2016-06

New perspectives on intelligence collection and processing

Tekin, Muhammet

Monterey, California: Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

NEW PERSPECTIVES ON INTELLIGENCE COLLECTION AND PROCESSING

by

Muhammet Tekin

June 2016

Thesis Advisor: Roberto Szechtman
Second Reader: Michael Atkinson

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704–0188

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	06-17-2016	Master's Thesis 09-29-2014 to 06-17-2016

4. TITLE AND SUBTITLE

NEW PERSPECTIVES ON INTELLIGENCE COLLECTION AND PROCESSING

5. FUNDING NUMBERS

6. AUTHOR(S)

Muhammet Tekin

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed in this document are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol Number: N/A.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

Intelligence-production activities are typically viewed as part of an intelligence cycle, consisting of planning, collection, processing, analysis, and dissemination stages. Once a request for information is issued, the intelligence agencies mostly deal with the collection and processing activities of the cycle. However, in most situations, there is an enormous amount of data to be collected. This overabundance of information requires methods that select only the useful data, to prevent intelligence personnel from wasting time and effort on non-relevant data. Online learning is an area of research that has gained attention in recent years with applications in areas such as web advertising, classification, and decision making. In this thesis, we develop a model aimed at the collection and processing phases of the intelligence cycle, applicable in situations where the data is obtained sequentially, so that learning algorithms are realistic. We analyze the performance of a modified Thompson Sampling algorithm, to help intelligence analysts make good decisions, regarding the sources from which to collect/process as well as the collection/processing capacity and its allocation over time, in order to bind the risk of missing valuable information below a certain threshold.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Online Learning, Thompson Sampling, Intelligence Collection	75
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

NEW PERSPECTIVES ON INTELLIGENCE COLLECTION AND PROCESSING

Muhammet Tekin
Captain, Turkish Army
B.S., Turkish Military Academy, 2007

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
June 2016

Author: Muhammet Tekin

Approved by: Roberto Szechtman
Thesis Advisor

Michael Atkinson
Second Reader

Patricia A. Jacobs
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Intelligence-production activities are typically viewed as part of an intelligence cycle, consisting of planning, collection, processing, analysis, and dissemination stages. Once a request for information is issued, the intelligence agencies mostly deal with the collection and processing activities of the cycle. However, in most situations, there is an enormous amount of data to be collected. This overabundance of information requires methods that select only the useful data, to prevent intelligence personnel from wasting time and effort on non-relevant data. Online learning is an area of research that has gained attention in recent years with applications in areas such as web advertising, classification, and decision making. In this thesis, we develop a model aimed at the collection and processing phases of the intelligence cycle, applicable in situations where the data is obtained sequentially, so that learning algorithms are realistic. We analyze the performance of a modified Thompson Sampling algorithm, to help intelligence analysts make good decisions, regarding the sources from which to collect/process as well as the collection/processing capacity and its allocation over time, in order to bind the risk of missing valuable information below a certain threshold.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

List of Figures

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

CIA	Central Intelligence Agency
COMINT	Communications Intelligence
CYBINT/DNINT	Cyber Intelligence/Digital Network Intelligence
DoD	Department of Defense
EM	Expectation Maximization
Exp3	Exponential-weight Algorithm for Exploration and Exploitation
FBI	Federal Bureau of Investigation
FININT	Financial Intelligence
GEOINT	Geospatial Intelligence
HUMINT	Human Intelligence
IC	Intelligence Community
MAB	Multi-Armed Bandit
MASINT	Measurement and Signature Intelligence
NPS	Naval Postgraduate School
OSINT	Open Source Intelligence
pdf	Probability Density Function
SIGINT	Signals Intelligence
TECHINT	Technical Intelligence
TS	Thompson Sampling
UCB	Upper Confidence Bound

U.S.	United States

Executive Summary

Intelligence activities are part of a framework called intelligence cycle. The main phases in the cycle are planning, collection, processing, analyzing, and dissemination of information. Some of the data collected from the operational environment is discarded due to irrelevance. This discarded data wastes time and effort for the intelligence organizations.

As demonstrated in [1], "The goal of online learning is to make a sequence of accurate predictions given knowledge of the correct answer to previous prediction tasks and possibly additional available information." Online learning is used when learning with training data is infeasible, or the data is non-stationary. It is also used to adapt to the changes in the environment. Learning as one goes along can be more robust than specifying a model and using mathematical optimization [2]. Although it was first defined in machine-learning literature, its application can be found in other areas such as optimization, game theory, and statistical modeling.

The main goal of the thesis is to address the challenge of how to efficiently explore the data originating from a large number of sources. For this purpose, we propose a model that is suitable for most of the situations in the collection and processing phases. In particular, we modify the well-known Thompson Sampling (TS) algorithm, originating in the machine-learning community, to sample from more than one source at a time, as is the case in intelligence organizations that analyze large numbers of items concurrently.

We address the following questions:

- How can we create a balance between exploration and exploitation to maximize the benefit when making decisions as to the collection, processing, or analysis of data if there are time or resource constraints?
- How can we decide what amount of resources should be allocated for collection, processing, or analyzing if we can not change the allocation in the short term?
- How can we adjust the allocation dynamically while learning occurs online?
- How can we quantify the risk of missing relevant data versus the amount of resources allocated?

- How can we use the data gathered to gain insights about the population of the sources that generated the data?

We modified the TS algorithm, so we can explore arbitrarily large number of sources, limited only by the resources of the intelligence organization. The measure of performance is the regret: the difference between the rewards obtained by the algorithm and the rewards that could have been obtained had the analyst known the true nature of each source of intelligence. The expected regret has recently been shown to grow sublinearly in number of time periods (T), so that learning does indeed occur. TS, in its base form, leads to learning on the order of $O(\log T)$, meaning that the expected average regret goes to zero as the time horizon T increases.

Our main conclusions and contributions can be summarized as follows:

- The model described can be used to allocate the collection/processing resources/efforts efficiently.
- The suggested algorithm employed in the model yields a sublinear performance in the simulations we conducted, meaning that the average regret tends to zero as the number of time periods gets larger.
- The model can be adapted to situations in which prior knowledge about the sources exists.
- We consider the capacity allocated as possibly changing over time, as information becomes available. With this approach, intelligence agencies can better control the regret in the exploration phase and avoid using excess capacity.
- The model can also be employed to gain insights about the risk of missing relevant data, which provide further guidance for the capacity required.
- We also use the Expectation Maximization (EM) algorithm to estimate the distributional parameters for the statistical model of the candidate subpopulations as data is collected.

List of References

[1] S. Shalev-Shwartz, "Online learning and online convex optimization," *Foundations and Trends in Machine Learning*, vol. 4, no. 2, pp. 107–194, 2011.

[2] H. Elad, "Draft: Introduction to online convex optimization," 2015, unpublished.

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgments

The time I have spent at the Naval Postgraduate School has been invaluable for me. Besides the academic education I received, I was given the chance for full-time professional development. Therefore, my first thanks goes to my fellow citizens, whose taxes I have relied on here.

My advisor, Dr. Roberto Szechtman, who pushed me to delve into online optimization, has been very kind and helpful with all his efforts including but not limited to guidance and discussions about the readings. I am grateful for his support during the endeavor to finish my thesis.

I thank my second reader, Dr. Michael Atkinson, for his guidance and suggestions.

Another thanks goes to the NPS staff, especially Library and International Office personnel, for their support in making my life easy over the entire education period.

I also want to thank my family, my father, mother, and brothers, as well as my wife, Emel, and son, Umit, for their love and presence in my life.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

Knowledge is power only if man knows what facts not to bother with.

Robert Staughton Lynd [1]

1.1 Introduction

The amount of data available to intelligence agencies has skyrocketed in recent years in parallel to technological advances. Fully handling this huge amount of data is beyond the capabilities (human and technological) of any organization if done in a naive manner.

The challenge is well summarized by Hedley:

> In the twenty-first century, a principal analytic challenge lies in the sheer volume of information available. Although especially hard targets such as terrorist cells are no less difficult to penetrate, the explosion of open-source information from news services and the Worldwide Web makes the speed and volume of reporting more difficult to sift through. Advances in information technology both help and hinder, as analysts strive to cope with the "noise," the chaff they must winnow away. Data multiply with dizzying speed. Whereas collecting solid intelligence information was the overriding problem of the past, selecting and validating it loom ever larger as problems for analysts today. [2]

Intelligence activities are commonly considered within a framework called the Intelligence Cycle. The main phases in the cycle are planning, collection, processing, analyzing, and dissemination of the information. The detailed discussion of these stages is provided in Chapter 2.

Some of the data collected from the operational environment is discarded because it is irrelevant by the point when final judgments are made by analysts. According to Wirtz [3],

"Observers ritualistically point out that analysts are constantly at risk of being overwhelmed by a deluge of information from both open and classified sources. Yet, the real danger may be the fact that, within this data stream, there is little valuable information about the highest priority targets and issues facing analysts."

Besides the relevancy of the data, the quality of the source that generates the data has to be understood in order to focus on the sources that are more likely to produce relevant data.

Online learning is characterized by updating the beliefs about the ground truth, which is unknown, as new data becomes available. Because the uncertainty is reduced as a result of explorations, one ought to adapt his decisions in light of new information. This approach is also applicable for most of the situations wherein decisions are made in sequence. In some collection contexts, data is collected in a sequential manner that allows analysts to apply online learning methods to assess the quality of the sources. For example, the sources could be a number of tracked Twitter accounts. The tweets that are collected result in the costs of time and effort. The data (i.e., messages) from these sources (i.e., Twitter accounts) become available sequentially. It is straightforward to come up with other similar scenarios for which online learning methods are germane. In summary, gaining information about the quality of the sources allows the analyst to select a subset of them, given the capacity and time constraints.

1.2 Research Questions and Methodology

In this study, we adopt a model that allows us to benefit from the ideas developed within the online learning community. We adapt the well-known Thompson Sampling (TS) algorithm to a case in which many samples can be explored at a time. We address the following questions (see Chapters 3 and 4 for more details):

- How can we create a balance between exploration and exploitation to maximize the benefit when making decisions as to the collection, processing, or analysis of data if there are time or resource constraints?
- How can we decide what amount of resources should be allocated for collection, processing, or analyzing if we can not change the allocation in the short term?
- How can we adjust the allocation dynamically while learning occurs online?

- How can we quantify the risk of missing relevant data vs. the amount of resources allocated?
- How can we use the data gathered to gain insights about the population of the sources that generated the data?

Initially, we develop a modeling framework that is suitable to answer the aforementioned questions and then analyze its performance with numerical examples.

1.3 Scope

We intentionally define the model to be analyzed in a generic fashion. The sources in the model could be discrete portions of a wide geographical area. Items from these sources are collected with a UAV, processed using specific algorithms at headquarters, and further analyzed by a human analyst. To keep the model as simple as possible, in this work, we do not consider specific issues for each setting (e.g., UAV travel time between non-adjacent geographic locations).

In Table 1.1, we include several other illustrative examples.

Table 1.1: Examples for Sources and Capacities (Resources) for Different Phases

Phase	**Source (Where to sample)**	**Resource**
Collection	A geographical area, A frequency band, An edge of a social network	A satellite, Signal interceptor
Processing	Data aggregated from the collection phase	Decryption tool, Automatic translator
Analysis	Processed data, Translated, decrypted message, Restructured data	Human analysts

1.4 Structure of the Thesis

This thesis consists of five chapters. Chapter 2 is dedicated to a background on intelligence activities as well as the literature review. Chapter 3 introduces the modeling framework, including the assumptions and the proposed approach. In Chapter 4, we analyze the model from Chapter 3 to answer the research questions. Also, Chapter 4 contains numerical results obtained from the simulations. In Chapter 5, we offer conclusions.

CHAPTER 2:
Background and Literature Review

In Chapter 2, we provide information about intelligence and online learning. Then, we discuss to what extent the activities performed in the intelligence cycle are related to, and can be modeled by, online learning. Finally, we look at the related studies on the overlap.

2.1 Intelligence

Intelligence is a means to an end [4]. From a state perspective, the most important goal of intelligence is to provide security to the people. Almost all states have dedicated intelligence agencies. These agencies have similar structures and procedures. Their typical missions are to collect relevant information and to conduct objective analyses. One of the key challenges is to leverage technological advances for better performance in agency missions [5].

2.1.1 Definition and Categories of Intelligence

Intelligence is an elusive term, so we need to clarify its meaning. A broad definition may be the information that has been collected, processed, and analyzed for the use of decision/policy makers. Besides the final product, the term *intelligence* is also used to refer to *the process* through which it is produced, *the organization* that produces it, or the whole *Intelligence Community* [6]. Formal definitions of intelligence from the Department of Defense *Dictionary of Military and Associated Terms* [7] are as follows:

> The **product** resulting from the collection, processing, integration, evaluation, analysis, and interpretation of available information concerning foreign nations, hostile or potentially hostile forces or elements, or areas of actual or potential operations.
>
> The **activities** that result in the product.
>
> The **organizations** engaged in such activities.

2.1.2 Intelligence Cycle

Although intelligence officers admit that effective intelligence efforts are not cyclic [8], production and consumption of the intelligence is traditionally considered a cyclic process, as shown in Figure 2.1. The main steps in the cycle include identifying requirements/needs, planning and direction, collection, processing, analysis and production, and dissemination [8].

The steps or categories of the cycle represent the related activities conducted by the agencies. Activities in each category may happen concurrently, or some steps may be bypassed. For example, a requirement may be addressed by analyzing existing data without any collection effort.

Briefly, the cycle can be explained as follows:

- Consumers determine the *requirements* and the priorities.
- Agencies *plan* all the efforts necessary through the process until the delivery of the final product to the consumer.
- Data is *collected* via intelligence gathering disciplines.
- Huge amounts of data are *processed* and converted into a form usable by the analysts.
- *Analysts* determine the relevancy and the importance of the data.
- The intelligence is *disseminated* to the consumer who demanded it.

