing Machinery, 2006. doi:10.1145/1134707.1134736. → pages 52

[130] D. Russo and B. Van Roy. Learning to optimize via posterior sampling. *Mathematics of Operations Research*, 39(4):1221–1243, 2014. doi:10.1287/moor.2014.0650. → pages 48

[131] J. Sarkar. One-armed bandit problems with covariates. *Annals of Statistics*, 19(4): 1978–2002, 1991. doi:10.1214/aos/1176348382. → pages 42

[132] S. L. Scott. A modern Bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry*, 26(6):639–658, 2010. → pages 47, 52

[133] S. L. Scott. Multi-armed bandit experiments in the online service economy. *Applied Stochastic Models in Business and Industry*, 31(1):37–45, 2015. doi:10.1002/asmb.2104. → pages 52

[134] Y. Seldin, C. Szepesvári, P. Auer, and Y. Abbasi-Yadkori. Evaluation and analysis of the performance of the EXP3 algorithm in stochastic environments. In *European Workshop on Reinforcement Learning*, pages 103–116. Microtome Publishing, 2013. → pages 14

[135] B. Shahriari, Z. Wang, M. W. Hoffman, A. Bouchard-Côté, and N. de Freitas. An entropy search portfolio for Bayesian optimization. *arXiv preprint arXiv:1406.4625*, 2014. → pages 48

[136] M. Sorensen. Learning by investing: Evidence from venture capital. In *New Orleans Meetings Paper*. AFA, 2008. doi:10.2139/ssrn.967822. → pages 7

[137] V. Srivastava, P. Reverdy, and N. Leonard. Optimal foraging and multi-armed bandits. In *Allerton Conference on Communication, Control, and Computing*, pages 494–499, 2013. doi:10.1109/Allerton.2013.6736565. → pages 47

[138] G. Steinbrecher and W. T. Shaw. Quantile mechanics. *European Journal of Applied Mathematics*, 19(02):87–112, 2008. → pages 73

[139] A. L. Strehl, C. Mesterharm, M. L. Littman, and H. Hirsh. Experience-efficient learning in associative bandit problems. In *International Conference on Machine Learning*, pages 889–896. Association for Computing Machinery, 2006. doi:10.1145/1143844.1143956. → pages 42

[140] R. S. Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *International Conference on Machine Learning*, volume 7, pages 216–224. Association for Computing Machinery, 1990. → pages 45

[141] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction.* MIT Press, 1998. → pages 25

[142] I. Szita and A. Lőrincz. The many faces of optimism: a unifying approach. In *International Conference on Machine Learning*, pages 1048–1055. Association for Computing Machinery, 2008. doi:10.1145/1390156.1390288. → pages 49

[143] L. Tang, R. Rosales, A. Singh, and D. Agarwal. Automatic ad format selection via contextual bandits. In *International Conference on Information and Knowledge Management*, pages 1587–1594. Association for Computing Machinery, 2013. doi:10.1145/2505515.2514700. → pages 52

[144] W. R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 1933. → pages 9, 11, 47, 49, 62

[145] A. Tikhonov. Solution of incorrectly formulated problems and the regularization method. In *Soviet Mathematics - Doklady*, volume 5, pages 1035–1038. American Mathematical Society, 1963. → pages 66

[146] L. Tran-Thanh, S. Stein, A. Rogers, and N. R. Jennings. Efficient crowdsourcing of unknown experts using multi-armed bandits. In *European Conference on Artificial Intelligence*, volume 20, pages 768–773, 2012. doi:10.1016/j.artint.2014.04.005. → pages 8

[147] H. Tyagi and B. Gärtner. Continuum armed bandit problem of few variables in high dimensions. In *Workshop on Approximation and Online Algorithms*, volume 8447, pages 108–119. Springer International Publishing, 2013. doi:10.1007/978-3-319-08001-7_10. → pages 33

[148] H. Tyagi, S. U. Stich, and B. Gärtner. On two continuum armed bandit problems in high dimensions. *Theory of Computing Systems*, pages 1–32, 2014. doi:10.1007/s00224-014-9570-8. → pages 33

[149] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984. doi:10.1145/1968.1972. → pages 20

[150] M. Valko, R. Munos, B. Kveton, and T. Kocák. Spectral bandits for smooth graph functions. In *International Conference on Machine Learning*, volume 31, pages 46–54. Association for Computing Machinery, 2014. → pages 35

[151] K. Van Moffaert, K. Van Vaerenbergh, P. Vrancx, and A. Nowé. Multi-objective χ-armed bandits. In *International Joint Conference on Neural Networks*, pages 2331–2338. IEEE, 2014. doi:10.1109/IJCNN.2014.6889753. → pages 33

[152] J. Vermorel and M. Mohri. Multi-armed bandit algorithms and empirical evaluation. In *European Conference on Machine Learning*, pages 437–448. Springer, 2005. doi:10.1007/11564096_42. → pages 12, 25, 26, 31, 32, 67

[153] S. S. Villar, J. Bowden, and J. Wason. Multi-armed bandit models for the optimal design of clinical trials: Benefits and challenges. *Statistical Science*, 30(2):199–215, 2015. doi:10.1214/14-STS504. → pages 6

[154] A. Wald. *Sequential Analysis.* Wiley Series in Probability and Mathematical Statistics. J. Wiley & Sons, Incorporated, 1947. → pages 12

[155] C.-C. Wang, S. R. Kulkarni, and H. V. Poor. Bandit problems with side observations. *Transactions on Automatic Control*, 50(3):338–355, 2005. doi:10.1109/TAC.2005.844079. → pages 42

[156] C. J. C. H. Watkins. *Learning from delayed rewards.* PhD thesis, University of Cambridge, 1989. → pages 12, 25

[157] A. Weinstein. Big-ol-bandit (BOB) implemented, tested. http://aresearch.wordpress.com/2010/07/13/big-ol-bandit-bob-implemented-tested/, 2010. Accessed: June 2, 2015. → pages 35

[158] M. L. Weitzman. Optimal search for the best alternative. *Econometrica: Journal of the Econometric Society*, pages 641–654, 1979. → pages 7

[159] P. H. Westfall, S. S. Young, and S. P. Wright. On adjusting p-values for multiplicity. *Biometrics*, pages 941–945, 1993. doi:10.2307/2532216. → pages 26

[160] M. Woodroofe. A one-armed bandit problem with a concomitant variable. *Journal of the American Statistical Association*, 74(368):799–806, 1979. doi:10.1080/01621459.1979.10481033. → pages 42