Bundle Selling by Online Estimation of Valuation Functions

Daniel Vainsencher DANIELV@TX.TECHNION.AC.IL
Department of Electrical Engineering, Technion, Haifa, Israel

Ofer Dekel OFERD@MICROSOFT.COM
Microsoft Research, Redmond, Washington, USA

Shie Mannor SHIE@EE.TECHNION.AC.IL
Department of Electrical Engineering, Technion, Haifa, Israel

Abstract

We consider the problem of online selection of a bundle of items when the cost of each item changes arbitrarily from round to round and the valuation function is initially unknown and revealed only through the noisy values of selected bundles (the bandit feedback setting). We are interested in learning schemes that have a small regret compared to an agent who knows the true valuation function. Since there are exponentially many bundles, further assumptions on the valuation functions are needed. We make the assumption that the valuation function is supermodular and has non-linear interactions that are of low *degree* in a certain sense. We develop efficient learning algorithms that balance exploration and exploitation to achieve low regret in this setting.

1. Introduction

A player takes part in the following repeated game: on each round, the player purchases a set of items, bundles them together, and offers the bundle for sale. The price of each item is announced at the beginning of the round and the player pays the sum of the prices of the items he purchased. The bundle sells for its intrinsic market value, which is assumed to be a fixed but unknown monotone supermodular function, plus stochastic noise. In other words, the synergy between the items in a bundle may increase the bundle's value beyond the cost of the items, and the player can profit.

Appearing in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

The player may not know the value of a bundle when he offers it for sale. For example, he may offer the bundle for sale by an auction. Although the valuation function is not known to the player in advance, his goal is to asymptotically profit as if the valuation function was known in advance. To achieve this goal, the player must incrementally construct an estimate of the valuation function while constructing profitable bundles. The player constructs this estimate based on his experience from past sales. This form of learning with partial feedback is commonly known as learning with bandit feedback.

Supermodular valuation functions capture the idea that the whole is worth more than the sum of its parts. They are commonly used in economics and game theory to model the phenomenon of synergy; see Topkis (1998) for a detailed discussion. The repeated game described above is a simplified version of a typical scenario that occurs in the real world. For example, a real estate developer will purchase small adjacent lots from individual homeowners, and sell the combined land to a contractor who wants to build an apartment complex. A virtual wireless operator will purchase bands of spectrum at different locations; once enough bands are acquired, the operator can sell these bands with the guarantee of uninterrupted wireless service. A patent firm will purchase individual patents from small failing companies, and then sell the entire portfolio of patents to one of the industry giants. A stamp collector will purchase individual stamps and then sell a complete stamp series with a significant price markup. In all of these examples, the price of the individual items is known at the time of purchase but the intrinsic value of the bundle must be estimated from past experience. In all of these examples, the player acts as the middleman, and hopes to make a handsome profit from the transaction.

If the player has access to n different items, he can construct 2^n different bundles. This exponential decision space leads to both computational and statistical problems, which we detail in the next section. The supermodularity assumption goes a long way to solve the computational aspect, but does not make the valuation function easy to learn in the statistical sense. For example, Balcan & Harvey (2010) prove that the set of supermodular (equivalently, submodular) functions is not learnable, even in an easier noise free setting. To facilitate the statistical learning difficulties, we require an additional assumption. Intuitively, we assume that the synergies that increase the value of a bundle only occur within small sets of items. That is, we assume that the valuation function can be written as a sum of functions that each depends on a small number of elements. In some sense, our assumption is analogous to assuming that a multivariate polynomial has a small degree. We formalize this intuitive notion in Section 2 and use our formalization to derive an algorithm in Section 3.

As noted above, we assume that the price of each item is arbitrarily determined by the environment on each round. In fact, the environment can choose a sequence of cost functions that does not allow the same bundle to be optimal twice in less than exponential time. Therefore, finding a strategy that makes profits consistently seems to require learning the entire valuation function. This sets our problem apart from the simpler problem of finding a minimum or a maximum of a submodular function in an online setting (Streeter & Golovin, 2009; Hazan & Kale, 2009).

The price of each item is announced at the beginning of each round, and can therefore be thought of as a special case of linear "context" (Auer, 2003; Li et al., 2010) in the framework of contextual bandits (Wang et al., 2005; Langford & Zhang, 2008). Efficient algorithms for bandit problems with exponentially many arms exist in the context-free setting (Awerbuch & Kleinberg, 2004; Cesa-Bianchi & Lugosi, 2006) and for some special cases of contextual bandits (Dani et al., 2008). To the best of our knowledge, none of these algorithms efficiently solves our problem.

The problem of learning valuations or preferences of an agent has been studied in different contexts. A few difficulties common to them include high dimensions (Chajewska & Koller, 1999), inconsistent observations (Nielsen & Jensen, 2004) and partial observations (for example, the difficulties due to observing a dynamic process; see Chajewska et al., 2001). The approach we present in this paper mitigates these difficulties, and we believe that our techniques may be relevant in other contexts as well.

This paper is organized as follows. We describe the problem and assumptions more formally in Section 2 and present our solution to the problem in Section 3. We demonstrate some characteristics of our algorithm with a set of preliminary experiments in Section 4, followed by conclusions and future work. Some proofs are omitted for lack of space, and may be found in the full version of this paper (Vainsencher et al., 2011).

2. Setup

Let E be a set of n items. At the beginning of round t, the environment announces a cost function c_t that assigns a non-negative cost to each item $e \in E$. The cost function c_t may be any function that satisfies $\sum_{s\in E} c_t(s) \leq 1$. The environment also draws a valuation function f_t from a fixed but unknown distribution μ. We assume that the distribution μ is such that $\mathbb{E}f_t$ is a supermodular function into $[0,1]$. f_t assigns a value to each bundle $S \subset E$, but is not revealed to the player. The player then chooses a bundle $S_t \subset E$ and receives a profit equal to the valuation of S_t, minus the sum of costs of the items in S_t:

$$r_t(S_t) \;=\; f_t(S_t) - \sum_{e\in S_t} c_t(e) \quad .$$

We recall the definition of a supermodular set-function f. A function f is supermodular if for every two sets X, Y we have that:

$$f(X \cup Y) + f(X \cap Y) \;\geq\; f(X) + f(Y) \quad .$$

We say that f is submodular iff $-f$ is supermodular. Intuitively, a supermodular function has increasing returns and a submodular function has diminishing returns.

Our goal is to find a polynomial time computable strategy for the player that achieves essentially the same profit as would be possible if μ were known in advance. More formally, let S_t^* denote the bundle with the highest expected profit on round t: $S_t^* = \arg\max_{S\subseteq E} \left\{\mathbb{E}f_t(S) - \sum_{e\in S} c_t(e)\right\}$, then our goal is to minimize the expected regret:

$$R_T \;=\; \mathbb{E}\sum_{t=1}^{T} r_t(S_t^*) - r_t(S_t) \quad .$$

2.1. Modeling limited interaction

If we do not restrict the valuation function and allow it to be completely arbitrary, we would have an in-

tractable learning problem. We provide a representation of the valuation function that allows us to express modeling assumptions about the complexity of interactions among sets of items. The goal is to reduce the number of variables that need to be learned from exponential in n to exponential in some constant k that represents the complexity of the reduced function class. Below we offer an alternative representation of valuation functions that facilitates such a reduction.

Valuation functions are *real-valued set-functions* of the form $f : 2^E \mapsto \mathbb{R}$ that satisfy $f(\emptyset) = 0$. Define the *interaction function* of a set-function f as the function $g : 2^E \setminus \{\emptyset\} \mapsto \mathbb{R}$ that satisfies

$$\forall b \in 2^E \quad f(b) \;=\; \sum_{s \in 2^E : s \cap b \neq \emptyset} g(s) \quad .$$

A graph-based interpretation of g takes E as the nodes of a hypergraph. The function g assigns weights to all possible hyperedges in the graph. Then $f(S)$ corresponds to the sum of the weights of edges covered by S.

Proposition 1. *Every valuation function f has a unique interaction function g such that $g(\emptyset) = 0$.*

See Appendix A for the proof.

We denote the indicator of a set intersecting with S in the $2^n - 1$ dimensional real space by

$$\phi(S) = \mathbf{1}_{T \subset E \,:\, T \cap S \neq \emptyset}.$$

We can now define our profit from a bundle S on round t as

$$r_t(S) = \langle \phi(S), g_t - c_t \rangle \,, \tag{1}$$

where we lift c_t to be a set-function that attains a value of zero on non-singletons. While the above notation is convenient, it requires exponential dimensionality without additional assumptions.

We now define a nested family of set-function classes of increasing complexity. Intuitively, a set-function of *degree* k is capable of expressing nonlinear interactions within subsets of size at most k. Formally, a set-function f has *degree* k if its interaction function g satisfies

$$\forall b \in 2^E \text{ with } |b| > k \qquad g(b) = 0 \quad .$$

In words, f has degree k if g equals zero on all sets whose size is larger than k. We define F_k to be the set of set-functions of degree k. Note that F_1 is the set of linear (modular) set-functions and F_n includes all set-functions that equal zero on $\emptyset$.

A low degree set-function has less expressive power than a high degree set-function. The natural analogy is a multivariate polynomial of degree k: a linear combination of monomials, each of which captures the nonlinear interaction within a small set of k or less variables. The restriction to F_k reduces the statistical and space complexity of the problem from 2^n independent parameters that need to be learned to only $O\left(n^k\right)$.

An application domain where a restriction to F_k seems natural is that of selecting sensors to cover an area. In such a problem, each sensor comes with a cost and the player's objective is determined by the amount of area covered. If at most k sensors cover each point, it is not hard to show that the valuation function is in F_k.

We note that the structure of Eq. (1) makes it amenable to linear contextual bandits algorithms such as SupLinRel and LinRel (Auer, 2003; Dani et al., 2008) with regret $O\left(T^{1/2}\right)$ by a simple reduction; unfortunately the reduction is not computationally efficient.

2.2. Efficient selection via supermodularity

An agent that knows $\mathbb{E}f$ needs to find an optimal bundle under the current costs. Even when $f \in F_2$ this search is NP hard without additional assumptions (see the full version for a simple reduction from minimum vertex cover pointed out by Gadi Aleksandrowicz).

We therefore assume that $\mathbb{E}f$ is supermodular, and we note that the problem of maximizing a supermodular function (or equivalently, minimizing a submodular function) has been extensively studied, and several efficient algorithms exist; we refer the reader to McCormick (2006) for a detailed survey. The next section considers the gap between $\mathbb{E}f$ needed for optimal play and the information available from playing the bundling game.

3. A general algorithm and conditions for efficiency

In this section we describe a generic bundle selection algorithm for online learning of supermodular functions; see Algorithm 1. This algorithm calls three "black boxes", denoted by $A.1$, $A.2$, and $A.3$. Before the game begins, the algorithm selects a set of bundles called the exploration set (see $A.1$). After the game begins, the algorithm devotes some rounds to exploration and others to exploitation. On exploration rounds, the algorithm obtains independent unbiased estimates of the interaction function g that corresponds to $\mathbb{E}f$ (see $A.2$). It then approximates the mean of these esti-

Algorithm 1 Bundle selection algorithm A

Input: # of rounds T, interaction size bound k
$A.1$: Choose the exploration set $B \subset \mathcal{P}(E)$
for $\tau = 1$ **to** L_T **do**
  Let $t = t_\tau$
  Observe c_t
  Play random bundle b_τ from B
  Receive profit $r_t = \langle \phi(b_\tau), g_t - c_t \rangle$
  $A.2$: Estimate valuations $\bar{f}_\tau$ of B
  $A.3$: Find supermodular $\tilde{g}_\tau$ almost consistent with $\tau^{-1} \sum_{j=1}^{\tau} \bar{f}_\tau$
  for $t \in X_\tau$ (exploitation stage) **do**
    Observe c_t.
    Play $S_t^* = \arg\max_S \langle \phi(S), \tilde{g}_\tau - c_t \rangle$.
  end for
end for

mates with an interaction function $\tilde{g}$ that corresponds to a supermodular function in F_k (see $A.3$). On exploitation rounds, the algorithm uses $\tilde{g}$ and the cost function c_t to choose its bundles, by solving

$$\arg\max_S \langle \phi(S), \tilde{g} - c_t \rangle \quad .$$

The implementation and analysis of these steps form the bulk of this section and its subsections.

The exploration and exploitation steps are combined by Algorithm 1 according to a well known schedule (Langford & Zhang, 2008). The algorithm partitions the available T time steps into L_T epochs. Epoch τ begins with an exploration round at time t_τ, followed by multiple exploitation rounds at times $X_\tau \subset \{t_{\tau+1}, \ldots, T\}$. The number of exploitation rounds in each epoch will be specified later. Our objective is to make sure that each round of the algorithm can be completed in time polynomial in n and that the expected regret increases sub-linearly.

Theorem 2. *When the expected valuation is supermodular and every valuation is in F_2, steps $A.1$, $A.2$ and $A.3$ in Algorithm 1 can be specified so that each step is completed in polynomial time and the total expected regret is $\tilde{O}\left(T^{2/3} n^2\right)$.*

Efficient exploration is crucial in bandit problems in which the number of arms is exponential in the number of independent parameters. For example, when the arms are paths from a source to a target through a known graph, and the cost of an arm is the sum of weights of edges along the path. Exploring by trying paths from the full set at random is inefficient compared to finding a polynomial subset of paths from which the parameters can be estimated. A general theoretical framework for such problems uses so-called approximate barycentric spanners, but these are not always easy to find; see Awerbuch & Kleinberg (2004). Instead, in $A.1$ we choose an exploration set B that includes all the bundles of size at least $n-k$; we show in Section 3.1 that estimating the valuations of bundles in B induces an estimate of all possible bundles.

Using a standard estimator given in Section 3.1, the algorithm (see $A.2$) obtains an unbiased estimate of the valuations of bundles in B. The variance of this estimate is $O(n^k)$, and therefore the empirical mean of a sequence of estimates converges to its expectation at a rate that is polynomial in n, for fixed k.

In A.3 our goal is to find a sequence of supermodular interaction functions $\tilde{g}_\tau$ that similarly converges to $\mathbb{E}g_t$, since we assume the corresponding expected valuation is itself supermodular. Efficiently projecting the estimated valuation function onto F_k is difficult because the class of supermodular functions (thus also of interaction functions) is defined by an exponential number of linear constraints. In the special case where $k = 2$ there exist tractable representations of the supermodular polytope. In fact, the following theorem shows that in this case $A.3$ can be implemented simply by zeroing the positive outputs of $\tilde{g}_\tau$. Efficiently projecting onto F_k, for arbitrary k, remains an open question.

Theorem 3. *Let $\mathcal{P}$ denote the set of all pairs in E. The function f is supermodular and has degree 2 if and only if its interaction function g fulfills the following:*

$$(i) \quad \forall S \in \mathcal{P} \qquad g(S) \leq 0, \text{ and}$$
$$(ii) \quad \forall S \in 2^E \text{ s.t. } |S| > 2 \qquad g(S) = 0.$$

3.1. Expected regret bound

In this section we assume the algorithm is specified as follows. The exploration set B is defined to be

$$B = \{E \backslash S : |S| \leq k\}$$

and we denote $d = |B|$.

In $A.2$ we use the observation that

$$r_t = \langle \phi(b_\tau), g_t - c_t \rangle$$

and the knowledge of c_t to compute

$$f_\tau = \langle \phi(b_\tau), g_t \rangle$$

and define the vector

$$\bar{f}_\tau(S) = \begin{cases} df_\tau & \text{if } b_\tau = S, \\ 0 & \text{otherwise} \end{cases} \quad .$$

Lemma 4. *The vector $\bar{f}_\tau$ is an unbiased estimator of valuations of all items in B with regard to the choice of b_τ.*

To specify $A.3$ we first define

$$\bar{g}_\tau(S) = \bar{f}(E) - \bar{f}(E \backslash S) - \sum_{\emptyset \neq T \subsetneq S} \bar{g}(T)$$

inductively in $|S|$ and project to obtain

$$\tilde{g}_\tau = \arg\min_{g \in \mathcal{S}} \left\| g - \frac{1}{\tau} \sum_{i=1}^{\tau} \bar{g}_i \right\|_1,$$

where $\mathcal{S}$ is the polytope of interaction functions that correspond to supermodular valuation functions in F_2.

This is motivated by the following result.

Lemma 5. *For any $S \subset E$, we have*

$$\langle \phi(S), (\tilde{g}_\tau - c_t) - (\mathbb{E}g_t - c_t) \rangle \leq 2 \left\| \frac{1}{\tau} \sum_{i=1}^{\tau} \bar{g}_i - \mathbb{E}\bar{g} \right\|_1.$$

The lemma follows from the Hölder and triangle inequalities, and the definition of $\tilde{g}_\tau$.

Lemma 5 shows that precise estimates of g lead to precise valuations of all bundles. The next result quantifies the rate at which exploration leads to precise estimates of g.

Lemma 6. *After τ exploration rounds, for $r \in (0,1)$, we have with probability at least $\exp\left\{\log(d) - r^2\tau / \left(72d^3\right)\right\}$ that*

$$2 \left\| \frac{1}{\tau} \sum_{i=1}^{\tau} \bar{g}_i - \mathbb{E}\bar{g} \right\|_1 \leq r.$$

The probabilistic analysis for Lemma 6 and the further derivation required to produce the following proposition are deferred to Appendix B.

Proposition 7. *Algorithm 1, when run with*

$$|X_\tau| = \sqrt{\frac{\tau}{1152d^3\left(\log \tau^{1/2} + \log d\right)}},$$

for at least 60 epochs and with $n > 4$, achieves a regret of

$$R_T \leq 15 \left(8T\sqrt{2d^3 \log(d)}\right)^{2/3} \ln\left(8T\sqrt{2d^3 \log(d)}\right).$$

Proposition 7 implies Theorem 2 for $n > 4$, and the same result holds with different constants for $n \in \{2,3,4\}$.

4. Experiments

We implemented our algorithm and conducted preliminary experiments on simulated data. The goals of our empirical study are to observe the algorithm's performance in practice, and to see the effect of different set sizes n and different levels of noise.

Recall that our algorithm begins by selecting an appropriate set of exploration bundles B, and that for F_2 this set can include $1 + n + \binom{n}{2}$ bundles: The bundle E, $E \setminus \{a\}$ for every $a \in E$, and $E \setminus \{a, b\}$ for every $a, b \in E$. Rather than the randomized exploration prescribed in Algorithm 1 (play a random element from B), we found it easier to implement a deterministic round-robin exploration of each bundle in B. In other words, the algorithm periodically goes into exploration mode, and plays each bundle in B once. The theoretical guarantees still hold in this case. The explore-exploit schedule is still maintained, as the exploration mode occurs less frequently with time.

We ran experiments with sets of size $n = 8, 16, 32, 64$. We note that from a computational standpoint, our naïve algorithm implementation could have easily scaled to larger sets. In each experiment, we began by generating a random supermodular function in F_2 by selecting each value of the interaction function independently. The values that correspond to singletons were chosen uniformly in $[0, 1]$, while the values that correspond to pairs were chosen uniformly in $[-\frac{1}{2n}, 0]$. As noted above, setting the singletons to positive values and the pairs to negative values ensures that the resulting valuation function is indeed supermodular. Note that if we had set the values that correspond to pairs in $[-1, 0]$, the resulting supermodular function would almost always attain its maximum valuation at 0, which is an uninteresting case.

On each round, we chose a random cost function, with values independently chosen in $[0, 1]$. We also added random Gaussian noise to the valuation function before applying it to the selected bundle. We experimented with three different noise levels, $\sigma = 0.001, 0.01, 0.1$. We observed that higher levels of noise disrupted the algorithm to the extent that we could not see any learning progress even after 10K rounds. Since we had the valuation function explicitly, we were able to compute the optimal bundle on each round, and to calculate regret. We repeated each experiment 8 times, and averaged results are presented in Figure 1.

Several observations can be made from the figure. First, the empirical regret curves indeed look like $t^{2/3}$, as predicted by the theory. Second, as expected increased noise causes more regret. Third, we notice

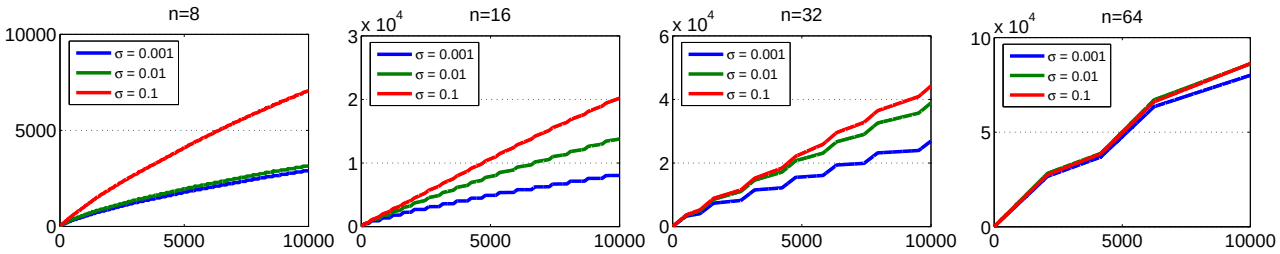

Figure 1. Regret as a function of number of rounds, with different set sizes ($n = 8, 16, 32, 64$) and noise levels (Gaussian noise with a standard deviation of 0.001, 0.01, 0.1)).

the anticipated step pattern that results from our deterministic exploration mode, which occurs once in a while (according to the Epoch-Greedy schedule) and lasts exactly $1 + n + \binom{n}{2}$ rounds.

Interestingly, different set sizes react differently to the different levels of noise. For example, for $n = 8$, increasing the noise level by a factor of ten (from 0.001 to 0.01) had almost no effect on regret, while the same cannot be said for $n = 16$. When the number of items was relatively large ($n = 64$) half of the 10K rounds were spent in exploration mode, and the asymptotic regret guaranteed by our theory has not yet kicked in. Overall, there are no surprises, and our algorithm performs just as the theory predicts.

5. Conclusions and future work

Learning a supermodular valuation function is in general a difficult problem (Balcan & Harvey, 2010). Under the assumption that only pair-wise interactions affect the valuation, we showed that the valuation function can not only be elicited offline, but learned online from noisy data. Our assumptions do not require prior knowledge about independence of particular item sets.

Our work solves a special type of the contextual multi-armed bandit problem with exponentially many arms. Such problems are made tractable by controlling the time and learning complexity by the "degree" of the model, in this case the valuation function, rather than the number of arms. Our definition of degree allowed us also to perform the projection of the valuation function to the set of supermodular (low degree) functions efficiently.

Our analysis focuses on F_2: the set of supermodular functions with interactions in pairs. The main challenge we have is to extend our results to more complex interaction structures. The major obstacle in meeting this challenge is the projection step of our algorithm. The projection is needed for two reasons. First, the true valuation function may not be supermodular and in this case we need to find the closest supermodular function in an appropriate sense. Second, the estimated valuation function may not be supermodular even if the true valuation is (due to the noise). The polytope of possible parameters for F_2 requires only a polynomial subset of the generally exponentially many constraints, but this does not hold for higher degrees. It is not yet clear whether efficient projections can be extended to more complex interaction structure such as F_3 or interactions that are described by a given set of possible interactions whose size is small.

Supermodularity and submodularity have found many applications in machine learning in recent years. Our work is novel in that it addresses the contextual problem where there is a cost that is associated with each item that changes from round to round. It is clear that the general problem is hard, but the question that remains is when are such problems solvable effectively? Our work provides a partial answer to that question, but a more general answer remains to be found.

Acknowledgements. Parts of this research were conducted while D. V. and S. M. were visiting Microsoft Research. This research was supported in part by the Google Inter-university center for Electronic Markets and Auctions. We thank Mohit Singh for helpful discussions.

References

Auer, P. Using confidence bounds for exploitation-exploration trade-offs. *The Journal of Machine Learning Research*, 3:397–422, 2003.

Awerbuch, B. and Kleinberg, R.D. Adaptive routing with end-to-end feedback: Distributed learning and geometric approaches. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pp. 45–53. ACM, 2004. ISBN 1581138520.

Balcan, M.F. and Harvey, N.J.A. Learning submodular functions. *Arxiv preprint arXiv:1008.2159*, 2010.

Cesa-Bianchi, N. and Lugosi, G. *Prediction, learning, and games*. Cambridge Univ Pr, 2006. ISBN 0521841089.

Chajewska, U. and Koller, D. Learning the Structure of Utility Functions. *Unpublished report. Stanford University, Stanford CA*, 1999.

Chajewska, U., Koller, D., and Ormoneit, D. Learning an agent's utility function by observing behavior. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML)*, pp. 35–42, 2001.

Dani, V., Hayes, T.P., and Kakade, S.M. Stochastic linear optimization under bandit feedback. In *Proceedings of the 21st Annual Conference on Learning Theory*. Citeseer, 2008.

Hazan, E. and Kale, S. Beyond convexity: Online submodular minimization. In *Advances in Neural Information Processing Systems 22*, pp. 700–708. MIT Press, 2009.

Langford, J. and Zhang, T. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in Neural Information Processing Systems 20*, pp. 817–824. MIT Press, 2008.

Li, L., Chu, W., Langford, J., and Schapire, R.E. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pp. 661–670. ACM, 2010.

McCormick, S. T. Submodular function minimization. In Aardal, K., Nemhauser, G., and Weismantel, R. (eds.), *Handbook on Discrete Optimization*, pp. 321–391. Elsevier, 2006.

Nielsen, T.D. and Jensen, F.V. Learning a decision maker's utility function from (possibly) inconsistent behavior. *Artificial Intelligence*, 160(1-2):53–78, 2004. ISSN 0004-3702.

Streeter, M. and Golovin, D. An online algorithm for maximizing submodular functions. In *Advances in Neural Information Processing Systems 21*, pp. 1577–1584. MIT Press, 2009.

Topkis, D. M. *Supermodularity and Complementarity*. Princeton University Press., 1998.

Vainsencher, D., Dekel, O., and Mannor, S. Bundle selling by online estimation of valuation functions. *Arxiv*, 2011.

Wang, C.C., Kulkarni, S.R., and Poor, H.V. Bandit problems with side observations. *Automatic Control, IEEE Transactions on*, 50(3):338–355, 2005. ISSN 0018-9286.

A. Proof of interaction function properties

Proof. (Of Proposition 1) Recall that E is the set of all possible items ("the ground set"). Let S denote a subset of E, and consider $f(N) - f(N \backslash S) = \sum_{T \cap N \neq \emptyset} g(T) - \sum_{T \cap (N \backslash S) \neq \emptyset} g(T)$. Since $T \cap (N \backslash S) \neq \emptyset \Rightarrow T \cap N \neq \emptyset$, it follows that T appears in the second sum only if it appears in the first. Then $f(N) - f(N \backslash S) = \sum_{T \cap N \neq \emptyset \wedge T \cap (N \backslash S) = \emptyset} g(T)$. So that $f(N) - f(N \backslash S) = \sum_{\emptyset \neq T \subseteq S} g(T) = g(S) + \sum_{\emptyset \neq T \subsetneq S} g(T)$.

Now given a set-function f, we can define $g(S)$ by induction on $|S|$:

$$g(S) = f(N) - f(N \backslash S) - \sum_{\emptyset \neq T \subsetneq S} g(T).$$

We conclude that g is determined linearly by f. □

Let A, B be sets, then we write $A \perp B$ if they are disjoint.

Lemma 8. *The function f is supermodular if and only if for all $A \subseteq E$ and $B \subseteq E$ it holds that $\sum_{S \in \mathcal{D}_{A,B}} g(S) \leq 0$, where $\mathcal{D}_{A,B} = \left\{ S \in 2^E \;:\; S \not\perp A \;,\; S \not\perp B \;,\; S \perp (A \cap B) \right\}$.*

This lemma is proved in the full version. Equipped with Lemma 8, we can prove Theorem 3.

Proof. (Of Theorem 3) The constraint (ii) is necessary and sufficient for f to have degree 2. Next, we prove that the constraint (i) is both necessary and sufficient for supermodularity.

Let $\{a, b\}$ be an item of $\mathcal{P}$. Define the sets

$$A = E \setminus \{b\} \qquad \text{and} \qquad B = E \setminus \{a\} \;\; .$$

Using Lemma 8, we know that a necessary condition for supermodularity is

$$\sum_{S \in \mathcal{D}_{A,B}} g(S) \;\leq\; 0 \;\; .$$

Note that $\mathcal{D}_{A,B}$ does not contain any singletons, and that the only pair in $\mathcal{D}_{A,B}$ is $\{a, b\}$. Since all other sets in $\mathcal{D}_{A,B}$ have $g(S) = 0$, we conclude that

$$\sum_{S \in \mathcal{D}_{A,B}} g(S) \;=\; g(\{a, b\}) \;\; .$$

This proves that the constraints of (i) are necessary.

Next, note that (i) and (ii) imply that $\sum_{S \in \mathcal{D}} g(S) \leq 0$ for any $\mathcal{D} \subseteq 2^E$, and specifically for $\mathcal{D}_{A,B}$ (where A and B are arbitrary subsets of E). This proves that the sufficient conditions for supermodularity given by Lemma 8 are satisfied. □

B. Proofs of regret results

Proof. (Of Lemma 6) We begin by analyzing the support and variance of $X_i^S = |\bar{g}_i(S) - \mathbb{E}\bar{g}(S)|$. Then an application of Bernstein inequality bounds the probability of a deviation of the average of the estimates from their expected value. A union bound completes the proof of the lemma.

The estimates $\bar{g}_i$ for different exploration times i are IID vector variables. It follows that $\bar{g}_i(s) = \bar{f}_i(E) - \bar{f}_i(E\backslash\{s\}) \in d[-1,1]$ because $f(S) \in [0,1]$. Similarly, $\bar{g}_i(\{s,t\}) = -\bar{f}_i(E) - \bar{f}_i(E\backslash\{s,t\}) + \bar{g}_i(s) + \bar{g}_i(t) \in 2(d+1)[-1,1]$. Then each scalar quantity in each vector is supported on $3d[-1,1]$.

The bound on the variance takes some effort because values of $\bar{g}$ on different subsets S are dependent through the choice of b_τ. But since the choice of b_τ is independent of f_t, we have $\mathrm{Var}\left(\bar{f}(S)\right) = \left(\mathbb{E}f(S)\right)^2(d-1) + d \cdot \mathrm{Var}(f(S)) \le 2d$ using the assumption that $f_t(S) \in [0,1]$. It is easy to verify that $\mathrm{Cov}\left(\bar{f}(E), \bar{f}(E\backslash S)\right) \le 0$, so we conclude that for a set S that is either a singleton or a pair we have $\mathrm{Var}(\bar{g}(S)) \le 8|B|$.

Applying Bernstein with the bounds obtained so far, we have $P\left(\frac{1}{\tau}\sum_{i=1}^n X_i^S > q\right) \le \exp\left\{-\frac{q^2\tau}{2d(8+q)}\right\}$.

A union bound over the sets in B, leads to

$$P\left(\left(\max_{S\in B}\frac{1}{\tau}\sum_{i=1}^n X_i^S\right) > q\right) \le \exp\left\{\ln d - \frac{q^2\tau}{2d(8+q)}\right\}.$$

To achieve a bound on $\left\|\frac{1}{\tau}\sum_{i=1}^\tau \bar{g}_i - \mathbb{E}\bar{g}\right\|_1$, it is enough to require that $\left\|\frac{1}{\tau}\sum_{i=1}^\tau \bar{g}_i - \mathbb{E}\bar{g}\right\|_\infty \le \frac{r}{2d}$, which corresponds to the condition in the lemma. □

Proof. (Of Proposition 7) Our proof consists of the following steps. We first bound the regret in an exploitation step; we then determine the number of exploitation steps that may be taken in an epoch without exceeding a contribution of the exploitation to the regret that exceeds 1. This implies that each epoch causes a regret of at most 3. We finally compute how many epochs might be started in T time steps.

We first note the regret of an exploitation step in epoch τ is at most $\sqrt{288d^3(x+\ln d)/\tau}$ with probability at least $1-\exp(-x)$. Algebraic manipulation implies that $\exp\left\{\ln d - r^2\tau/\left(72d^3\right)\right\} \le \exp\{-x\} \iff r \ge \sqrt{\frac{72d^3(x+\ln d)}{\tau}}$. If the RHS is greater than 1, then twice as much is trivially greater than the regret of any step. Otherwise, we choose r with equality there. From earlier results we conclude with probability at least $1-\exp(-x)$, that $\left|\left\langle\phi(S), \left(\Phi^{-1}\gamma_\tau - c_t\right) - (\mathbb{E}g_t - c_t)\right\rangle\right| \le \sqrt{72d^3(x+\ln d)/\tau}$. At worst we will choose a bundle that is worse than the best by twice that much. For compactness, we denote below $a = 288d^3$.

The regret of a single exploration step is at most 2 (the difference between maximal valuation and maximal cost). The expected regret of an exploitation step is bounded by $\sqrt{\frac{a(x+\ln d)}{\tau}} + 2\exp(-x)$, we take $\exp(-x) = \tau^{-1/2} \iff x = \ln\tau^{1/2}$, then the bound on the expected regret is $\sqrt{\frac{a\left(\ln\tau^{1/2}+\ln d\right)}{\tau}} + 2\tau^{-1/2} \le \frac{\sqrt{a\left(\ln\tau^{1/2}+\ln d\right)}+2}{\sqrt{\tau}} \le \frac{\sqrt{4a\left(\ln\tau^{1/2}+\ln d\right)}}{\sqrt{\tau}}$.

Choosing $|X(\tau)|$ as in the proposition implies that the expected total regret of an epoch is upper bounded by 3. Having determined the length and total regret of an epoch, we turn to inverting the relationship to bound the number of epochs and therefore the total regret up to some particular time.

At time T, we have expected regret bounded by $3q$, where q is the number of epochs that cover T: the minimal q such that $t(q) = q + \sum_{\tau=1}^q \sqrt{\frac{\tau}{4a\left(\ln\tau^{1/2}+\ln d\right)}} \ge T$.

$$\begin{aligned} t(q) &\ge \sqrt{\frac{1}{4a\left(\ln q^{1/2}+\ln d\right)}}\sum_{\tau=1}^q\sqrt{\tau} \\ &\ge \sqrt{\frac{1}{4a\left(\ln q^{1/2}+\ln d\right)}}\frac{3}{2}q^{3/2} \end{aligned}$$

and $q^{1/2}, d > e^2$ so

$$\begin{aligned} &\ge \sqrt{\frac{1}{4a\ln(d)\ln\left(q^{1/2}\right)}}\frac{3}{2}q^{3/2} \\ &\ge \sqrt{\frac{1}{4a\ln(d)}}\frac{3}{2}\left(\frac{q}{\ln q}\right)^{3/2}. \end{aligned}$$

We now choose q such that $\sqrt{\frac{1}{4a\ln(d)}}\frac{3}{2}\left(\frac{q}{\ln q}\right)^{3/2} \ge T \iff \frac{q}{\ln q} \ge \left(\frac{4T}{3}\sqrt{a\ln(d)}\right)^{2/3}$. From Lemma 9, it is enough to take $q = 5\left(\frac{4T}{3}\sqrt{a\ln(d)}\right)^{2/3}\ln\left(\frac{4T}{3}\sqrt{a\ln(d)}\right)$. Combining the results, it follows that the regret at time T is no more than $15\left(\frac{4T}{3}\sqrt{a\ln(d)}\right)^{2/3}\ln\left(\frac{4T}{3}\sqrt{a\ln(d)}\right) \in \tilde{O}\left(T^{2/3}a^{1/3}\right)$. □

Lemma 9. *Let $x \ge 2$ and $q = 5x^{2/3}\ln x$. Then $q/\ln q \ge x^{2/3}$.*

The proof is included in the full version.