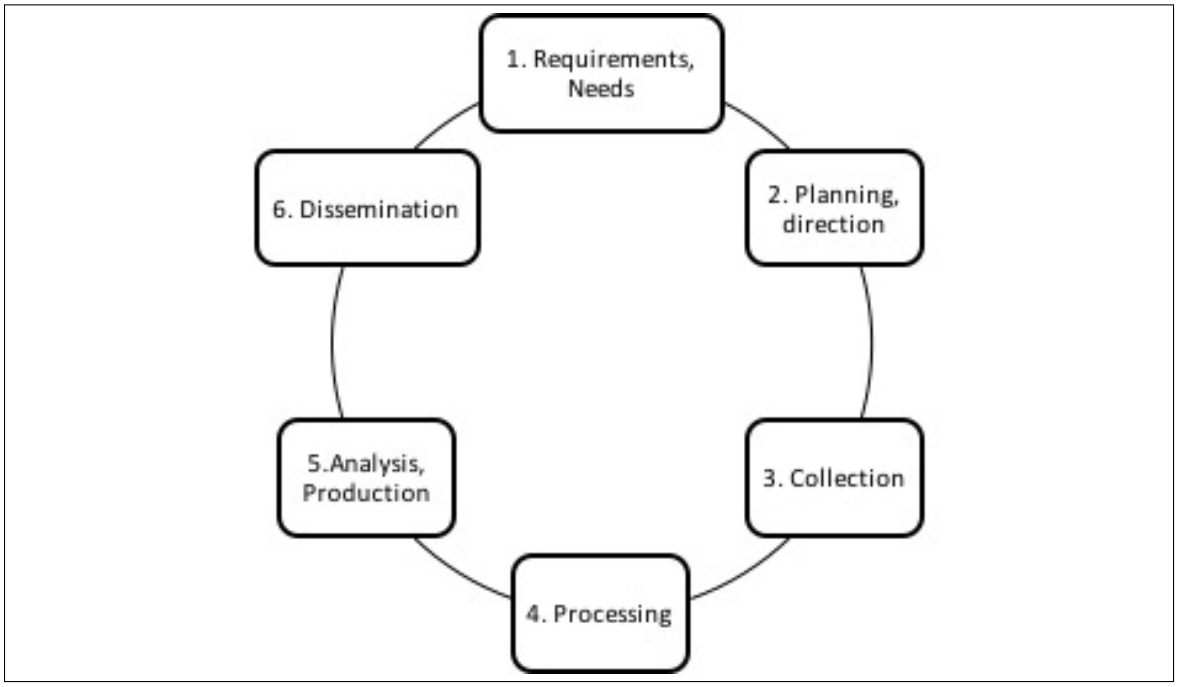

Figure 2.1: Intelligence Cycle.

Raw data is filtered out depending on its relevancy (importance) as it goes through the phases of the intelligence cycle, and it takes on different names along the way: data, information, and intelligence. The relationship between data, information, and intelligence [9] is shown in Figure 2.2. The operational environment harbors all the information we collect, but we generally collect only a fraction of it. Because we do not know the exact importance of data before it goes through the entire cycle, the decision regarding where to collect the data is itself a filtration. After the data is collected, it is (pre)processed and, therefore, subject to another filtration. Finally, analysts examine the information and possibly discard some portion of the information as irrelevant. Throughout the collection phase, it is important to collect the least amount of data needed so as to increase efficiency. This efficiency requirement drives our motivation to apply online learning ideas when appropriate.

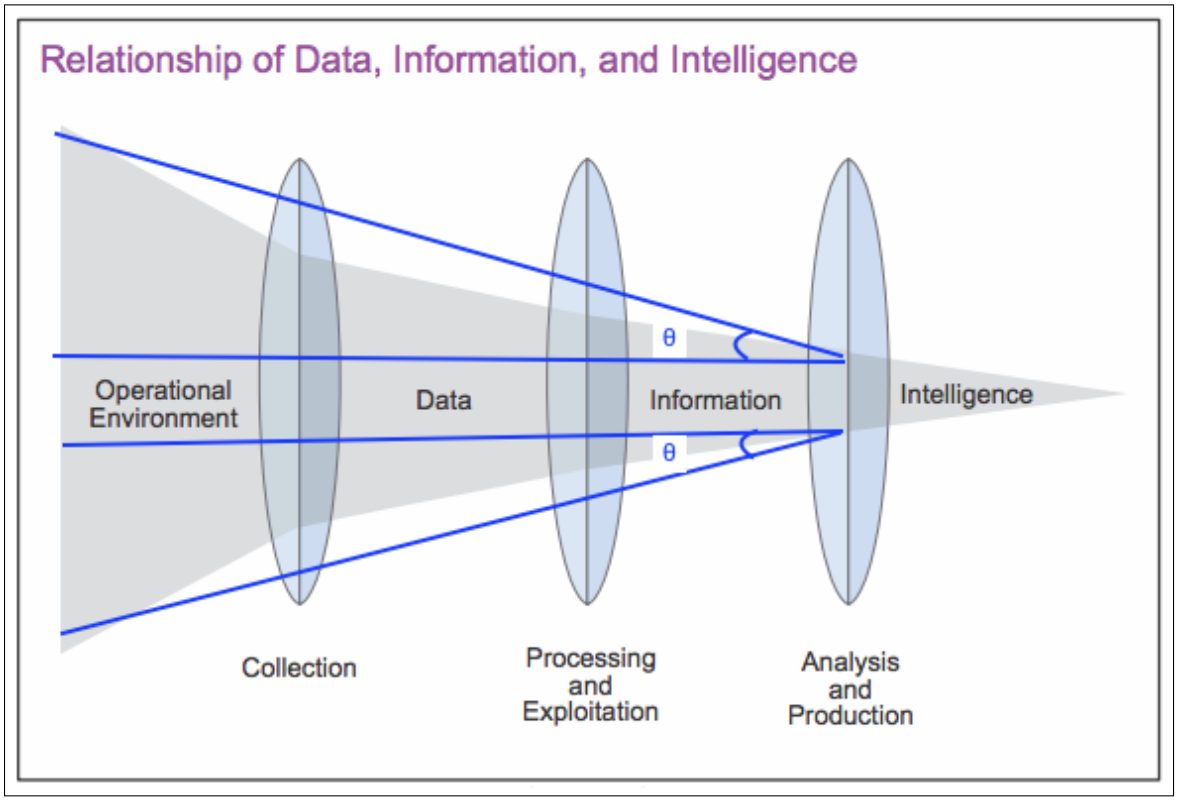

Figure 2.2: Relationship between Data, Information and Intelligence. Adapted from [9].

2.1.3 Intelligence Collection

In the collection phase, the data is gathered from diverse domains with many techniques or assets, varying from human to very sophisticated instruments. These differ according to the following intelligence-gathering disciplines:

- Human Intelligence (HUMINT): humans on the ground
- Geospatial Intelligence (GEOINT): satellite, aerial photography, mapping/terrain data
- Measurement and Signature Intelligence (MASINT): different types of sensors
- Open Source Intelligence (OSINT): from all open sources
- Signals Intelligence (SIGINT): intercepting the signals
- Technical Intelligence (TECHINT): analysis of technical information of the weapons and equipment used the by foreign nations

- Cyber Intelligence/Digital Network Intelligence (CYBINT/DNINT): cyber space
- Financial Intelligence (FININT): analysis of monetary transactions

Parallel to the advancement of technology, the capabilities of the collection assets, especially technical ones, constantly improve. However, the required resolution of the information collected still prevents the agencies from collecting from all possible sources, e.g., geographical areas, spectrum, Internet traffic. As for an intelligence satellite, if its movement is not synchronized with the earth's movement (geostationary), it can only look at a portion of the area for a limited time period. On the other hand, collection or processing data from a source may be costly due to encryption, deception, or other denial techniques [4]. For example, people may exchange encrypted messages within a social communication network. Here, allocating collection efforts to the candidate sources has to be done wisely to maximize the value obtained. These typical situations pose an exploration/exploitation dilemma similar to that of a Multi-Armed Bandit (MAB) problem, as discussed in Section 2.2.

2.1.4 Intelligence Processing

In the collection phase, a large amount of data (especially from SIGINT) is obtained that requires processing before being delivered to an analyst. Processing may include organizing, structuring, or translating the data. Given the constraints on the processing efforts and the large volume of data complete analysis may be infeasible. Even time-critical data may be left untouched until a processing effort is allocated.

2.1.5 Intelligence Analysis

The flood of information may keep analysts busy just reading the incoming information without producing any intelligence [10]. Processed intelligence is first put before the analysts. Daily, they look at the information coming from different sources. Naturally, analytical capacities often lag behind the collection and processing capacities [4]. Given the time constraint and difficulty of analyzing all the information, the exploration and exploitation of the relevant sources is also viable for the analyzing phase.

2.2 Online Learning

Section 2.2 defines online learning and examines how it can be applied to intelligence.

2.2.1 Overview

Although it was first defined in machine-learning literature, online learning can be applied to other areas such as optimization, game theory, and statistical modeling.

According to [11], "The goal of online learning is to make a sequence of accurate predictions given knowledge of the correct answer to previous prediction tasks and possibly additional available information." It is used when learning with a training data is infeasible or the data is non-stationary. Online learning is also used to adapt to the changes in the environment. Learning as one goes along can be more robust than specifying a model and using mathematical optimization [12].

2.2.2 Multi-Armed Bandit Problem

The MAB problem is one particular setting for online learning. In a MAB problem, an agent chooses one of K machines (bandits) to play in each game (iteration). In order to maximize his gain, the agent has to allocate his money wisely between exploring the *good* bandits and exploiting the information learned.

The problem is important when the decision maker has a budget constraint that prevents him from learning the truth about each alternative before making a decision. Important applications of the bandit model:

- Clinical trials investigating the effects of different experimental treatments while minimizing patient losses [13]
- Adaptive routing efforts for minimizing delays in a network [14]
- Financial portfolio design [15]
- Resource allocation to various projects given uncertainty about the difficulty and profit of each possibility [16]

2.2.3 Exponential-weight Algorithm for Exploration and Exploitation

Exponential-weight Algorithm for Exploration and Exploitation (Exp3) [17] can be used to approximately solve the MAB problem as previously described. The performance of Exp3 is measured by *weak regret*, the difference between the total rewards accumulated by the best machine and the sum of the obtained rewards throughout the game history [17].

Figure 2.3 shows the pseudo-code of the algorithm [17]. An obvious downside of the algorithm is that the parameter γ, which represents the uniformly allocated part of the probability to the machines, needs to be provided in advance. As proved by Auer et al. [17], optimal γ is calculated as follows: $\gamma = min\left(1, \sqrt{\frac{KlnK}{(e-1)g}}\right)$. Here, parameter g is the upper bound of the total weak regret after a prospected time horizon T [17]. When rewards are at the interval [0,1], maximum regret cannot be greater than T. Therefore, we can replace g with T in the equation.

Parameters: real γ
Initialization: $w_i(1) = 1$ for $i = 1, \ldots, K$
For each $t = 1, 2, \ldots$

- set $p_i(t) = (1-\gamma)\frac{w_i(t)}{\sum_{j=1}^{K} w_j(t)} + \frac{\gamma}{K}$ for $i = 1, \ldots, K$
- draw i_t randomly accordingly to the probabilities $p_1(t), \ldots, p_K(t)$
- receive reward $x_{i_t}(t) \in [0, 1]$
- for $j = 1, \ldots, K$ set
  $\hat{x}_j(t) = \frac{x_j(t)}{p_j(t)}$ if $j = i_t$ otherwise 0
  $w_j(t+1) = w_j(t) exp\left[\frac{\gamma \hat{x}_j(t)}{K}\right]$

Figure 2.3: Pseudocode of Exp3 Algorithm. Adapted from [17].

In other words, decisions are selected randomly from a probability mass function of bandits, which is updated at each iteration according to the reward obtained. One salient characteristic of the algorithm is that a portion of the probability is allocated uniformly to all of the bandits so as to keep exploring the bandits.

2.2.4 Thompson Sampling

TS, as demonstrated in [18], is a heuristic method that can be utilized to address the exploration and exploitation tradeoff posed by MAB. The idea behind TS is to choose the action (choice of a machine to play) that has the largest expected reward according to the posterior reward distributions of the actions. Since the expectations can not be analytically computed and it is easier to sample from a posterior distribution, actions are drawn randomly from the corresponding posterior distributions in each time period; then, the belief distributions are updated using past observations (Bayesian update).

TS is described in [19] as follows:

Consider a set of actions $\mathscr{A}$, and rewards in $\mathbb{R}$. In each round, the player chooses an action $a \in \mathscr{A}$ and obtains a reward $r \in \mathbb{R}$ following a distribution that depends on the issued action. The aim of the player is to play actions such as to maximize the cumulative rewards.

The following are elements of Thompson sampling:

- A likelihood function $P(r|\theta, a)$;
- A set Θ of parameters θ of the distribution of r;
- A prior distribution $P(\theta)$ on these parameters;
- Past observations $\mathscr{D} = \{(a; r)\}$;
- A posterior distribution $P(\theta|\mathscr{D}) \propto P(\mathscr{D}|\theta)P(\theta)$.

TS consists in playing the action $a^* \in \mathscr{A}$ according to the probability that it maximizes the expected reward, i.e., $\int \mathbb{I}[\mathbb{E}(r|a, \theta) = \max_{a'} \mathbb{E}(r|a', \theta)]P(\theta|\mathscr{D})\, d\theta$. In practice, the rule is implemented by sampling, in each time period, a parameter θ^* from the posterior $P(\theta|\mathscr{D})$ and choosing the action a^* that maximizes $\mathbb{E}[r|\theta^*, a^*]$, i.e., the expected reward given the parameter and the action [20]. In words, the player selects his beliefs randomly from posteriors, and then acts optimally according to them [20].

TS was not common in literature until recently. Chapelle and Lee [19] present "empirical results using TS on simulated and real data, and show that it is highly competitive," compared to other known algorithms, e.g., Upper Confidence Bound (UCB), and robust to observation delays. Furthermore, TS is easy to implement and takes no parameter that has to be determined in advance, unlike Exp3.

2.3 Intelligence and Online Learning

As discussed in Section 1 and demonstrated in Figure 2.2, the data moves through until the ultimate consumption by the consumer and is subject to filtration. Not all of the collected data will eventually become full-fledged intelligence. It may be discarded as irrelevant or insignificant after being collected, processed, or analyzed. It may not even be collected in the first place because of its presence in an irrelevant domain.

The amount or number of items that are discarded as irrelevant in any phase of the intelligence cycle can be an indicator of the efficiency of the process. In other words, we need to collect, process, and analyze the least amount of data necessary to produce a certain intelligence without wasting any effort. This can be achieved by collecting the data from the domain in an adaptive manner. The potential of the domain's parts can be learned as the data are collected and fed into the cycle, which in turn can redirect the collection asset to more promising parts of the domain. The same logic is applicable for the processing and analyzing phases.

Online learning principles offer promising guidance regarding how this filtration can be done. The inherent exploration and exploitation dilemmas that exist in all collection, processing, and analyzing efforts are conducive to approaching the filtration as an online learning setting such as MAB.

Different settings exposed by collection, processing, and analyzing efforts, or even by different techniques employed in each effort, may lead to different assumptions regarding the dependence of the sources (both temporal and spatial). Optimal learning approach can be modeled using modifications of existing learning algorithms/heuristics, such as Exp3 and TS in pursuit of efficient solutions resulting in the maximum gain with the available collecting, processing, or analyzing capacity.

2.4 Related Studies

To the best of our knowledge, there are three Naval Postgraduate School (NPS) master's theses that touch upon efficient intelligence collection and processing.

Costica [21] considers the bottleneck or congestion caused by a huge information flow and proposes a tandem queue model for a preliminary classification of intelligence items regarding their relevance to an intelligence request.

Nevo [22] specifies a network model for social network communication and treats the edges as the sources. To maximize the relevant data discovered, he compares the performances of several learning algorithms under the time constraints.

Ellis [23] analyzes the performance of some learning algorithms in detecting relevant conversations from an intercepted social communication network, as represented by the model developed by Nevo.

CHAPTER 3:
Model

In this chapter, we set the framework for our model and parameter estimation approach that addresses the efficient data collection.

3.1 Setting

In order to keep the developments as generic as possible, we define several terms. As described in Chapter 2, intelligence cycle consists of the collection, processing, and analysis stages. For each of these stages, items originate from different *sources*. Items that are examined require a certain amount of resources (capacity), depending on the cycle stage.

For instance, the source could be a certain geographical area. Items from an area are collected with a UAV, processed using specific algorithms at headquarters, and further analyzed by a human analyst. To keep the model as simple as possible, in this work, we do not consider specific issues for each setting (e.g., UAV travel time between non-adjacent geographic locations). In the Table 3.1, we include several other illustrative examples.

Table 3.1: Examples for Sources and Capacities (Resources) for Different Phases

Phase	**Source (Where to sample)**	**Resource**
Collection	A geographical area, A frequency band, An edge of a social network	A satellite, Signal interceptor
Processing	Data aggregated from the collection phase	Decryption tool, Automatic translator
Analysis	Processed data, Translated, decrypted message, Restructured data	Human analysts

3.2 Model

There are S sources assumed to be independent. In each time period t, source $s \in \{1, \ldots, S\}$ generates a relevant item with probability p_s, assumed constant in time, independently of past observations. Only $q < S$ (but often times, $q \ll S$) sources can be explored each time period. For simplicity, we assume that there are no misjudgments, meaning that there are no classification mistakes by the resource employed to examine the item (see Table 3.1). Exploring a relevant item yields a reward of 1, while non-relevant items yield no reward, or a reward of 0. We also assume that the cost of exploring an item from a source is fixed and equal for all sources and is, thus, not explicitly considered in the model. The measure of performance is the total expected reward over some finite horizon T. Hence, if the values of p_s for all sources s were known, then simply exploring the q sources with the largest probability p_s of yielding a relevant item would maximize the expected reward.

However, in most realistic situations, the values of p_s are unknown because there is little available information from the source. In this work, we assume that existing and past information can be subsumed into a prior distribution for p_s for each source s. On the one extreme, if the analyst has a very high degree of certainty on the value of p_s for some source s, then he would put a prior density centered at some value at $(0,1)$, with most of the mass concentrated around that value. If, on the other hand, the analyst knows nothing about p_s, then he would assume a uniform prior. In other words, we model the probabilities of yielding a relevant item for each source, $p_1, \ldots, p_S$, as themselves being randomly drawn from some distribution that may depend on the source s.

The beta distribution, with Probability Density Function (pdf)

$$f(p; \alpha_s, \beta_s) \propto p^{\alpha_s}(1 - p^{\beta_s}),$$

is proportional to the likelihood of p given α_i successes and β_i failures. The beta distribution is appealing because it is conjugate with the Bernoulli distribution, which is associated with $\{0,1\}$ reward situations such as ours. The beta distribution is continuous over 0 to 1, with mean $\alpha_s/(\alpha_s+\beta_s)$ and variance $\alpha_s\beta_s/[(\alpha_s+\beta_s)^2(\alpha_s+\beta_s+1)]$. Under the assumption that $p_s \sim \text{Beta}(\alpha_s, \beta_s)$, the analyst may use historical data to estimate the parameters α_s and β_s. If there is no historical data, choosing $\alpha_s = \beta_s = 1$ is akin to assuming a uniform prior.

A source that with high certainty yields non-relevant items would have $1 \ll \alpha_s \ll \beta_s$, while one that with low certainty generates relevant items would have $1 \gg \alpha_s \gg \beta_s > 0$.

Source s gets explored from

$$X_s \sim \text{Bernoulli}(p_s),$$

and the posterior distribution for p_s becomes $\text{Beta}(\alpha_s + x_s, \beta_s + 1 - x_s)$, on $X_s = x_s \in \{0,1\}$. Hence, after $n_{s,t}$ explorations of items generated by source s by period t, we have

$$p_s | x_{s,1}, \ldots, x_{s,n_{s,t}} \sim \text{Beta}(\alpha_s + y_{s,t}, \beta_s + n_{s,t} - y_{s,t}),$$

where $y_{s,t} = \sum_j^{n_{s,t}} x_{s,j}$ is the number of relevant items generated by source s by time t.

As source s is explored, the analyst gains a degree of certainty about its probability of generating relevant items, since its variance, equal to

$$\frac{(\alpha_s + y_{s,t})(\beta_s + n_{s,t} - y_{s,t})}{(\alpha_s + \beta_s + n_{s,t})^2(\alpha_s + \beta_s + n_{s,t} + 1)} = O(1/n_{s,t}),$$

decays to zero at a rate of order one over the number of explorations at that source.

Each time period the analyst has to decide which sources to explore, in a way that allows her or him to balance the sources with high uncertainty about p_s (i.e., $\alpha_s + \beta_s + n_{s,t}$ small) against those that are likely to yield relevant items (i.e., those with $\alpha_s + y_{s,t} \gg \beta_s + n_{s,t} - y_{s,t}$). Intuitively, the analyst should choose to explore the items from the q sources with the largest chance of generating relevant items.

Recently, it has been shown that an approach known as Thompson Sampling [24] (referred to as TS throughout this work) achieves the best learning rate for this situation from a theoretical standpoint, as indicated by [25]. Our model is very similar to TS, with one main difference: whereas in TS, one can only sample a single source at a time, here, we can sample q sources per time period. TS examines items from sources with the greatest probability of generating a relevant item. This is done in two steps, first, by drawing a random p from the posterior distribution of each source and, second, by exploring items with the largest values of p drawn in the first stage. We summarize these steps as follows:

1. Draw a sample $\hat{p}_s$ from each source s from a Beta$(\alpha + y_{s,t}, \beta + n_{s,t} - y_{s,t})$. and sort the values in increasing order $\hat{p}_{(1)}, \hat{p}_{(2)}, \ldots, \hat{p}_{(S)}$.
2. From q sources corresponding to $\hat{p}_{(S-q+1)}, \ldots, \hat{p}_{(S)}$, draw a sample x_s from Bernoulli(p_s).

Agrawal and Goyal [24] have shown, for $q = 1$, the expected regret

$$E[\text{Regret}(T)] = \sum_{t}^{T} E[\max_{s}\{p_s\} - p_{s(t)}],$$

where $p_{s(t)}$ is the source sampled at time t, grows like

$$E[\text{Regret}(T)] = O\left[\left(\sum_{s=1, s\neq s^*}^{S} \frac{1}{(\max_s\{p_s\} - p_i)^2}\right)^2 \log T\right], \tag{3.1}$$

where $s^* = \arg\max_s p_s$, a function $f(T) = O(\log T)$ if $|f(T)| \leq k \log T$ for all $T > 0$ and some $0 < k < \infty$. In words, the expected difference between the total reward gained by exploring the best source if we knew all the reward probabilities p_s in advance, and the total reward obtained by the previous algorithm (that is, the expected regret), grows at order $O(S \log T)$. This implies that the average regret goes to zero, meaning that learning occurs. Lai and Robbins [25] show that this learning rate is optimal, in the sense that it is not possible to have the regret grow slower than $O(\log T)$. Notably, the dominating term driving the $O(\cdot)$ rate of growth in the regret is one divided by the square of the smallest difference between the best and second best sources. This is because the algorithm takes a long time to find the best source when its probability of yielding a relevant item is similar to that of the second best source.

We view q as a decision variable for the intelligence organization. In this setting, the regret at time t is the (random) difference in reward obtained by drawing a sample from each of the best q sources and the reward attained by exploring sources $p_{1,t}, \ldots, p_{q,t}$, where $p_{1,t}, \ldots, p_{q,t}$ are the probability of yielding a relevant item for sources selected to explore at time t.

Thus, the expected regret is

$$E[\text{Regret}(T)] = \sum_{t}^{T} E\left[\sum_{s=S-q+1}^{S} p_{(s)} - \sum_{s=1}^{q} p_{s,t}\right],$$

where $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(S)}$.

Because the intelligence organization typically pools its resources, the value of q (i.e., a measure of the resources devoted to the request for information under consideration) can change over time, but its average value is upper bounded. This *relaxation* motivates the following question. How should q change over time as subject to a bound on its mean value? What is the associated risk with any given q? How should we adjust q in future time periods? These questions are the subject of Chapter 4.

3.3 Parameter Estimation after the Data Is Available

In some cases, it makes sense for the analyst to assume that the items from a source come from a mixture of distributions, each distribution corresponding to a particular component (e.g., age group). More precisely, the analyst assumes that p_s has a mixture distribution,

$$p_s \sim \sum_{i=1}^{n} w_i f_i(\cdot),$$

where n is the number of components, w_i and $f_i(\cdot)$ are weight and density functions of ith component, respectively, with the constraint $\sum_i w_i = 1$.

This is often complicated by the fact that the source-class may not be observable; that is, the analyst has a collection of zeros and ones from a source, but does not know the component of each item. To handle this scenario, we use the Expectation Maximization (EM) algorithm. EM can be used to estimate the weights and the parameters of the compound distributions when the information to which compound an observation belongs to is missing [26]. Employing EM we can do the following:

- Estimate the sizes of each component (weights)
- Estimate the parameters of each component

- Estimate the component of each observation

EM iteratively repeats in two steps: expectation and maximization. In the first step, conditional expectation is calculated. In the second step, the parameters that maximize the expectation are found [27]. It has been proved that the estimates converge to the true parameters [27]. For this purpose, we use the *betareg* package of R, which has been developed for beta mixture models. For the details of implementation for beta mixtures within the package, see Grun et al. [28].

For a formal explanation of EM, let X be the vector of observations from the mixture distribution, and let Z be the vector indicating the compounds that are unknown (hidden). θ_t is the vector of parameters to be estimated in iteration t. Then:

Expectation Step: Determine the conditional expectation $E_{Z|X,\theta_t}\left(P(X,z|\theta)\right)$

Maximization Step: Find the θ that maximizes this expectation

In our case, the densities $f_i(\cdot)$ are beta with parameters α_i and β_i. Despite the fact that we show the results for beta mixtures, which is more convenient for our purpose, EM for other mixture models can be implemented, or available packages can be employed.

Expectation Step: Calculate

$$t_i = E(logp_i|X_i = x_i) = \Psi(\alpha_{old} + x_i) - \Psi(\alpha_{old} + \beta_{old} + N),$$

$$s_i = E(log(1-p_i)|X_i = x_i) = \Psi(\beta_{old} + N - x_i) - \Psi(\alpha_{old} + \beta_{old} + N),$$

for $i = 1, \ldots, n.$ and N is the number of trials.

Karlis [29] proposes a scheme to update the current estimates of the parameters of Beta distribution as follows:

Maximization Step: Make an one-step ahead Newton Raphson iteration for ML estimation of a beta density using the expectations of the E-step. To do so, calculate

$$\bar{t} = \frac{\sum_i = 1^n t_i}{n},$$

$$\bar{s} = \frac{\sum_i = 1^n s_i}{n},$$

and, then, update the estimates as

$$\alpha_{new} = \alpha_{old} - \frac{\Psi(\alpha_{old}) - \Psi(\alpha_{old} + \beta_{old}) - \bar{t}}{\Psi_3(\alpha_{old}) - \Psi_3(\alpha_{old} + \beta_{old})},$$

$$\beta_{new} = \beta_{old} - \frac{\Psi(\beta_{old}) - \Psi(\alpha_{new} + \beta_{old}) - \bar{s}}{\Psi_3(\beta_{old}) - \Psi_3(\alpha_{new} + \beta_{old})}.$$

$\Psi(\cdot)$ denotes the digamma function:

$$\Psi(x) = \frac{d}{dx} ln(\Gamma(x)) = \frac{\Gamma'(x)}{\Gamma(x)}.$$

$\Psi_3(\cdot)$ denotes the trigamma function:

$$\Psi_3(x) = \frac{d}{dx}\Psi(x) = \frac{d^2}{dx^2} ln\Gamma(x),$$

gamma function for positive real numbers:

$$\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt.$$

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 4:
Analysis

As our main contribution in this thesis to the ongoing effort to collect, process, and analyze data for intelligence, we extend the TS framework, so more than one item can be explored each time period. As shown in Chapter 3, the capacity q is a decision variable for the intelligence organization. While there is no variable cost for exploring a source, there is a limit to how large q can be in each time period or on average. The rationale for this is that the resources of the intelligence organization (e.g., technological or human) are viewed as a fixed cost, to be used at will.

Exploring more than one source per time period triggers a change in the interpretation of *expected regret*. In our case, we calculate expected regret as follows:

$$E[\text{Regret}(T)] = \sum_{t}^{T} E\left[\sum_{s=S-q+1}^{S} p_{(s)} - \sum_{s=1}^{q} p_{s,t}\right],$$

as shown in Chapter 3. In this setting, the regret at time t is the (random) difference in reward obtained by drawing a sample from each of the best q (unknown) sources and the reward attained by exploring sources $p_{1,t},\ldots,p_{q,t}$, where $p_{1,t},\ldots,p_{q,t}$ are the probabilities of yielding a relevant item for sources selected to explore at time t.

In this thesis, we sample q different sources each time period, but there are other possible ways to sample q sources, such as sampling q items from the same source. However, the operational settings in which such an approach is possible are limited and, thus, omitted from consideration here.

In this chapter, we explain the algorithm in detail, and we interpret the results obtained from the simulations. First, in Section 4.2, we look at the performance of the algorithm when the p_s are from either a pure or mixture population. The goal in that section is to get a feel for the algorithm's behavior. Does the expected regret grow logarithmically in time? How is the expected regret affected by prior knowledge?

Then, in Section 4.3, we treat q as a decision variable, allowing it to change over time. We use a normal approximation to find the capacity required to detect a certain percentage of the relevant items. While relatively trivial, this is another contribution of our analysis.

In Section 4.4, we develop the notion of risk. We view *risk* as the expected fraction of relevant sources unexplored in a time period. We analyze the trade off between the resources allocated against the risk assumed.

In some cases, it makes sense for the analyst to have a prior for p_s that is a mixture distribution, for instance, when intelligence agencies categorize population into an *innocent* group and a *dangerous* group, or when a satellite takes pictures in areas of interest and past non-interest. Hence, it is important to include such scenarios in our analysis. Thusly motivated, in Section 4.5, we employ the EM algorithm in situations wherein the intelligence analyst has historical data that stem from a mixture distribution, and can be used for a prior for the parameters $(p_s)_{s=1}^{S}$. While this section is a bit disconnected from the other parts of this chapter, we view it as important, because in most realistic situations, there is past data available.

4.1 Algorithm

In this section we discuss the algorithms employed for the analysis. Because the intelligence organization typically pools its resources, the value of q (i.e., a measure of the resources devoted to the request for information under consideration) can change over time, but its average value is upper bounded. This *relaxation* motivates the following questions. How does q change over time, subject to a bound on its mean value? What is the associated risk with any given q? How should we adjust q in future time periods? These questions are the subject of Chapter 4.

We assume that the probability that source s generates a relevant item (p_s) comes from some arbitrary distribution with support over $[0,1]$, meaning that for source s nature gets a sample p_s, which then is used to generate the rewards from a Bernoulli distribution with parameter p_s. The analyst updates the beta parameters as discussed in Chapter 3. From the analyst's standpoint, he or she has a prior distribution for p_s, which does not necessarily coincide with the true underlying distribution of p_s. The analyst faces a number of different

scenarios, depending on whether there is historical data, on the assumptions he or she makes about the true distribution of p_s and on the type of information revealed:

1. There is no prior information about p_s, so the analyst assumes a uniform distribution over (0,1), i.e., Beta(1,1). The p_s are drawn from some arbitrary distribution, for example, a mixture of a Beta and a triangular density over $[0,1]$, as shown in Figure 4.6 Figure 4.7.
2. Nature sets the distribution of p_s as Beta, and the analyst knows this. If there is no historical data, we end as in the first scenario (this is the case of Figure 4.2). If there is historical data available, the analyst estimates α_s and β_s using maximum likelihood estimation.
3. The analyst knows that nature issues a mixture of Beta distributions for p_s; for instance, $p_s \sim .9\,\text{Beta}(1,5) + .1\,\text{Beta}(5,1)$. The analyst observes the sequence of zeros and ones from each source. However, the analyst does not know from which of the components, either Beta(1,5) or Beta(5,1) in the preceding example, the data originates, nor does he or she know the mixing probabilities (.1 and .9 in the example). In this case, the analyst uses historical data along with the EM algorithm to estimate the Beta parameters (α_s and β_s that appear in the algorithm of Figure 4.1, in 1(b)iii) as well as the mixing probabilities. This scenario is discussed in Section 4.5.

The main algorithm is shown in Figure 4.1.

1. Initialization:
   (a) Set $t = 0$,
   (b) For each source s,
      i. Draw p_s from a some arbitrary density.
      ii. Set $y_{s,t} = 0$ and $n_{s,t} = 0$.
      iii. Set $\alpha_s = 1, \beta_s = 1$ (or use EM to estimate them)
2. Draw a sample $\hat{p}_s$ from each source s from a Beta$(\alpha_s + y_{s,t}, \beta_s + n_{s,t} - y_{s,t})$. and sort the values in increasing order $\hat{p}_{(1)}, \hat{p}_{(2)}, \ldots, \hat{p}_{(S)}$.
3. From q sources corresponding to $\hat{p}_{(S-q+1)}, \ldots, \hat{p}_{(S)}$, draw a sample x_s from Bernoulli(p_s).
4. Set $y_{s,t} = y_{s,t-1} + 1$ if $x_s = 1$ for sampled sources.
5. Set $n_{s,t} = n_{s,t-1} + 1$ for all q sampled sources.
6. Set $t = t + 1$.
7. Go back to 2.

Figure 4.1: Pseudocode of the Algorithm for Arbitrary Priors

4.2 Results

In this section we analyze the performance of the algorithm through numerical experiments. We consider two scenarios. In the first scenario, p_s are sampled from one population ($s = 1, \ldots, S$, and S is the number of sources). In the second scenario, p_s are sampled from two populations. We show the results in terms of cumulative regret and regret per time period.

4.2.1 When p_s are Sampled from One Population

In this subsection, we treat the simplest scenario, one in which all the parameters p_s are independent and identically distributed from a Beta distribution. In particular, we assume that $p_s \sim$ Beta$(0.02, 0.18)$. The density is shown in Figure 4.2. This assumption implies that sources have mostly either low or high p_s values and rarely intermediate values, capturing situations wherein the population rarely produces relevant item, but those items that are relevant come from a small subset of the population.

The regret realized in each time period is shown in Figure 4.4 and the cumulative regret appears in Figure 4.3, with 95% confidence interval bands obtained by 200 simulation replications ($q = 20$ and $S = 100$). The per-period regret decays toward zero as learning becomes realized, and the sources with the largest p_s become more likely to be sampled.

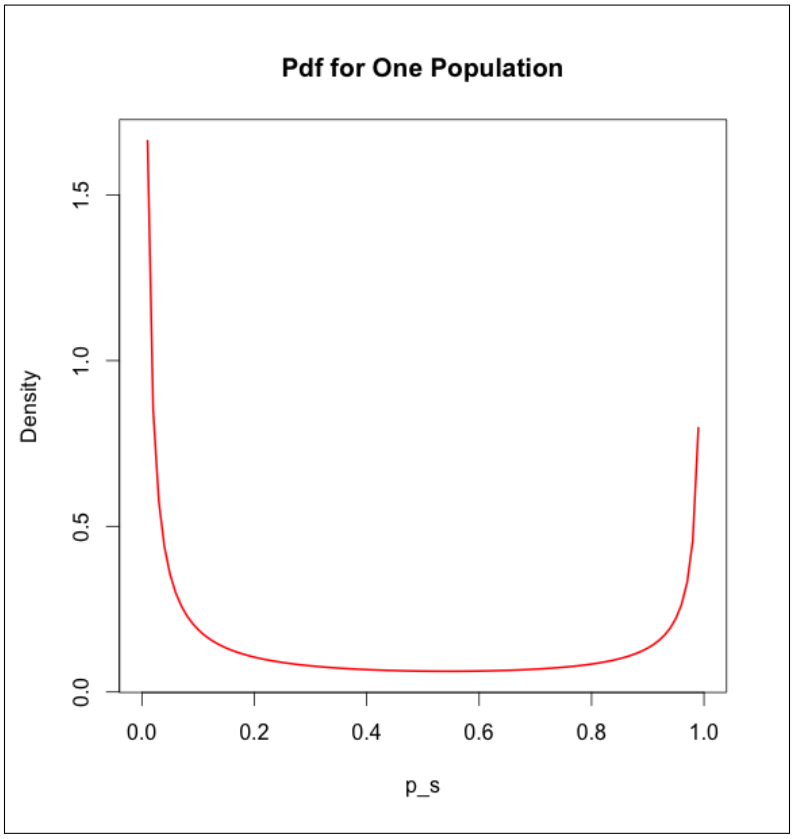

Figure 4.2: Pdf of the Distribution of the p_s

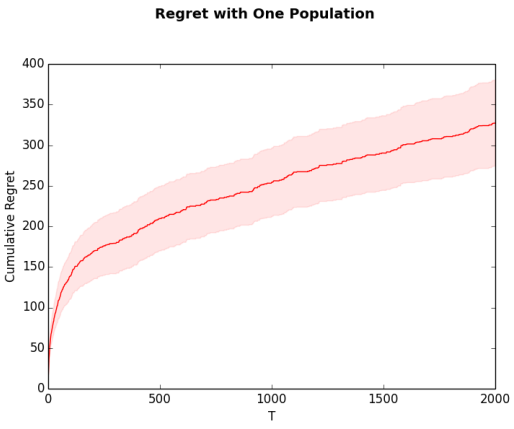

Figure 4.3: Cumulative Regret for One Population

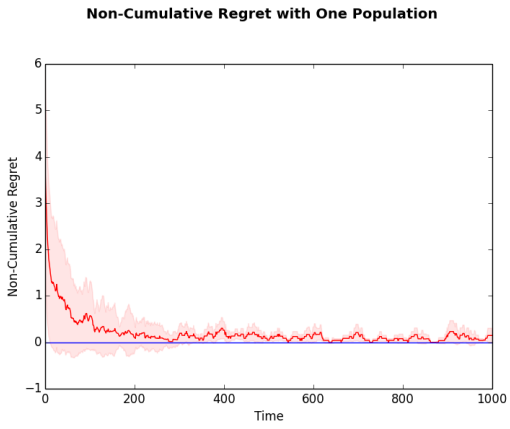

Figure 4.4: Non-cumulative Regret for One Population

In Figure 4.5, we plot the average cumulative regret over 200 sample paths as a function of $\log t$ as well as the 95% confidence interval bands. The motivation is Equation 3.1: the expected cumulative regret grows at order $\log t$. The average cumulative regret does not appear to grow linearly for $t = \exp(6) \approx 400$, but it does thereafter. This is not in disagreement with Equation 3.1, as it only applies as t grows larger. We believe this is because we sampled $q = 20$ sources per period while Equation 3.1 applies to the case in which $q = 1$. In other words, we accrue regret over the poor sources that get sampled among the 20 selected sources per time period.

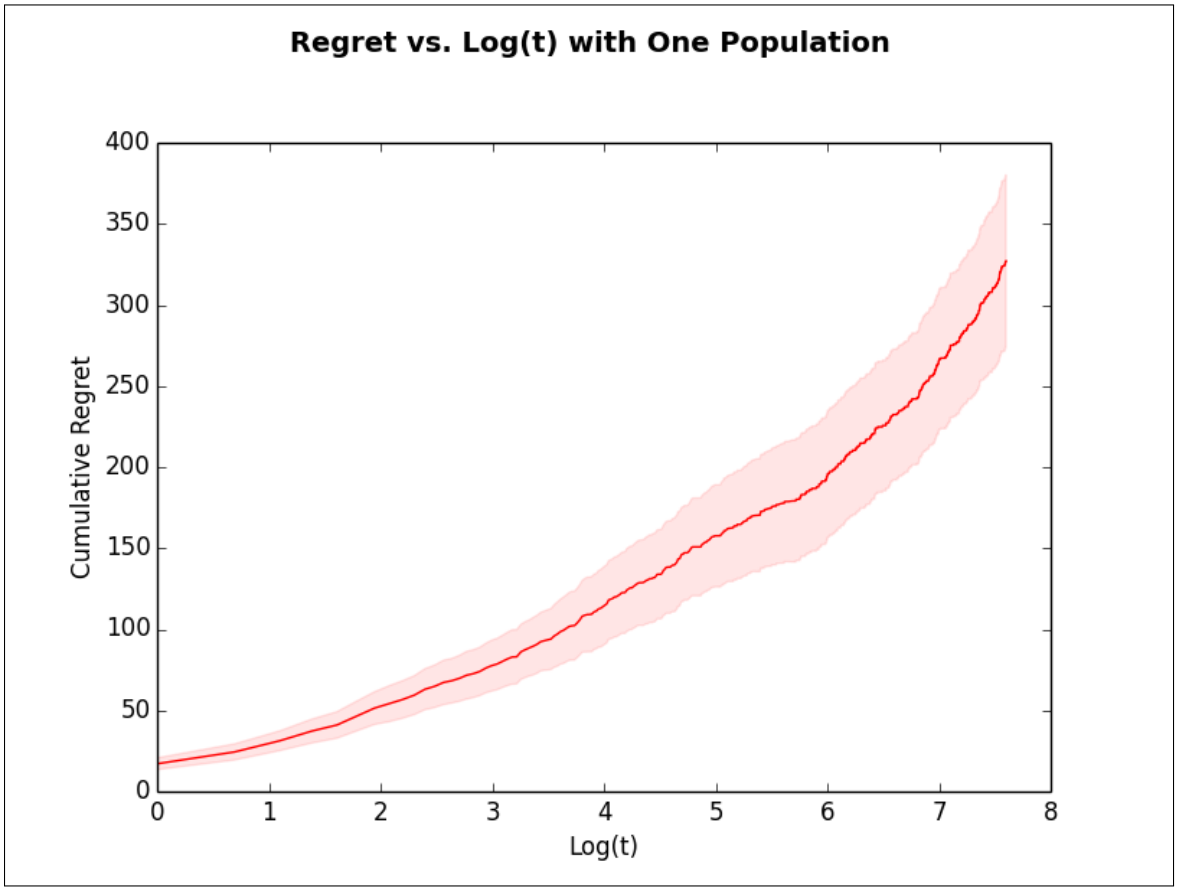

Figure 4.5: Regret for One Population

4.2.2 When p_s are Sampled from Two Populations

In this subsection, we consider a mixture distribution for the priors, with known mixing probabilities. This will be relaxed later on, when we use the EM algorithm to estimate the mixing probabilities. This would be appropriate for situations in which a subpopulation group is viewed as a source, with the mixing probability representing the weight of the subpopulation in the broader population. In particular, we assume that the probability of producing a relevant item (p_s) for 99% of the sources are from Beta(0.05, 0.95) distribution and for 1% of the sources from Triangular(0,1,1) distribution.

The initial prior is Beta(1,1), but as the number of rewards observed grows larger, its impact becomes relatively smaller and the updated values of α and β (c.f., Line 2 of the algorithm in Figure 4.1) eventually force the algorithm to emphasize exploring from the best sources.

The densities of both distributions are shown in Figure 4.6 and Figure 4.7. This assumption is reasonable for screening communication items because most of the people are innocent and have a low probability of generating a relevant item (e.g., e-mail or phone conversation), while a very small percentage of people (e.g., criminals or terrorist suspects) have a higher probability relevant items. Regret in each time period is shown in Figure 4.9, and cumulative regret is shown in Figure 4.8. As seen in the figures, the mixture of source populations has created a similar regret pattern to the one population scenario.

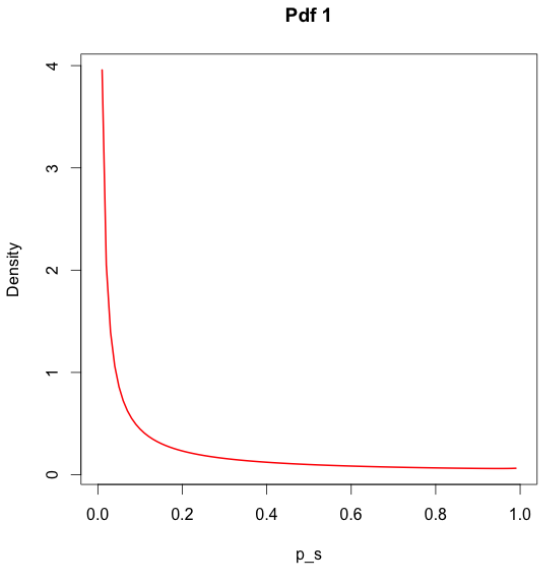

Figure 4.6: Pdf 1

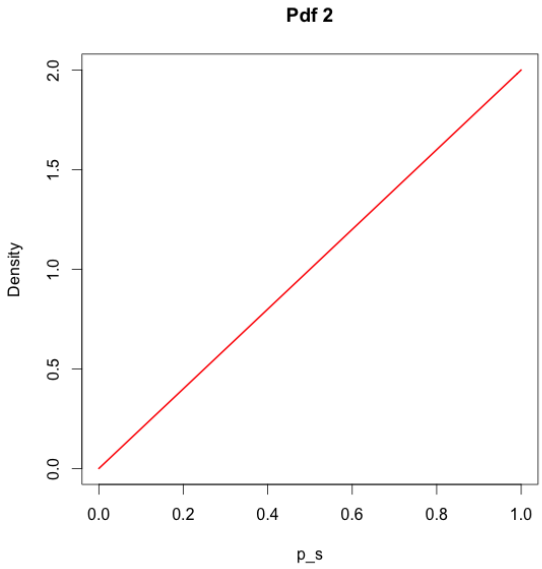

Figure 4.7: Pdf 2

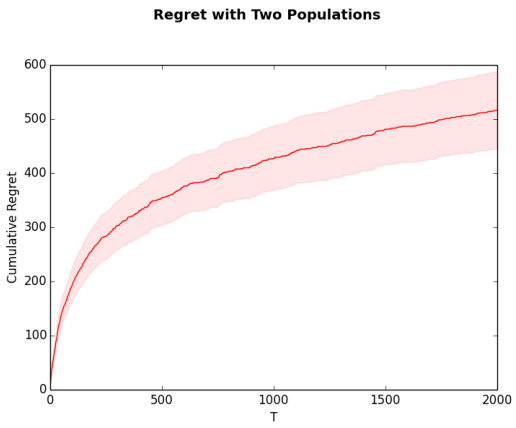

Figure 4.8: Cumulative Regret for Two (Mix) Populations

Figure 4.9: Non-cumulative Regret for Two (Mix) Populations

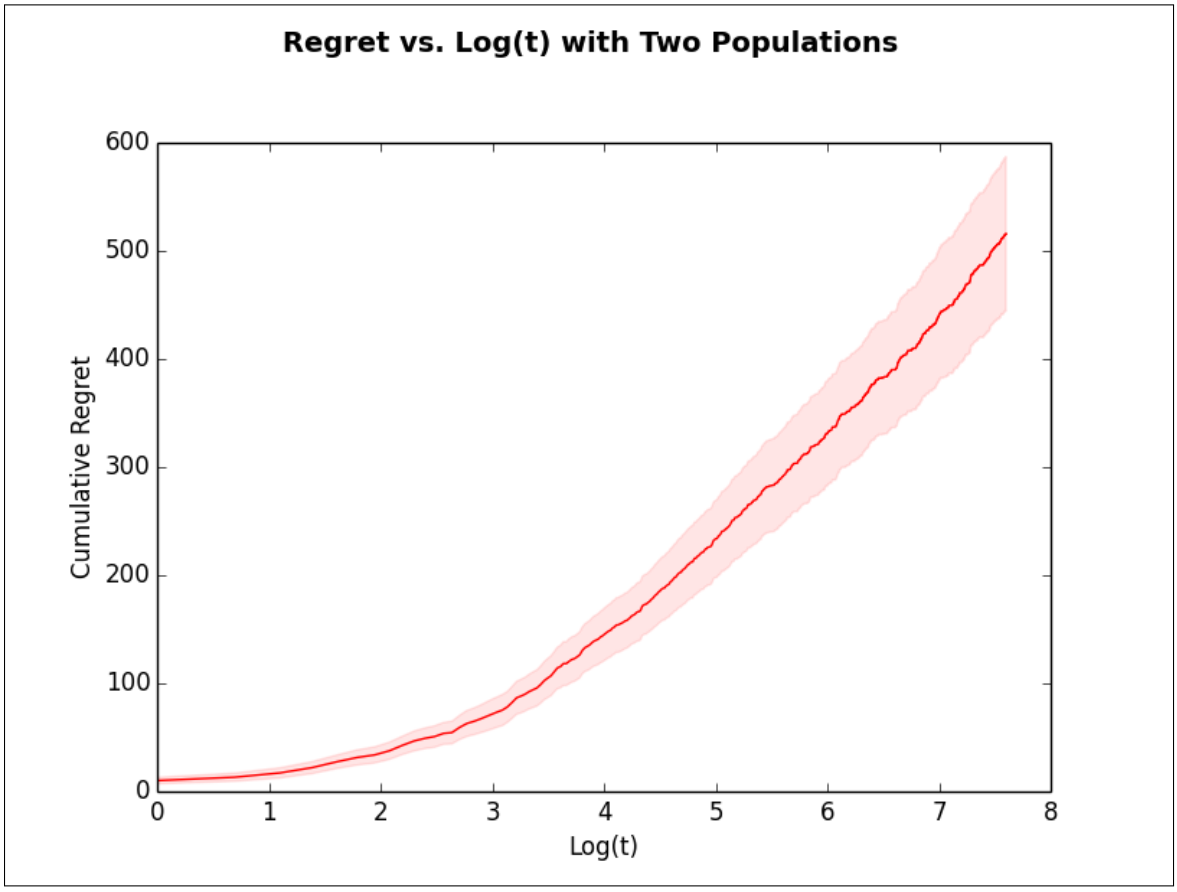

Figure 4.10: Regret for Two Populations

As shown in Figure 4.10, we plot the average cumulative regret in terms $\log t$, including 95% confidence bands, based on 200 replications, for $q = 20$ sources sampled per time period. As with Figure 4.5, the expected regret appears to grow linearly only for values of t larger than $e^4 \approx 60$. In this case, we have the extra difference, relative to Equation 3.1, that the prior distribution of the sources is a mixture of a beta and a triangular density but unknown and initialized as Beta(1,1) in the simulation.

4.3 Determining the Number of Sources to Sample (q)

An output of a single replication of the simulation for $S = 100$ sources during which the analyst samples $q = 10$ sources per time period is shown in Figure 4.11 (p_s are sampled as described in Subsection 4.2.2). We observe that after an exploration period, the regret

per time period stabilizes under the red curve, which shows the real number of relevant items throughout the time periods. The blue line, represents the number of relevant items discovered.

Clearly, the larger the number of sources explored per time period means the larger the expected number of relevant items discovered. Accordingly, the blue line approaches the red line when q is increased from 10 (Figure 4.11) to 20 (Figure 4.12). Another observation is that the increase in q almost cuts in half the number of time periods required for the curve to stabilize (from about 100 to 50).

This raises the question: How does the expected regret change as a function of the number of sources explored per time period (q)? Our goal in this subsection is address this issue.

In Figure 4.13, we show the regrets obtained for different q values ($q = 1, 10, 20, 40, 60, 80$). After the exploration phase, the regret grows like a constant C times the logarithm of T, wherein the proportionality constant depends on q. As shown in Equation 3.1, the growth's constant depends on how similar the best source is to the other sources when $q = 1$. However, when q takes on other values, we conjecture that it depends on how similar the best q sources are to the rest of the sources. We observe that the growth constant for $q = 1$ is greater than that for $q = 10$. The rationale for this is that we sample $q > 1$ sources per period, so that there is extra regret due to the poor sources that get sampled among the q selected.

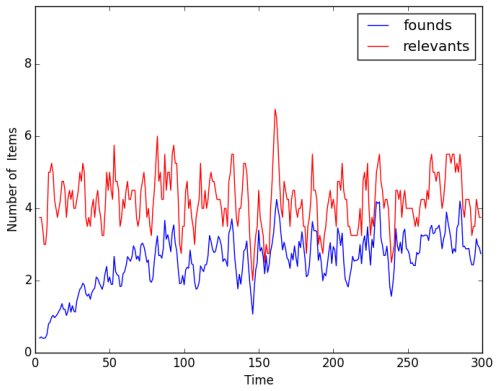

Figure 4.11: $q = 10$

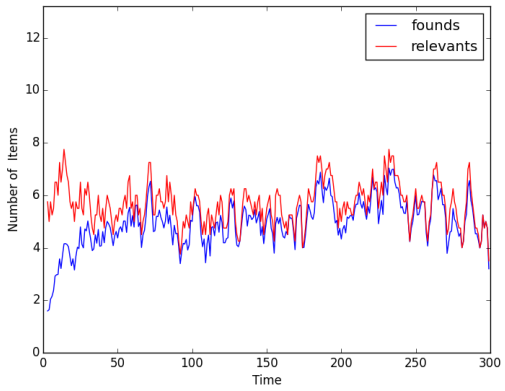

Figure 4.12: $q = 20$

Let Y_t be the number of relevant items in period t across all sources. The posterior distribution of Y_t given the exploration to date is a sum of S independent Bernoulli random variables, $\sum_s X_{s,t}$. Each $X_{s,t}$ is Bernoulli with parameter $p_{s,t} = \alpha_{s,t}/(\alpha_{s,t} + \beta_{s,t})$, for $\alpha_{s,t} = \alpha_s + y_{s,t}$ and $\beta_{s,t} = \beta_s + n_{s,t} - y_{s,t}$. The interpretation, as in Chapter 3, is that $\alpha_{s,t}$ is the prior parameter α_s plus the number of relevant items explored from source s to-date, and $\beta_{s,t}$ is the initial β_s plus the number of irrelevant items in source s in the initial t periods. Therefore,

$$E[Y_t|\text{exploration in periods } t = 1,\ldots,t-1] = \sum_{s=1}^{S} \frac{\alpha_{s,t}}{\alpha_{s,t} + \beta_{s,t}}, \tag{4.1}$$

$$\text{Var}(Y_t|\text{exploration in periods } t = 1,\ldots,t-1) = \sum_{s=1}^{S} \text{Var}(X_{s,t}).$$

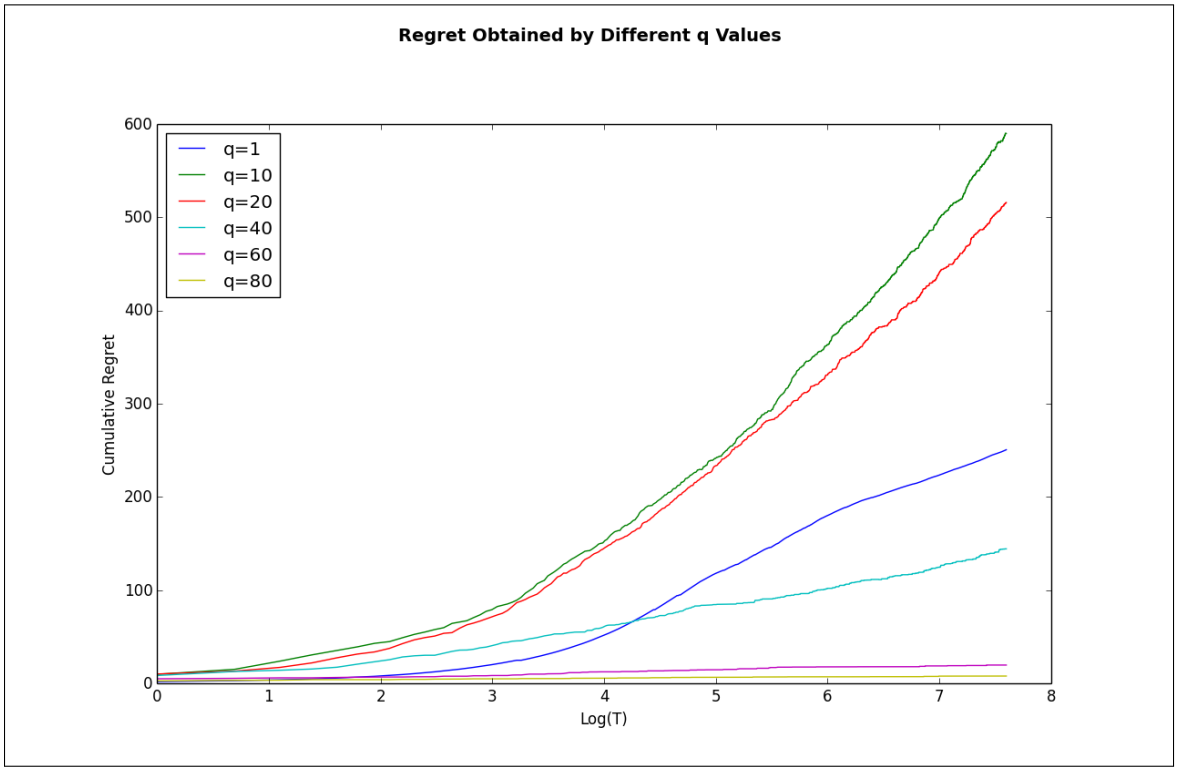

Figure 4.13: Regret Obtained by Different q Values

Since $p_{s,t}$ are also random, we may employ the total variance formula for $\text{Var}(X_{s,t})$,

$$\begin{aligned}
\text{Var}(X_{s,t}) &= E[\text{Var}(X_{s,t}|p_{s,t})] + \text{Var}(E[X_{s,t}|p_{s,t}]) \\
&= E[(p_{s,t})(1-p_{s,t})] + \text{Var}(p_{s,t}) \\
&= E[p_{s,t}] - E[p_{s,t}^2] + \frac{\alpha_{s,t}\beta_{s,t}}{(\alpha_{s,t}+\beta_{s,t})^2(\alpha_{s,t}+\beta_{s,t}+1)} \quad \text{(the } p_{s,t}\text{'s are Beta distributed)} \\
&= \frac{\alpha_{s,t}}{\alpha_{s,t}+\beta_{s,t}} - \frac{\alpha_{s,t}(\alpha_{s,t}+1)}{(\alpha_{s,t}+\beta_{s,t})(\alpha_{s,t}+\beta_{s,t}+1)} + \frac{\alpha_{s,t}\beta_{s,t}}{(\alpha_{s,t}+\beta_{s,t})^2(\alpha_{s,t}+\beta_{s,t}+1)}.
\end{aligned}$$

We conclude that

$$\begin{aligned}
&\text{Var}(Y_t|\text{exploration in periods } t=1,\ldots,t-1) \\
&= \sum_{s=1}^{S} \frac{\alpha_{s,t}}{\alpha_{s,t}+\beta_{s,t}} - \frac{\alpha_{s,t}(\alpha_{s,t}+1)}{(\alpha_{s,t}+\beta_{s,t})(\alpha_{s,t}+\beta_{s,t}+1)} + \frac{\alpha_{s,t}\beta_{s,t}}{(\alpha_{s,t}+\beta_{s,t})^2(\alpha_{s,t}+\beta_{s,t}+1)}. \quad (4.2)
\end{aligned}$$

Observe that the posterior variance of the total number of relevant items by period t decays toward the variance of a sum of Bernoulli random variables, considered the systemic variance, as exploration eliminates the variance due to the uncertainty about the p_s's.

The central limit theorem suggests that the total number of relevant items at time t given by Y_t is approximately normally distributed when the number of sources is large. This motivates the study of two different scenarios. In the first case, we assume that the number of sources to sample has been decided upfront and must remain constant thereafter. In the second scenario, the analyst can change the number of sources to sample dynamically. In both cases, the goal of the analyst is to sample as many sources as needed, so he or she has a 95% probability of capturing the reward of a source; that is, on average, the analyst collects 95% of the total rewards.

Thus, we use a normal approximation to provide an upper bound with c confidence:

$$\text{Upper Bound} = E[Y_0] + \Phi^{-1}(c)\sigma_{Y_0}, \tag{4.3}$$

where $E[Y_0]$ is the sum of all the prior means, and σ_{Y_0} is the sum of the standard deviation of all the priors at time zero.

Using these equations, we are able to provide an upper bound for Y_t with confidence level c. Then, it will be reasonable for q to be greater than or equal to this bound. Here, c can be regarded as another decision variable.

In order to capture all relevant items after the exploration phase, the allocated capacity has to be greater than or equal to the number of relevant items in each iteration ($q \geq Y_t$). Because Y_t is a random variable, it is safe to choose a value for q that is greater than or equal to the value obtained by the upper bound in Equation 4.3.

Corresponding upper bounds are shown in Table 4.1. In other words, a minimum number of sources explored, q, for certain number of sources S, with 0.99 confidence level (c). Although these bounds are for prior Beta(0.02,0.18), providing similar bounds for any distribution of the p_s seems straightforward.

Table 4.1: Minimum q Values Required

S	q
10	4
20	6
50	10
100	17
500	66
1000	123
10000	1070

If the analyst must choose the value of q in advance, he or she should follow the recommendation above. On the other hand, since α_s and β_s do change over time as relevant and non-relevant items are examined, the analyst may change the value of q dynamically.

Next, we analyze the second scenario, wherein the analyst can adjust the number of sources explored (q) dynamically according to the posteriors. If the analyst has a capacity that is at least as large as the number of sources, then there is no risk of missing any relevant items. However, some capacity may become idle or useless as the sources are explored over time. Hence, instead of allocating a constant capacity for all time periods, it is more efficient to adjust the capacity q as learning occurs.

The idea is similar to the first scenario, namely, to use a normal approximation and to compute the percentile of the number of relevant items using the posterior mean and variances, as in Equations 4.2 and 4.1. More precisely, we determine the value of the upper bound:

$$\begin{aligned}&\text{Upper Bound}\\ &= E[Y_t|\text{exploration in } t=1,\ldots,t-1] + \Phi^{-1}(c)\sqrt{\text{Var}(Y_t|\text{exploration in } t=1,\ldots,t-1)},\end{aligned} \tag{4.4}$$

where $E[Y_t|\text{exploration in periods } t=1,\ldots,t-1]$ and $\text{Var}(Y_t|\text{exploration in } t=1,\ldots,t-1)$ are as defined above. Notably, at time t, the posterior parameters for source s are the initial α_s plus the number of relevant items detected to date and β_s plus the number of non-relevant items up to time t.

We summarize the algorithm when the capacity q is selected dynamically, as shown in Figure 4.16. We use the posterior mean plus two standard deviations ($E[Y_t] + 2 \times \sigma_{Y_t}$) to update q. In simulation, we used Beta(0.02,0.18) to sample p_s. However, assuming no knowledge, we initialized the prior distributions for each source as Beta(1,1). Output, for one replication of the simulation, is shown in Figure 4.14, and the corresponding change in q is shown in Figure 4.15. The red line represents the actual number of relevant items generated by all sources throughout the time horizon, and the blue line represents the number of relevant items discovered by the algorithm. It can be observed that with dynamic q, the algorithm captures almost all of the relevant items, even in the exploration phase of the algorithm. In this scenario, the capacity q is sufficient to explore almost all of the expected (with respect to the posterior distribution) relevant items.

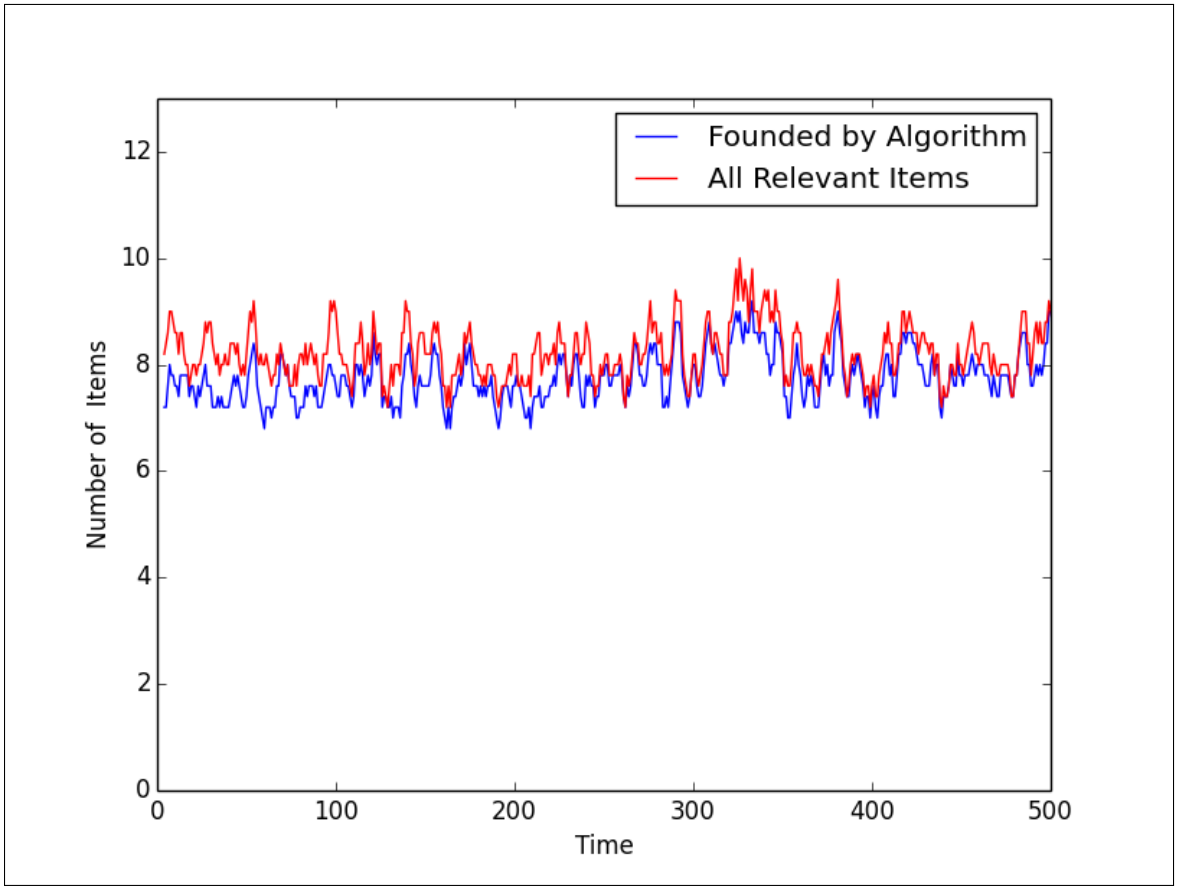

Figure 4.14: Learning with Dynamic q

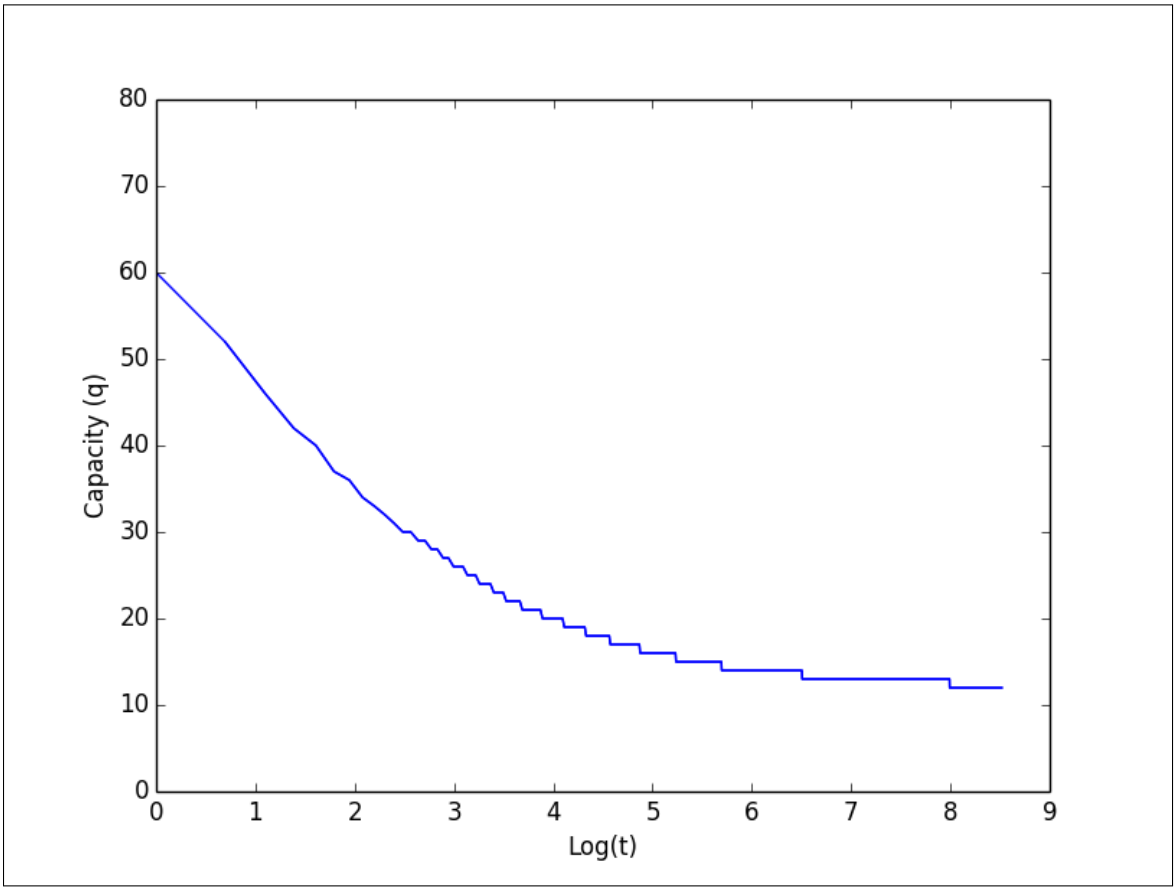

Figure 4.15: Change in q

Regarding the behavior of the capacity q in Figure 4.15, its value, as determined by Equation 4.4, decays as the uncertainty about the values of the p_s probabilities are revealed. As mentioned above, as time increases the posterior variance of Y_t converges to the sum of S Bernoulli variances. The closer the initial prior is to the true p_s values means the shorter the time until the capacity q stabilizes. We view the uniform initial prior as a worst-case scenario absent any “wrong" knowledge.

1. Initialization:
   (a) Set a and b.
   (b) **Set c**
   (c) Set $t = 0$,
   (d) For each source s,
      i. Draw p_s from a Beta(a,b).
      ii. Set $y_{s,t} = 0$ and $n_{s,t} = 0$.
      iii. Set $\alpha_s = 1, \beta_s = 1$ (if there are initial information about any source set accordingly)
2. Draw a sample $\hat{p}_s$ from each source s from a Beta$(\alpha_s + y_{s,t}, \beta_s + n_{s,t} - y_{s,t})$. and sort the values in increasing order $\hat{p}_{(1)}, \hat{p}_{(2)}, \ldots, \hat{p}_{(S)}$.
3. From q sources corresponding to $\hat{p}_{(S-q+1)}, \ldots, \hat{p}_{(S)}$, draw a sample x_s from Bernouilli(p_s).
4. Set $y_{s,t} = y_{s,t-1} + 1$ if $x_s = 1$ for sampled sources.
5. Set $n_{s,t} = n_{s,t-1} + 1$ for all q sampled sources.
6. **Calculate the Mean and Standard deviation of Y_t according to the posterior distribution at time t**
7. **Set** $q = E[Y_t] + \Phi^{-1}(c)\sigma_{Y_t}$
8. Set $t = t + 1$.
9. Go back to 2.

Figure 4.16: Pseudocode of the Dynamic Algorithm

4.4 Risk vs. Resource Allocated

Having analyzed how the capacity q should change over time, in this subsection, we inspect the effect of q on the risk of missing relevant items. In order to do so, we first define a metric to measure the risk. We define the risk in period t as the conditional expectation:

$$E[\text{fraction of relevant items } \textit{not} \text{ explored in period } t | Y_t],$$

where Y_t is the total number of relevant items in period t. For example, if nature sets the total number of relevant items at time t equal to 80, and the expected number relevant items explored by the algorithm equals 60 conditioned on the 80 relevant items, then the risk is 25%.

The expression above is difficult to compute analytically because it depends on the q sources selected by the algorithm in period t, and Y_t is unknown. This is the reason for

using Thompson sampling—it randomizes the selection of the sources to sample—by selecting the largest q samples, each drawn from the posterior distribution of p_s.

The average risk over the time horizon $t = 1, \ldots, T$ is the *grand* average over the time horizon T of the risk in each period:

$$\text{Risk}(T) = \frac{1}{T} \sum_{t=1}^{T} E[\text{fraction of relevant items } \textit{not} \text{ explored in period } t | Y_t].$$

The average risk is random, because it depends on the total number of relevant items in each period, $(Y_t)_{t=1}^{T}$, which are random and not known. The expected risk essentially unconditions the number of relevant items in each period and can be estimated by Monte Carlo simulation.

We provide a numerical example. For this purpose, after sampling p_s once as described in Subsection 4.2.2, we run the algorithm to learn about p_s 30 replications. The relation between the capacity q and the expected risk obtained is shown in Table 4.2 and Figure 4.17. In the latter, a 95% confidence interval appears in light red, centered around the sample average.

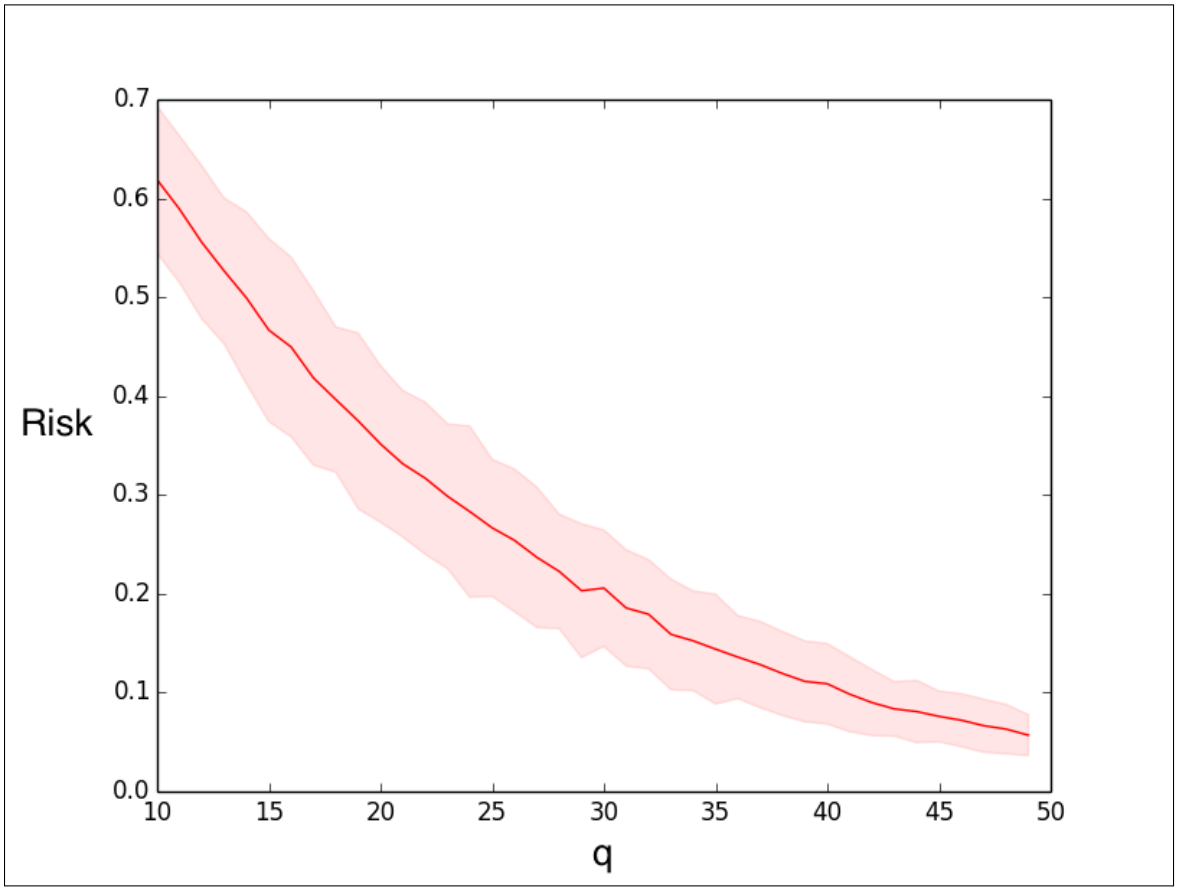

Figure 4.17: Tradeoff between Allocated Resource per Time Period (q) and the Risk

In Figure 4.18, we observe that the risk depends on both the resource allocated and the time horizon. Although we can estimate risk after truncating the initial learning period, we do not do this to penalize long learning periods. If the model is to be used for a long T, then the effect of initial exploration period naturally tends to zero. Otherwise, short time horizons force the analyst to choose a greater q to obtain the same risk level comparing to the longer T.

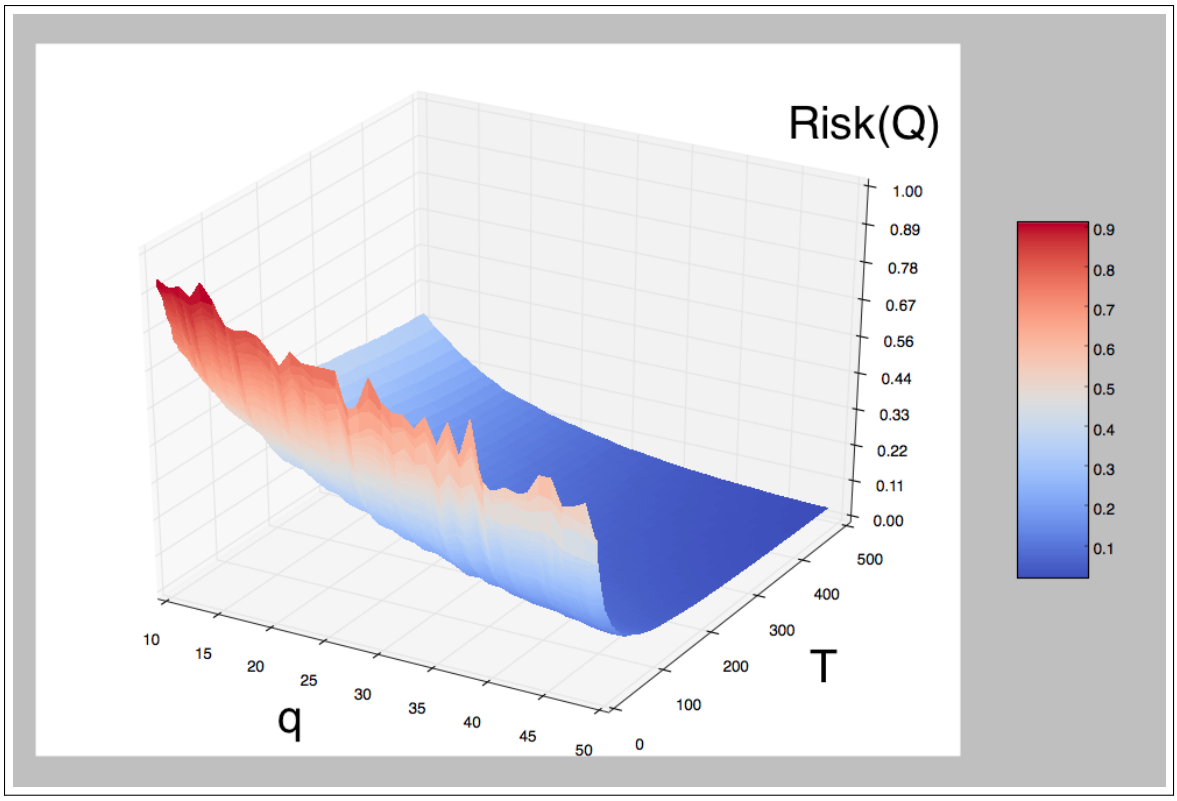

Figure 4.18: Risk according to q and Time Horizon

Table 4.2: Estimated Risk for Allocated Resources (q) (S = 100 and T = 300)

q	Risk	q	Risk	q	Risk	q	Risk
10	**0.33,**	20	**0.137,**	30	**0.06,**	40	**0.034,**
11	**0.299,**	21	**0.128,**	31	**0.06,**	41	**0.032,**
12	**0.269,**	22	**0.118,**	32	**0.054,**	42	**0.03,**
13	**0.239,**	23	**0.109,**	33	**0.051,**	43	**0.029,**
14	**0.223,**	24	**0.101,**	34	**0.048,**	44	**0.027,**
15	**0.201,**	25	**0.096,**	35	**0.047,**	45	**0.025,**
16	**0.186,**	26	**0.087,**	36	**0.044,**	46	**0.024,**
17	**0.174,**	27	**0.08,**	37	**0.04,**	47	**0.024,**
18	**0.164,**	28	**0.076,**	38	**0.039,**	48	**0.022,**
19	**0.153,**	29	**0.072,**	39	**0.034,**	49	**0.021**

4.5 Using Posteriors to Learn about the Distribution of p_s

As previously mentioned, it is common to be in situations with historical data for relevant and non-relevant items, whereby each p_s is sampled from a mixture distribution. The issue for the analyst is that the membership of a p_s to either one of the two components is unobservable, but the zeros and ones can be ascribed to a particular source. It is in situations such as these that the EM algorithm is applicable. In this section, we provide two numerical examples for the method. The main idea is to treat the means of posterior beta distributions as if they are samples from the unknown distribution of p_s.

For simulation, we assume that p_s come from a mixture distribution,

$$0.1 \times Beta(1,5) + 0.9 \times Beta(5,1)$$

and generated p_s for 100 sources. We set $q = 20$ and $T = 2000$.

True density and sampled p_s are shown in Figure 4.19. The fitted density and mean posteriors are shown in Figure 4.20. Parameters estimated by the algorithm are shown in Table 4.3. Since we gave comparatively small number of sources (10% expected out of 100) from the first component, Beta(1,5), it was very hard to estimate its true parameters and weight. However, its effect on the fitted distribution is low compared to component 1. When we compare Figure 4.19 and Figure 4.20, we can conclude that the EM algorithm generated a density that is similar to the true one.

Another assumption is that

$$0.25 \times Beta(1,5) + 0.75 \times Beta(5,1)$$

and generated p_s for 1000 sources. We set $q = 200$. The resulting fitted densities of the mixture population for certain time points ($T = 2000, 4000, 6000, 8000$) are shown in Figure 4.21. We observe that the fitted density approaches to the true density (also shown in the figure) as data becomes available.

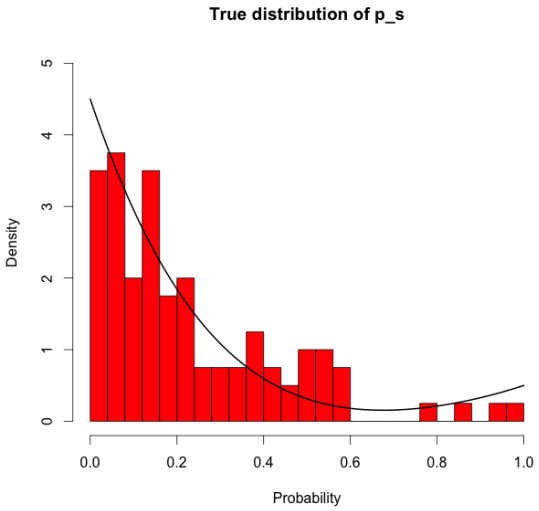

Figure 4.19: Assumed (True) Distribution and Histogram of 100 Samples

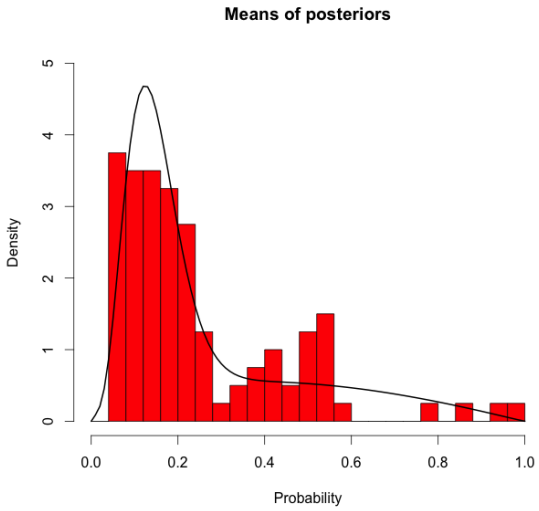

Figure 4.20: Fitted Distribution and Histogram of 100 Means of Posteriors

Table 4.3: Comparison of the True Weights and Parameters to Those Fitted

		True	**Estimated**
Component 1	Alpha	1	4.9
	Beta	5	28.8
	Mean	0.16	0.14
	Weight	0.9	0.63
Component 2	Alpha	5	1.65
	Beta	1	2.06
	Mean	0.84	0.44
	Weight	0.1	0.37

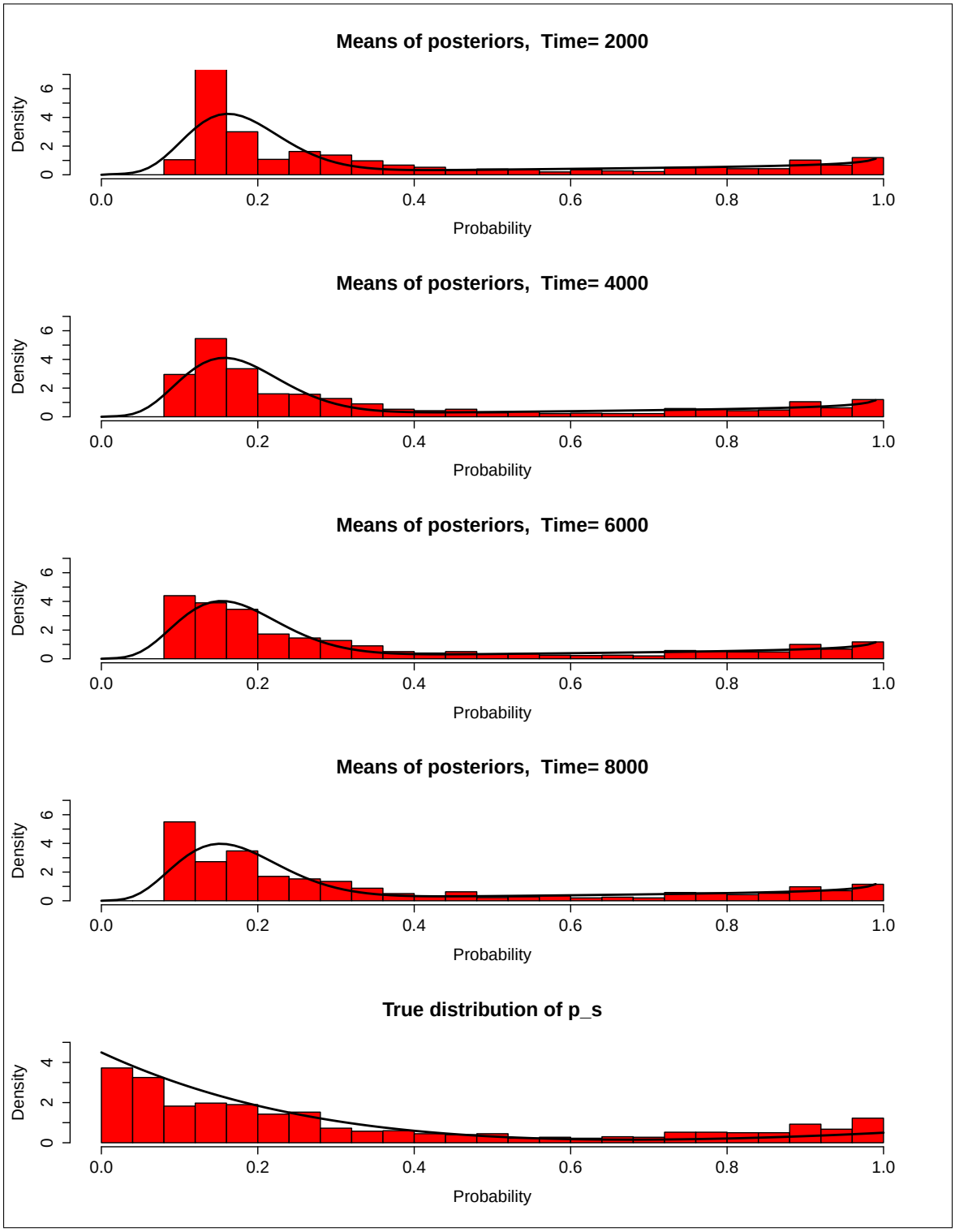

Figure 4.21: Fitted Mixture Distributions at Different Time Periods

These two numerical illustrations suggest that the EM algorithm is useful in scenarios wherein *nature* generates relevant and non-relevant items from sources that have a parameter p_s that itself is sampled from a mixture distribution.

CHAPTER 5:
Conclusion and Further Study

In this chapter, we summarize the conclusions drawn from the analysis and propose some suggestions and scenarios for future research.

5.1 Conclusion

In this thesis, we focus on the problem of efficiently processing the vast amount of data handled within the intelligence cycle. We propose a learning model that can be used efficiently to allocate the efforts available to the sources that generate data. We suggest a method that can be used to dynamically adapt the amount of effort that is allocated as data becomes available.

Our main conclusions and contributions can be summarized as follows:

- The model described can be used to allocate the resources/efforts for collecting/processing efficiently.
- The suggested algorithm employed in the model yields a sublinear performance in the simulations we conducted, meaning that the average regret tends to zero as the number of time periods increase.
- The model performs well when the p_s are from either a pure or a mixture population.
- The model can be adapted to situations in which there exists prior knowledge about the sources.
- We consider the number of sources chosen/capacity as possibly changing over time as information becomes available. With this approach, intelligence agencies can better control the regret in the exploration phase and avoid using excess capacity as the p_s values are better estimated.
- The model can also be employed to gain insights about the risk, which provides further guidance for the capacity required.
- We also use the EM algorithm to estimate the distributional parameters for the candidate subpopulations as the data is collected.

5.2 Further Study

Further studies can be conducted by relaxing the assumptions and settings we established for our model and methods.

First, the number of sources can be permitted to change, as some sources leave and new ones come. As an example of relaxing this assumption, one might consider translating a number of Twitter messages. Here, the Twitter accounts are the sources, and the messages are the items. Some of the sources may be inactive for some period of time; there may also be new accounts to look into or others that close.

Second, the p_s probabilities may be permitted to change in time. Third, we believe the most challenging aspect is to capture the dependencies between the sources and the item values over time. We intentionally did not specify what the source and the collection/processing asset are. Focusing on a particular asset would determine the setting and assumption.

List of References

[1] R. S. Lynd. (n.d.). Brainyquote.com. [Online]. Available: http://www.brainyquote.com/quotes/quotes/r/robertstau407186.html. Accessed Apr. 28, 2016.

[2] J. H. Hedley, "Analysis for strategic intelligence," in *Handbook of Intelligence Studies*, L. K. Johnson, Ed. New York, NY: Routledge, 2007, pp. 211–226.

[3] J. J. Wirtz, "The American approach to intelligence studies," in *Handbook of Intelligence Studies*, L. K. Johnson, Ed. New York, NY: Routledge, 2007, pp. 28–38.

[4] P. Gill and M. Phythian, *Intelligence in an Insecure World*. Malden, MA: Polity, 2006.

[5] CIA vision, mission, ethos & challenges? (n.d.). Central Intelligence Agency (CIA). [Online]. Available: https://www.cia.gov/about-cia/cia-vision-mission-values. Accessed Dec. 29, 2015.

[6] Intelligence defined. (n.d.). Federal Bureau of Investigation (FBI). [Online]. Available: https://www.fbi.gov/about-us/intelligence/defined. Accessed Dec. 24, 2015.

[7] *Department of Defense Dictionary of Military and Associated Terms*, JP 1-02, U.S. Joint Chiefs of Staff, Washington, DC, 2010.

[8] R. M. Clark, *Intelligence Analysis: a Target-centric Approach*, 4th ed. Washington, DC: CQ Press, 2013.

[9] *Joint Intelligence*, JP 2-0, U.S. Joint Chiefs of Staff, Washington, DC, 2013.

[10] M. V. Kauppi, "Counterterrorism analysis 101," *Defense Intelligence Journal*, vol. 11, no. 1, pp. 39–53, 2002.

[11] S. Shalev-Shwartz, "Online learning and online convex optimization," *Foundations and Trends in Machine Learning*, vol. 4, no. 2, pp. 107–194, 2011.

[12] H. Elad, "Draft: Introduction to online convex optimization," 2015, unpublished.

[13] W. H. Press, "Bandit solutions provide unified ethical models for randomized clinical trials and comparative effectiveness research," *Proceedings of the National Academy of Sciences*, vol. 106, no. 52, pp. 22,387–22,392, 2009.

[14] B. Awerbuch and R. Kleinberg, "Online linear optimization and adaptive routing," *Journal of Computer and System Sciences*, vol. 74, no. 1, pp. 97–114, 2008.

[15] E. Brochu et al., "Hedging strategies for bayesian optimization," *CoRR*, vol. abs/1009.5419, 2010. [Online]. Available: http://arxiv.org/abs/1009.5419

[16] J. C. Gittins, *Multi-armed Bandit Allocation Indices*. Chichester: Wiley, 1989.

[17] P. Auer et al., "The nonstochastic multiarmed bandit problem," *SIAM Journal on Computing*, vol. 32, no. 1, pp. 48–77, 2002.

[18] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285–294, 1933. [Online]. Available: http://www.jstor.org/stable/2332286

[19] O. Chapelle and L. Li, "An empirical evaluation of Thompson Sampling," in *Advances in Neural Information Processing Systems 24*, J. Shawe-Taylor et al., Eds. Curran Associates, Inc., 2011, pp. 2249–2257. [Online]. Available: http://papers.nips.cc/paper/4321-an-empirical-evaluation-of-thompson-sampling.pdf

[20] P. Viappiani, "Thompson sampling for Bayesian bandits with resets," in *Algorithmic Decision Theory*. Berlin Heidelberg: Springer, 2013, pp. 399–410.

[21] Y. Costica, "Optimizing classification in intelligence processing," M.S. thesis, Naval Postgraduate School, Monterey, CA, 2010.

[22] Y. Nevo, "Information selection in intelligence processing," M.S. thesis, Naval Postgraduate School, Monterey, CA, 2011.

[23] D. R. Ellis and M. Kress, "Algorithms for efficient intelligence collection," M.S. thesis, Naval Postgraduate School, Monterey, CA, 2013.

[24] S. Agrawal and N. Goyal, "Analysis of Thompson Sampling for the multi-armed bandit problem," *CoRR*, vol. abs/1111.1797, 2011. [Online]. Available: http://arxiv.org/abs/1111.1797

[25] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[26] R. A. Redner and H. F. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Review*, vol. 26, no. 2, pp. 195–239, 1984.

[27] A. P. Dempster et al., "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, series B (methodological)*, pp. 1–38, 1977.

[28] B. Grün et al., "Extended beta regression in R: Shaken, stirred, mixed, and partitioned," Working Papers in Economics and Statistics 2011-22, 2011. [Online]. Available: http://hdl.handle.net/10419/73505

[29] D. Karlis, "EM algorithm for mixed poisson and other discrete distributions," *Astin Bulletin*, vol. 35, no. 1, pp. 3–24, 2005.

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California