Maximizing Profit with Convex Costs in the Random-order Model*

Anupam Gupta  Ruta Mehta  Marco Molinaro

Abstract

Suppose a set of requests arrives online: each request gives some value \( v_i \) if accepted, but requires using some amount of each of \( d \) resources. Our cost is a convex function of the vector of total utilization of these \( d \) resources. Which requests should be accept to maximize our profit, i.e., the sum of values of the accepted demands, minus the convex cost?

We consider this problem in the random-order a.k.a. secretary model, and show an \( O(d) \)-competitive algorithm for the case where the convex cost function is also supermodular. If the set of accepted demands must also be independent in a given matroid, we give an \( O(d^3 \alpha) \)-competitive algorithm for the supermodular case, and an improved \( O(d^2 \alpha) \) if the convex cost function is also separable. Here \( \alpha \) is the competitive ratio of the best algorithm for the submodular secretary problem. These extend and improve previous results known for this problem. Our techniques are simple but use powerful ideas from convex duality, which give clean interpretations of existing work, and allow us to give the extensions and improvements.

1 Introduction

The problem we consider is a basic convex optimization problem in the online setting: \( n \) items appear one-by-one. Each item/element \( e \) has a \( d \)-dimensional size \( s(e) \in \mathbb{R}^d_+ \) and a value \( v(e) \in \mathbb{R}_+ \), which are both revealed to us when the item arrives. We must either accept or reject an item when it arrives, before seeing the future items. If we accept a certain subset \( A \subseteq [n] \) of the items, we get their total value \( v(A) := \sum_{e \in A} v(e) \), but incur a production cost \( g(s(A)) := g(\sum_{e \in A} s(e)) \), where \( g : \mathbb{R}^d_+ \to \mathbb{R}_+ \) is a non-decreasing convex cost function with \( g(0) = 0 \).

Optionally, we may also be given a downwards-closed family of subsets \( \mathcal{F} \subseteq 2^{[n]} \), and now the accepted set of elements \( A \) must lie in \( \mathcal{F} \). More formally, we want to solve

\[
\max_{A \in \mathcal{F}} \text{profit } \pi(A) := [v(A) - g(s(A))].
\]  

This question arises, e.g., when we are selling some service that depends on \( d \) commodities, where the value is the amount of money customer \( e \) is willing to pay for the service, and the size vector \( s(e) \) is the amount of resources she will require. The cost function \( g(\cdot) \) captures our operating expenses; its convexity models diseconomies of scale that arise when dealing with scarce commodities. In particular, it can capture \( d \)-dimensional knapsack constraints, by setting \( g(z) = 0 \) until the knapsack size, and \( \infty \) afterwards. When the cost function is linear \( g(z) = \langle a, z \rangle \), we want to pick a max-weight subset from \( \mathcal{F} \) using item weights \( v(e) - \langle a, s(e) \rangle \), which is tractable/approximable for \( \mathcal{F} \) being a matroid, \( p \)-system, etc.

Blum et al. [BGMT11] defined this problem in the adversarial model, and gave posted-price algorithms for “low-degree” separable cost functions \( g \), that is, of the form \( g(z) = \sum_{i=1}^d g_i(z_i) \) for 1-dimensional functions \( g_i \)’s. This result was tightened by Huang and Kim [HKT15], still for separable functions with additional growth control. More recently, Azar et al. [ABC\textsuperscript{+}16] studied this problem for more general supermodular non-separable convex functions \( g \) (see also [EP16]). A differentiable function \( g \) is supermodular if for any vectors \( x \leq x' \) we

\*This work was done in part while the authors were visiting the Simons Institute for the Theory of Computing.
have $\nabla g(x) \leq \nabla g(x')$. Equivalently, if $g$ is twice-differentiable, it is supermodular if $\frac{\partial^2 g}{\partial x_i \partial x_j} \geq 0$ for all $i \neq j$, i.e., increasing the consumption of a resource cannot decrease the marginal cost for another. However, to handle the worst-case ordering, