Discrete Stochastic Submodular Maximization: Adaptive vs. Non-Adaptive vs. Offline

Lisa Hellerstein Devorah Kletenik Patrick Lin

Department of Computer Science & Engineering
Polytechnic School of Engineering
New York University
{lisa.hellerstein,dkletenik,patrick.lin}@nyu.edu

Abstract

We consider the problem of stochastic monotone submodular function maximization, subject to constraints. We give results on adaptivity gaps, and on the gap between the optimal offline and online solutions. We present a procedure that transforms a decision tree (adaptive algorithm) into a non-adaptive chain. We prove that this chain achieves at least τ times the utility of the decision tree, over a product distribution and binary state space, where $\tau = \min_{i,j} \Pr[x_i = j]$. This proves an adaptivity gap of $\frac{1}{\tau}$ (which is 2 in the case of a uniform distribution) for the problem of stochastic monotone submodular maximization subject to state-independent constraints. For a cardinality constraint, we prove that a simple adaptive greedy algorithm achieves an approximation factor of $(1 - \frac{1}{e^{\tau}})$ with respect to the optimal offline solution; previously, it has been proven that the algorithm achieves an approximation factor of $(1 - \frac{1}{e})$ with respect to the optimal adaptive online solution. Finally, we show that there exists a non-adaptive solution for the stochastic max coverage problem that is within a factor $(1 - \frac{1}{e})$ of the optimal adaptive solution and within a factor of $\tau(1 - \frac{1}{e})$ of the optimal offline solution.

1 Introduction

We consider stochastic submodular function maximization, subject to constraints. This problem is motivated by problems in application areas such as machine learning, social networks, and recommendation systems.

In traditional (non-stochastic) submodular function maximization, the goal is to find a subset of "items" with maximum utility, as measured by a submodular utility function assigning a real value to each possible subset of items. In *stochastic* submodular function maximization, items have *states*. For example, if each item is a sensor, the item might be either working or broken. The utility of a subset of items depends not only on which items are in the subset, but also on their states. The state of each item is initially unknown, and can only be determined by performing a "test" on the item.

Algorithms for stochastic submodular maximization work in an on-line setting, sequentially choosing which item to test next. The choice can be adaptive, depending on the outcomes of previous tests. The state of each item is an independent random variable. The goal is to maximize the expected utility of the tested items. Previous work has sought to determine the adaptivity gap, which is the ratio between the optimal adaptive and non-adaptive solutions. In this paper we present new adaptivity gap results for discrete monotone submodular functions. We also consider another type of gap that has not been previously explored in the context of stochastic submodular maximization: the ratio between the optimal offline solution and the optimal adaptive solution.

Our main result is an adaptivity gap of 2 for all *state-independent* constraints, when the state set is binary and the item state distribution is uniform. More generally, for arbitrary product distributions, we prove an adaptivity gap of $\frac{1}{\tau}$. Here τ is the minimum value of $p_{i,j}$, where $p_{i,j}$ is the probability that item i is in state j. We say that a constraint is state-independent if the restriction on the items tested does not depend on their states. (A constraint requiring testing to stop when an item is found to be in state 1 is not state-independent.) A standard knapsack constraint is state-independent, and this is the first adaptivity gap for knapsack constraints. We prove the gap using a simple, bottom-up procedure that transforms a decision tree (adaptive algorithm) into a single non-adaptive chain corresponding to a root-leaf path in the tree.

Asadpour and Nazerzadeh previously showed an adaptivity gap of $\frac{e}{e-1}$ for a matroid constraint, using a stronger monotonicity condition than the one we use here (their results also apply to continuous states) [3]. For a cardinality constraint, we show that the simple adaptive greedy algorithm gives a $(1-\frac{1}{e^{\tau}})$-approximation with respect to the optimal *offline* solution, and that a dependence on τ in the approximation factor is necessary.

Finally, we consider the discrete stochastic version of the maximum coverage problem, which is a special case of submodular maximization subject to a cardinality constraint. We modify an approximation algorithm for the deterministic version of this problem, due to Ageev and Sviridenko [1, 2], to prove that the optimal non-adaptive solution for this problem is within a factor of $1-\frac{1}{e}$ of the optimal adaptive solution. We also show that the optimal non-adaptive solution for this problem is within a factor of $\tau(1-\frac{1}{e})$ with respect to the optimal offline solution.

2 Preliminaries and Definitions

Let $\mathbb{S} = \{0,\ldots,\ell-1\}$, and $\hat{\mathbb{S}} = \mathbb{S} \cup \{*\}$. A partial assignment is a vector $b \in \hat{\mathbb{S}}^n$. Hence b can be viewed as an assignment to variables $x_1,\ldots,x_n$. We will use these partial assignments to represent the outcomes of tests giving the states of n items, where each item can be in one of ℓ states. We write $b_i = s$ to indicate that item i has been tested and found to be in state s, and $b_i = *$ to indicate that the state of item i is unknown. We assume that the states of different items are independent.

If $b', b \in \hat{\mathbb{S}}^n$ and $b'_i = b_i$ for all $b_i \neq *$, then we call b' an *extension* of b, which we will write as $b' \succ b$. We will use $b_{j\leftarrow s}$ to denote the extension of b setting the j-th bit of b to s.

As is standard in the literature, given a set $N = \{1,\ldots,n\}$, we say a function $g : 2^N \to \mathbb{R}_{\geqslant 0}$ is a *utility function*. We will use the notation $g_S(j)$ to denote $g(S \cup \{j\}) - g(S)$.

We extend the notion of a utility function to the stochastic setting, wherein we have $g : \hat{\mathbb{S}}^n \to \mathbb{R}_{\geqslant 0}$ defined on partial assignments. In this case, we will write $g(S,b) := g(b')$ where b' is a partial assignment consistent with b on all entries i where $i \in S$, and $b_i = *$ whenever $i \notin S$. The notation $g_{S,b}(j)$ will denote $g(S \cup \{j\}, b) - g(S,b)$. If S' is a set, then $g_S(S')$ will mean $g(S \cup S') - g(S)$.

Utility function $g : \hat{\mathbb{S}}^n \to \mathbb{R}_{\geqslant 0}$ is called *submodular* if $g(b_{i\leftarrow s}) - g(b) \geqslant g(b'_{i\leftarrow s}) - g(b')$ when $b' \succ b$, $b'_i = b_i = *$, and $s \in \mathbb{S}$. We say g is *monotone* if $g(b_{i\leftarrow s}) \geqslant g(b)$ when $b_i = *$. That is, testing a bit can only increase the utility.

We will work with product distributions over the vectors $\mathbb{S}^n$: for $i \in N$, $j \in \mathbb{S}$, we use $p_{i,j}$ to mean the probability of the i-th coordinate being j (so that for each i, $\sum_j p_{i,j} = 1$). Many of our results are with respect to $\tau = \min_{i,j} p_{i,j}$. We use $\mathbb{E}[g_{S,b}(j)]$ to denote the expected increase in utility from testing the j-th bit.

We define the *Stochastic Submodular Maximization* problem as the problem of maximizing a monotone submodular function, in the stochastic setting with a discrete state space, subject to one or more constraints. More specifically, in this problem, we are given as input a monotone submodular $g : \hat{\mathbb{S}}^n \to \mathbb{R}_{\geqslant 0}$, the constraints, and the parameters of a product distribution over $\hat{\mathbb{S}}^n$.

Solving a Stochastic Submodular Maximization problem entails finding an *adaptive solution* that builds a set $Q \subseteq N$ item by item, testing each item after selecting it, that maximizes $\mathbb{E}[g(Q,b)]$ subject to the constraints. This is effectively a decision tree, whose nodes are labeled with $j \in N$, and we branch depending on the outcome of b_j. A *chain* is a balanced tree such that the labels are the same for all nodes at the same level. A chain is a *non-adaptive procedure*. We also consider the so-called *offline solution*, which knows *a priori* the outcomes of the random bits, and takes the optimal set with respect to said outcome.

A knapsack constraint has the form $\sum_{j \in Q} c_j \leqslant B$ where each $c_j \geqslant 0$, and $B \geqslant 0$. The c_j are called costs, and B is a budget. A special case is when $c_j = 1$ for all j and B is an integer; this is a cardinality constraint. (A cardinality constraint is also a special case of a matroid constraint.)

The *Stochastic Max Coverage* problem is a special case of the Stochastic Submodular Maximization problem with a cardinality constraint. The utility function $g : \hat{\mathbb{S}} \to \mathbb{R}_{\geqslant 0}$ in the Stochastic Max Coverage problem is defined as follows: let $E = \{e_1, \ldots, e_m\}$ be a ground set of elements, and for $i \in N$ and $r \in \mathbb{S}$, $S_{i,r} \subseteq E$. For convenience of notation, we will also write $S_{i,a}$ to mean S_{i,a_i} where $a \in \mathbb{S}^n$. Then $g(S,b) = |\bigcup_{i \in S} S_{i,b}|$. If b is a partial assignment, then we say e_j is *covered* with respect to b if $e_j \in \bigcup_{i: b_i \neq *} S_{i,b}$. This function g is clearly submodular and monotone.

The expected values of the optimal adaptive, non-adaptive, and offline solutions will be denoted by ADAPT, NONADAPT, and OFFLINE. We are interested in the *adaptivity gap* $\frac{\mathsf{ADAPT}}{\mathsf{NONADAPT}}$, as well as the ratios $\frac{\mathsf{OFFLINE}}{\mathsf{ADAPT}}$ and $\frac{\mathsf{OFFLINE}}{\mathsf{NONADAPT}}$.

3 Related Work

The submodular maximization problems studied in this paper were all initially studied in the deterministic setting. Feige showed that for all of these problems, under the assumption of P ≠ NP, no polynomial time algorithm can achieve an approximation factor better than $(1 - \frac{1}{e})$ [5]. For the problem of maximizing a monotone submodular function subject to a cardinality constraint, Nemhauser et al. showed that the natural greedy algorithm achieves an approximation factor of $(1 - \frac{1}{e})$ [8]. For the problem with a knapsack constraint, Sviridenko subsequently showed that an algorithm of Khuller et al. also achieves an approximation factor of $(1 - \frac{1}{e})$ [10, 7]. The results of Golovin and Krause achieve the same approximation factor for a cardinality constraint in the stochastic setting [6].

As mentioned earlier, Asadpour and Nazerzadeh showed an adaptivity gap of $\frac{e}{e-1}$ for Stochastic Submodular Maximization with a matroid constraint, using a stronger definition of monotonicity. In their definition of monotonicity, for any partial assignment b and $s \in \mathbb{S}$, they require that $g(b_{i \leftarrow s}) \geqslant g(b)$ if either $b_i = *$ or $s \geqslant b_i$, whereas we only require that $g(b_{i \leftarrow s}) \geqslant g(b)$ if $b_i = *$. Their proof is based on Poisson clocks and pipage rounding [3], and applies to continuous state spaces. Our adaptivity gap results do not apply to continuous state spaces, but our proofs are combinatorial.

Chan and Farias studied the related Stochastic Depletion problem and gave a $\frac{1}{2}$-approximation for the problem with respect to what they call the offline solution in their model [4]; in our model, their algorithm translates to a $\frac{1}{2}$-approximation with respect to ADAPT for Stochastic Submodular Maximization with a cardinality constraint.

	Knapsack Constraint	Cardinality Constraint	Max Coverage
Deterministic	$(1-\frac{1}{e})$OPT [10]	$(1-\frac{1}{e})$OPT [9]	$(1-\frac{1}{e})$OPT [7]
Stochastic: Adaptive	OPEN	$(1-\frac{1}{e})$ADAPT [6, 3] **$(\mathbf{1}-\frac{\mathbf{1}}{\mathbf{e}^{\tau}})$OFFLINE**	$(1-\frac{1}{e})$ADAPT [6, 3] **$(\mathbf{1}-\frac{\mathbf{1}}{\mathbf{e}^{\tau}})$OFFLINE**
Stochastic: Non-Adaptive	**τADAPT**	$(1-\frac{1}{e})$ADAPT† [3] **τADAPT**	$(1-\frac{1}{e})$ADAPT† [3] **$(\mathbf{1}-\frac{\mathbf{1}}{\mathbf{e}})$ADAPT** **$\tau(\mathbf{1}-\frac{\mathbf{1}}{\mathbf{e}})$OFFLINE**

Table 1: Bounds for Stochastic Submodular Maximization

Table 1 summarizes approximation bounds for stochastic discrete monotone submodular function maximization with a knapsack constraint, a cardinality constraint, and for max-coverage. Results from this paper are in bold. We denote with a † bounds relying on the stronger definition of monotonicity of [3].

The entries in the first row give the best bounds known for polynomial-time algorithms solving the deterministic versions of the problems, assuming oracle access to the utility function g. The bounds are given in terms of OPT, the optimal solution to the deterministic problem. As noted above, these are the best bounds possible, assuming $\mathrm{P} \neq \mathrm{NP}$ [5].

The entries in the second row refer to the best nontrivial bounds achieved by polynomial-time algorithms for the stochastic versions of the problems, assuming polynomial-time access to g. Both bounds are achieved by the Adaptive Greedy algorithm (described in Section 5). We note that Golovin and Krause give a randomized version of the algorithm for the knapsack constraint achieving an approximation factor of $(1-\frac{1}{e})$, but with the relaxation that the budget only needs to be met in expectation [6]. The last row refers to the bounds achieved by the respective best possible non-adaptive solutions for the problems in the stochastic setting (irrespective of running time).

4 An Adaptivity Gap for State-Independent Constraints

In this section we present an adaptivity gap for Stochastic Submodular Maximization with state-independent constraints. We use a technique that takes a decision tree and outputs a root-leaf path by collapsing the tree bottom up in a greedy manner; at each step, one child chain of a node replaces the other, leaving a single longer chain.

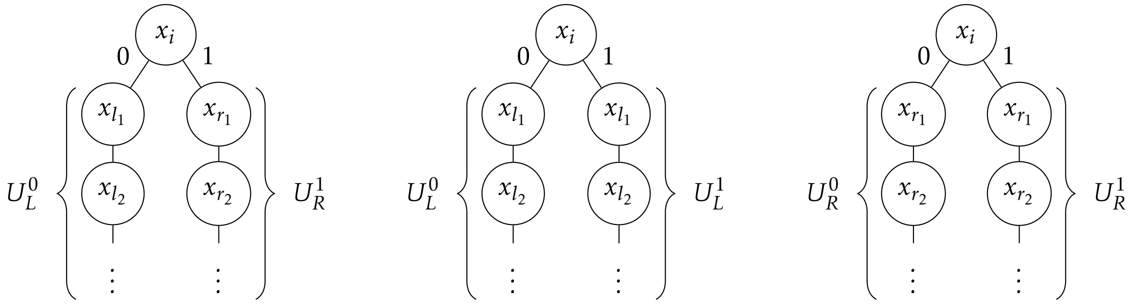
We show that under a product distribution over $\{0,1\}^n$, this gives a non-adaptive procedure that is a τ-approximation of expected utility of the original tree, where $\tau = \min_{i,j} p_{i,j}$. This gives a bound on the adaptivity gap for the problem: $\frac{\text{ADAPT}}{\text{NONADAPT}} \leqslant \frac{1}{\tau}$ for binary states.

Theorem 1. *For binary states, the Stochastic Submodular Maximization problem with state-independent constraints has an adaptivity gap of at most $\frac{1}{\tau}$.*

Proof. Let T be a decision tree corresponding to a solution to an instance of Stochastic Submodular Maximization with state-independent constraints, and binary states. We show that if T achieves expected utility U, then there exists a chain, corresponding to a root-leaf path in T, that achieves expected utility $\tau \cdot U$. Since the constraints are state-independent, this root-leaf path must obey the constraints, and the theorem follows.

We use a recursive procedure to turn T into a chain. At any intermediate step, we have a subtree consisting of a parent node whose child subtrees are chains. We show that when performing the procedure on this subtree, the loss incurred in expected utility is at most $1-\tau$ times the expected utility contributed by the parent node. Since the expected utility of a decision tree is a weighted sum of the expected utility contributed by its nodes, the total loss from the entire procedure is at most $1-\tau$ times the expected utility of the whole tree.

Hence without loss of generality, suppose T is a tree with a root node labeled x_i for some $i \in N$ whose two child subtrees are chains, as shown in Figure 1a. For convenience of notation, assume the nodes on the left child chain are labeled x_{l_1} through x_{l_c}, and the nodes on the right child chain are labeled x_{r_1} through x_{r_d}.



(a) T

(b) T_L

(c) T_R

Figure 1: T and its variants

Let $L = \{l_m\}_{m=1}^c$ and $R = \{r_m\}_{m=1}^d$. For $D \in \{L, R\}$ and fixed assignment b, monotonicity gives $g_{\varnothing,b}(\{i\} \cup D) \geqslant g_{\varnothing,b}(D)$, and submodularity gives $g_{\varnothing,b}(D) \geqslant g_{\{i\},b}(D)$. Thus, in expectation,

$$\mathbb{E}[g_{\varnothing,b}(\{i\} \cup L) \mid b_i = 1] \geqslant \mathbb{E}[g_{\varnothing,b}(L) \mid b_i = 1] = \mathbb{E}[g_{\varnothing,b}(L) \mid b_i = 0] \geqslant \mathbb{E}[g_{\{i\},b}(L) \mid b_i = 0] \tag{1}$$

and similarly

$$\mathbb{E}[g_{\varnothing,b}(\{i\} \cup R) \mid b_i = 0] \geqslant \mathbb{E}[g_{\{i\},b}(R) \mid b_i = 1]. \tag{2}$$

We introduce the following notation: Set $U_i^0 = \mathbb{E}[g_{\varnothing,b}(i) \mid b_i = 0]$, $U_i^1 = \mathbb{E}[g_{\varnothing,b}(i) \mid b_i = 1]$ (note that $g_{\varnothing,b}(i)$ is actually constant on all b with the same value for b_i), and $U_i = \mathbb{E}[g_{\varnothing,b}(i)] = p_{i,0}U_i^0 + p_{i,1}U_i^1$. So U_i is the expected utility contributed by the root node x_i.

Set $U_L^0 = \mathbb{E}[g_{\{i\},b}(L) \mid b_i = 0]$, that is, the expected increase in utility from testing $x_{l_1}, \ldots, x_{l_c}$ after testing x_i and getting $b_i = 0$. Similarly, we set $U_L^1 = \mathbb{E}[g_{\{i\},b}(L) \mid b_i = 1]$, $U_R^0 = \mathbb{E}[g_{\{i\},b}(R) \mid b_i = 0]$ and $U_R^1 = \mathbb{E}[g_{\{i\},b}(R) \mid b_i = 1]$.

With this notation in mind, we can rewrite, respectively, (1) and (2) as $U_i^1 \geqslant U_L^0 - U_L^1$ and $U_i^0 \geqslant U_R^1 - U_R^0$, that is,

$$U_i \geqslant p_{i,0}(U_R^1 - U_R^0) + p_{i,1}(U_L^0 - U_L^1). \tag{3}$$

Consider the two variants T_L and T_R of T as follows: T_L is the result of replacing the left child chain of T with the right child chain, i.e, both the left and right child chains are labeled x_{l_1} through x_{l_d}; T_R is instead the result of replacing the right child chain with the left. T_L and T_R are shown in Figures 1b and 1c, respectively. Note that they are both fully non-adaptive, and thus can just as easily be represented by pure chains.

Without loss of generality, we will assume

$$p_{i,0}(U_L^0 - U_R^0) \leqslant p_{i,1}(U_R^1 - U_L^1) \tag{4}$$

and substitute T_R for T by replacing the left child chain of T with its right child chain (in the symmetric setting, we have $p_{i,0}(U_L^0 - U_R^0) \geqslant p_{i,1}(U_R^1 - U_L^1)$ and substitute T_L for T). Set

$$S(b) = \begin{cases} \{i\} \cup L & \text{when } b_i = 0 \\ \{i\} \cup R & \text{when } b_i = 1 \end{cases}.$$

Let Δ be the expected loss in utility. We want to show that $\tau U_i \geqslant \Delta$. We compute Δ:

$$\begin{aligned}\Delta &= \mathbb{E}[g(S(b), b)] - \mathbb{E}[g(\{i\} \cup R, b)] \\ &= [p_{i,0}(U_i^0 + U_L^0) + p_{i,1}(U_i^1 + U_R^1)] - [p_{i,0}(U_i^0 + U_R^0) + p_{i,1}(U_i^1 + U_R^1)] \\ &= p_{i,0}(U_L^0 - U_R^0).\end{aligned}$$

There are two cases: (i) $p_{i,0} \leqslant p_{i,1}$ and (ii) $p_{i,0} > p_{i,1}$.

In case (i) we show that $\frac{1}{p_{i,1}}\Delta \leqslant U_i$, from which it follows that $\Delta \leqslant \tau U_i$ since $\tau \leqslant p_{i,1}$. Indeed, using $p_{i,1} = 1 - p_{i,0}$ we have

$$\begin{aligned}\frac{1}{p_{i,1}}\Delta &= \left(1 + \frac{p_{i,0}}{p_{i,1}}\right)\Delta \\ &\leqslant \Delta + p_{i,0}(U_R^1 - U_L^1) \\ &= p_{i,0}[(U_L^0 - U_R^0) + (U_R^1 - U_L^1)] \\ &= p_{i,0}[(U_L^0 - U_L^1) + (U_R^1 - U_R^0)] \\ &\leqslant p_{i,0}(U_L^0 - U_L^1) + p_{i,1}(U_R^1 - U_R^0) \\ &\leqslant U_i\end{aligned}$$

where the first inequality follows by (4), the second inequality since $p_{i,0} \leqslant p_{i,1}$, and the last inequality from (3).

In case (ii) we show instead that $\frac{1}{p_{i,0}}\Delta \leqslant U_i$, from which again we have $\Delta \leqslant \tau U_i$. The computation is similar:

$$\begin{aligned}\frac{1}{p_{i,0}}\Delta &= \left(1 + \frac{p_{i,1}}{p_{i,0}}\right)\Delta \\ &\leqslant \Delta + p_{i,1}(U_L^0 - U_R^0) \\ &\leqslant p_{i,1}[(U_R^1 - U_L^1) + (U_L^0 - U_R^0)] \\ &= p_{i,1}[(U_L^0 - U_L^1) + (U_R^1 - U_R^0)] \\ &< p_{i,0}(U_L^0 - U_L^1) + p_{i,1}(U_R^1 - U_R^0).\end{aligned}$$

This completes the proof. □

Our bottom-up greedy approach cannot achieve a constant-factor approximation; for every threshold $\theta \in [0, \frac{1}{2}]$, it is possible to construct a simple example where $\tau < \theta$ in which the procedure used in the proof loses more than a $1 - \theta$ fraction of the original utility. Some examples can be found in the appendix. Other methods may be able to achieve better bounds.

5 The Gap Between ADAPT and OFFLINE

In this section, we consider the gap $\frac{\mathsf{OFFLINE}}{\mathsf{ADAPT}}$ for Stochastic Submodular Maximization with a cardinality constraint.

We first define some notation. We use k to denote the number of items allowed by the cardinality constraint. For consistency, any variant of b will refer to a partial assignment, and any variant of a will refer to only a full assignment, that is, $a \in \mathbb{S}^n$. For the sake of clarity, in this section we will explicitly specify the assignments over which we are taking expectations, except with the shorthand that when b is a fixed partial assignment, $\mathbb{E}_{a \succ b}[\cdot]$ will mean $\mathbb{E}_a[\cdot \mid a \succ b]$. If we have a sequence $\{S^t\}_{t=0}^k$, we write $g_t(j)$ in place of $g_{S^t}(j)$.

We use AGREEDY to denote the expected utility of the *Adaptive Greedy* algorithm of Golovin and Krause [6] (also called *Adaptive Myopic* by Asadpour and Nazerzadeh [3]), which, starting with $Q^0 = \varnothing$, at each step t, based on the partial assignment b^{t-1} of bits tested so far, adaptively picks i_t satisfying

$$i_t = \operatorname*{arg\,max}_{i \in N \setminus Q^{t-1}} \mathop{\mathbb{E}}_{a \succ b^{t-1}} [g_{t-1,a}(i)],$$

sets $Q^t = Q^{t-1} \cup \{i_t\}$, and tests b_{i_t} to get b^t. This implicitly forms a decision tree of depth k that branches based on the outcome of b_{i_t}, and outputs Q^k.

It is clear that AGREEDY $\leqslant$ ADAPT $\leqslant$ OFFLINE. We show:

Theorem 2. *AGREEDY $\geqslant (1 - \frac{1}{e^\tau}) \cdot$ OFFLINE for Stochastic Submodular Maximization with a cardinality constraint.*

In other words, $\frac{\mathsf{OFFLINE}}{\mathsf{ADAPT}} \leqslant \frac{e^\tau}{e^\tau - 1}$. Although this is bound is weak for small τ, we observe that some dependence on τ is unfortunately unavoidable:

Proposition 3. *ADAPT cannot achieve an approximation bound better than τ relative to OFFLINE.*

The proof of Proposition 3 is by example, and is given in the appendix. We will note, however, that τ and $1 - \frac{1}{e^\tau}$ are close: for $0 < \tau \leqslant \frac{1}{2}$ the difference between τ and $1 - \frac{1}{e^\tau}$ is at most ~ 0.107, which is achieved at $\tau = \frac{1}{2}$.

We now prove Theorem 2. We use the following lemma due to Wolsey:

Lemma 4 ([11])**.** *Let k be a positive integer, and $s > 0$, $\rho_1, \ldots, \rho_k \geqslant 0$ be reals. Then*

$$\frac{\sum_{i=1}^k \rho_i}{\min_{t \in \{1,\ldots,k\}} \left(s\rho_t + \sum_{i=1}^{t-1} \rho_i \right)} \geqslant 1 - \left(1 - \frac{1}{s}\right)^k \geqslant 1 - \frac{1}{e^{k/s}}.$$

Next, let Q_a^* be the optimal offline solution on assignment a, $\tau = \min_{i,j} p_{i,j}$, and (Q^t, b^t) be the collection Q^t given by the Adaptive Greedy algorithm at step t corresponding to the partial assignment b^t. Then:

Lemma 5. *For $t = 1, 2, \ldots, k$,*

$$\mathop{\mathbb{E}}_a [g(Q_a^*, a)] \leqslant \mathop{\mathbb{E}}_a \left[g(Q^{t-1}, a) \right] + \frac{k}{\tau} \cdot \mathop{\mathbb{E}}_a \left[g(Q^t, a) - g(Q^{t-1}, a) \right]$$

Proof. Suppose Greedy chooses i_t at step t, that is, $Q^t \setminus Q^{t-1} = \{i_t\}$. By definition we have

$$\mathop{\mathbb{E}}_{a' \succ b^{t-1}}\left[g_{t-1,a'}(i_t)\right] = \sum_{m=1}^{\ell} p_{i_t,m}\left[g_{t-1,b^{t-1}_{i_t \leftarrow m}}(i_t)\right].$$

Next, suppose $j \in Q^*_{a'} \setminus Q^{t-1}$ where a' is a full assignment and $a' \succ b^{t-1}$. We can write $\Pr[a'] = \prod_i \psi_i$ where $\psi_i \in \{p_{i,0}, \dots, p_{i,\ell-1}\}$. If we then let $a'_{j \leftarrow m}$ be the assignment a' with the j-th bit set to m, we have

$$g_{t-1,a'}(j) \leqslant \frac{1}{\psi_j}\left(\sum_{m=0}^{\ell-1} p_{j,m}\left[g_{t-1,a'_{j\leftarrow m}}(j)\right]\right) \leqslant \frac{1}{\tau}\left(\sum_{m=0}^{\ell-1} p_{i_t,m}\left[g_{t-1,b^{t-1}_{i_t\leftarrow m}}(i_t)\right]\right) = \frac{1}{\tau} \cdot \mathop{\mathbb{E}}_{a' \succ b^{t-1}}\left[g_{t-1,a'}(i_t)\right]$$

where the first inequality follows from the fact that $\psi_j = p_{j,m}$ for some m, and the second from the definitions of τ and i_t.

Then since $|Q^*_{a'} \setminus Q^{t-1}| \leqslant k$ we get

$$\sum_{j \in Q^*_{a'} \setminus Q^{t-1}} g_{t-1,a'}(j) \leqslant \frac{k}{\tau} \cdot \mathop{\mathbb{E}}_{a' \succ b^{t-1}}\left[g_{t,a'}(i_t)\right]. \tag{5}$$

Next, we have

$$\begin{aligned}
\mathop{\mathbb{E}}_{a}\left[g(Q^*_a, a)\right] &= \mathop{\mathbb{E}}_{b^{t-1}}\left[\mathop{\mathbb{E}}_{a' \succ b^{t-1}}\left[g(Q^*_{a'}, a')\right]\right] \\
&\leqslant \mathop{\mathbb{E}}_{b^{t-1}}\left[\mathop{\mathbb{E}}_{a' \succ b^{t-1}}\left[g(Q^*_{a'} \cup Q^{t-1}, a')\right]\right] \\
&\leqslant \mathop{\mathbb{E}}_{b^{t-1}}\left[\mathop{\mathbb{E}}_{a' \succ b^{t-1}}\left[g(Q^{t-1}, a') + \sum_{j \in Q^*_{a'} \setminus Q^{t-1}} g_{t-1,a'}(j)\right]\right] \\
&\leqslant \mathop{\mathbb{E}}_{a}\left[g(Q^{t-1}, a)\right] + \mathop{\mathbb{E}}_{b^{t-1}}\left[\frac{k}{\tau} \cdot \mathop{\mathbb{E}}_{a' \succ b^{t-1}}\left[g_{t-1,a'}(i_t)\right]\right] \\
&= \mathop{\mathbb{E}}_{a}\left[g(Q^{t-1}, a)\right] + \frac{k}{\tau} \cdot \mathop{\mathbb{E}}_{a}\left[g_{t-1,a}(i_t)\right]
\end{aligned}$$

where the first inequality follows from monotonicity, the second from submodularity, and the third from (5). □

In light of Lemma 4 we see that Theorem 2 follows from Lemma 5:

of Theorem 2. Let $\rho_i = \mathbb{E}_a[g(Q^i, a) - g(Q^{i-1}, a)]$, then clearly we have $\mathbb{E}_a[g(Q^{t-1}, a)] = \sum_{i=1}^{t-1} \rho_i$. Since the Adaptive Greedy algorithm outputs Q^k,

$$\frac{\mathsf{AGREEDY}}{\mathsf{OFFLINE}} = \frac{\sum_{i=1}^{k} \rho_i}{\mathbb{E}_a[g(Q^*_a, a)]} \geqslant \frac{\sum_{i=1}^{k} \rho_i}{\min_{t \in \{1,\dots,k\}}\left(\frac{k}{\tau}\rho_t + \sum_{i=1}^{t-1} \rho_i\right)} \geqslant 1 - \frac{1}{e^{\tau}}$$

as desired. □

6 Gaps for Stochastic Max Coverage

In this section we consider the special case of Stochastic Max Coverage. We still achieve an adaptivity gap of $\frac{e}{e-1}$, which is tight by an example given by Asadpour and Nazerzadeh [3, §3.1]. Furthermore, we obtain a bound of $\frac{e}{\tau(e-1)}$ for $\frac{\mathsf{OFFLINE}}{\mathsf{NONADAPT}}$.

We use of the following fact, which is easily seen by inspection or by calculus.

Lemma 6. *For all* $x \geqslant 0$, $1-(1-\frac{x}{k})^k \geqslant x(1-\frac{1}{e})$.

We give a combinatorial proof of the following result, which is adapted from work of Ageev and Sviridenko on the deterministic version of the problem [1, 2].

Theorem 7. *For the Stochastic Max Coverage problem, there exists a non-adaptive solution that achieves coverage* $(1-\frac{1}{e})$*ADAPT.*

Proof. For simplicity, we give the analysis for the uniform distribution over binary states; the analysis is similar for product distributions over $\ell > 2$ states.

We make use of the *neighbor property*, which every adaptive algorithm satisfies by definition: given two assignments a, a' differing only in bit j, either b_j is tested for both assignments, or for neither. We denote the expected value of the optimal offline solution satisfying the neighbor property by NBR. Clearly, NONADAPT $\leqslant$ ADAPT $\leqslant$ NBR $\leqslant$ OFFLINE. We will prove that NONADAPT $\geqslant (1-\frac{1}{e})$NBR, thus achieving the desired adaptivity gap. Let $\mathcal{X}$ denote the procedure giving the optimal offline solution satisfying the neighbor property.

For each $i \in N$ and assignment a we assign a variable $x_{i,a}$ such that $x_{i,a} = 1$ if and only if $S_{i,a}$ is included in Q_a, the subcollection given by $\mathcal{X}$. Correspondingly we assign to each $e_j \in E$ a variable $y_{j,a}$ such that $y_{j,a} = 1$ if and only if $j \in \bigcup_{i\in Q_a} S_{i,a}$. Since NBR denotes the expected number of ground elements by the solution given by $\mathcal{X}$, NBR $= \sum_j \mathbb{E}_a[y_{j,a}]$.

Consider the following (randomized) algorithm for producing a non-adaptive solution from the solution given by $\mathcal{X}$: randomly pick k sets according to the following probability distribution: pick i with probability $\frac{1}{k}\sum_a \Pr[a]\sum_{i=1}^n x_{i,a} = \frac{1}{k2^n}\sum_a\sum_{i=1}^n x_{i,a}$.

For each e_j we say i is a *promising cover* for j if either $e_j \in S_{i,0}$ or $e_j \in S_{i,1}$. We divide the promising covers into two categories: i is of type B if $e_j \in S_{i,0} \cap S_{i,1}$ and i is of type A otherwise. We abuse notation slightly and let $A = \frac{1}{2^n}\sum_{i \text{ of type } A}\sum_{a:e_j\in S_{i,a}} x_{i,a}$, and similarly $B = \frac{1}{2^n}\sum_{i \text{ of type } B}\sum_a x_{i,a}$. By definition $A+B \geqslant \mathbb{E}_a[y_{j,a}]$.

In expectation, the probability that a random i produces a promising cover of type A for e_j is $\frac{2A}{k}$; hence the probability that, in expectation, at least one of the chosen i is a promising cover of type A for e_j is at least $1-(1-\frac{2A}{k})^k \geqslant 2A(1-\frac{1}{e})$. A promising cover of type A covers e_j with probability $\frac{1}{2}$, so e_j is covered by at least one promising cover of type A with probability at least $A(1-\frac{1}{e})$.

Similarly, picking a random set produces a promising cover of type B for j with probability $\frac{B}{k}$, so at least one of the chosen i is a promising cover of type B for e_j is at least $1-(1-\frac{B}{k})^k \geqslant B(1-\frac{1}{e})$, and a promising cover of type B covers e_j with probability 1, so e_j is covered by at least one promising cover of type B with probability at least $B(1-\frac{1}{e})$. Since the promising covers of type A and B are disjoint, we conclude that e_j is covered with probability at least $(1-\frac{1}{e})(A+B) \geqslant (1-\frac{1}{e})\mathbb{E}_a[y_{j,a}]$.

Before finishing the proof, we note that for product distributions, we will actually have many more such categories: for each $i \in N$ we will have a different set of categories. Furthermore, when there are $\ell > 2$ states, there is a larger number of possible ways of covering each e_j and hence a larger number of categories. However, the analysis will be similar, so that it will still the case that e_j is covered with probability $\geqslant (1-\frac{1}{e})\mathbb{E}_a[y_{j,a}]$.

We return to the proof at hand. By linearity of expectation, the expected number of elements covered by this non-adaptive solution is equal to the sum of the probabilities that each j is covered, in other words,

$$\mathbb{E}[\#\text{ elements covered}] = \sum_j \Pr[j \text{ is covered}] \geqslant \left(1-\frac{1}{e}\right)\sum_j \mathbb{E}_a[y_{j,a}] = \left(1-\frac{1}{e}\right)\mathsf{NBR}.$$

Since this randomized procedure achieves at least $(1-\frac{1}{e})\mathsf{NBR}$ in expectation, then there must exist a non-adaptive solution that achieves at least $(1-\frac{1}{e})\mathsf{NBR} \geqslant (1-\frac{1}{e})\mathsf{ADAPT}$, thus completing the proof. □

A similar analysis gives the following result:

Theorem 8. *For the Stochastic Max Coverage problem, there exists a non-adaptive solution that achieves coverage $\tau(1-\frac{1}{e})$OFFLINE.*

Proof. The proof is similar to the proof of Theorem 7. Again we use variables $x_{i,a}$ such that $x_{i,a} = 1$ if and only if $S_{i,a}$ is included in the optimal subcollection Q_a for a, and $y_{j,a}$ such that $y_{j,a} = 1$ if and only if $e_j \in \bigcup_{i\in Q_a} S_{i,a}$. Clearly $\sum_{i:e_j\in S_{i,a}} x_{i,a} \geqslant y_{j,a}$ for each assignment a and $\mathsf{OFFLINE} = \sum_j \mathbb{E}_a[y_{j,a}]$.

Again we randomly pick k sets according to the following probability distribution: pick i with probability $\frac{1}{k}\sum_a \Pr[a]\sum_{i=1}^n x_{i,a}$.

In expectation, the probability that picking a random i produces a promising cover for e_j is

$$\frac{1}{k}\sum_a \Pr[a]\sum_{i:e_j\in S_{i,a}} x_{i,a} \geqslant \frac{1}{k}\sum_a \Pr[a]y_{j,a} = \frac{\mathbb{E}_a[y_{j,a}]}{k}$$

hence the probability that at least one promising cover for j is chosen is

$$1-\left(1-\frac{\mathbb{E}_a[y_{j,a}]}{k}\right)^k \geqslant (1-\frac{1}{e})\mathbb{E}_a[y_{j,a}].$$

Since each promising cover covers j with probability $\geqslant \tau$, the probability that j is actually covered is at least $\tau(1-\frac{1}{e})\mathbb{E}_a[y_{j,a}]$. The rest of the proof follows from analysis similar to the analysis used in the proof of Theorem 7. □

References

[1] Alexander A. Ageev and Maxim Sviridenko. Approximation algorithms for maximum coverage and max cut with given sizes of parts. In Gérard Cornuéjols, Rainer E. Burkard, and Gerhard J. Woeginger, editors, *IPCO 1999*, volume 1610 of *LNCS*, pages 17–30. Springer, 1999.

[2] Alexander A. Ageev and Maxim Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *J. Comb. Optim.*, 8(3):307–328, 2004.

[3] Arash Asadpour and Hamid Nazerzadeh. Maximizing stochastic monotone submodular functions. *arXiv preprint arXiv:0908.2788v2*, 2014.

[4] Carri W. Chan and Vivek F. Farias. Stochastic depletion problems: Effective myopic policies for a class of dynamic optimization problems. *Math. Oper. Res.*, 34(2):333–350, 2009.

[5] Uriel Feige. A threshold of ln n for approximating set cover. *J. ACM*, 45(4):634–652, 1998.

[6] Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *J. Artif. Intell. Res. (JAIR)*, 42:427–486, 2011.

[7] Samir Khuller, Anna Moss, and Joseph Naor. The budgeted maximum coverage problem. *Inf. Process. Lett.*, 70(1):39–45, 1999.

[8] George L. Nemhauser and Laurence A. Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Math. Oper. Res.*, 3(3), 1978.

[9] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions – I. *Math. Program.*, 14(1):265–294, 1978.

[10] Maxim Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Oper. Res. Lett.*, 32(1):41–43, 2004.

[11] Laurence A Wolsey. Maximising real-valued submodular functions: Primal and dual heuristics for location problems. *Math. Oper. Res.*, 7(3):410–425, 1982.

A Appendix: Counterexamples

A.1 Counterexamples for Section 4

Without loss of generality we will assume $p_{i,1}(U_R^1 - U_L^1) \geqslant p_{i,0}(U_L^0 - U_R^0)$. This is the situation in which we substitute T_R for T. The expected loss in utility is $\Delta = \mathbb{E}[g(S(b), b)] - \mathbb{E}[g(\{i\} \cup R, b)] = p_{i,0}(U_L^0 - U_R^0)$.

We give a systematic way of finding counterexamples roughly showing that for $\theta \in [0, \frac{1}{2}]$, by picking $\tau < \theta$ we can achieve $\Delta \geqslant \theta \cdot U_i$ where $U_i = p_{i,0} U_i^0 + p_{i,1} U_i^1$.

Example 9. Suppose $\theta = 1 - 10^{-m}$, eg. if $m = 3$ then $\theta = 0.999$. Then let $p_{i,0} = 10^{-m-1}$ (so $p_{i,1} = 1 - 10^{-m-1}$), $U_R^1 = 10^{m+1}$, $U_R^0 = 5$, $U_L^1 = 10^{m+1} - 1$, $U_L^0 = 10^{m+1} - 2$, $U_i^0 = 10^{m+1} - 4$, and $U_i^1 = 3 \cdot 10^{-m-1}$. Then

$$\Delta = 10^{-m-1}(10^{m+1} - 2 - 5) = 1 - 7 \cdot 10^{-m-1}$$

and

$$\begin{aligned}\theta \cdot U_i &= (1 - 10^{-m})[3 \cdot 10^{-m-1}(1 - 10^{-m-1}) + (10^{m+1} - 4) \cdot 10^{-m-1}] \\ &\approx 1 - 2 \cdot 10^{-m}\end{aligned}$$

as desired. It is easy to verify that $p_{i,1}(U_R^1 - U_L^1) \geqslant p_{i,0}(U_L^0 - U_R^0)$. Further, the utility values given here can be shown to be consistent with a monotone, submodular utility function g. □

A.2 Proof of Proposition 3

We construct a counterexample for which ADAPT $\approx (\tau - \varepsilon)$OFFLINE.

Example 10. Consider an instance of the stochastic submodular coverage with a cardinality constraint problem as follows: say $\ell = 2$ (so this is over binary states), let $t = \frac{1}{\tau}$ and set $n = 1 + t^2$. Let $p_{i,1} = \tau$ and $p_{i,0} = 1 - \tau$ for all $i \in N$. Let $B = 1$, so the problem is to maximize the expected utility of picking a single bit.

Consider the following stochastic monotone submodular function g defined over $\mathbb{S}$: for all subcollections $Q \subseteq N$ such that $i \notin Q$, let

$$g_{Q,a}(i) = \begin{cases} 1 & \text{if } i = 1 \\ t - \varepsilon & \text{if } i \in \{2, \ldots, n\} \text{ and } a_i = 1 \\ 0 & \text{if } i \in \{2, \ldots, n\} \text{ and } a_i = 0 \end{cases}$$

where the monotonicity and submodularity of g follow from the fact that g is an additive utility function.

For $i \notin Q$, $\mathbb{E}[g_{Q,b}(i)] = 1$ if $i = 1$ and $\mathbb{E}[g_{Q,b}(i)] = \frac{t-\varepsilon}{t}$ if $i \in \{2, \ldots, n\}$. Due to the low probability of any variable from the second group having a value of 1, no adaptive tree outperforms the greedy choice of picking the first variable. Thus $\mathsf{ADAPT} = 1$.

The optimal offline procedure, on the other hand, will pick any variable of the second group that has a value of 1. With probability $1 - (1 - \frac{1}{t})^{t^2}$, at least one variable of the second group will evaluate to 1; hence, $\mathsf{OFFLINE} = \left(1 - (1 - \frac{1}{t})^{t^2}\right) \cdot (t - \varepsilon) + (1 - \frac{1}{t})^{t^2} \cdot 1$.

It follows that for sufficiently large t, $\mathsf{ADAPT} \approx (\tau - \varepsilon)\mathsf{OFFLINE}$. □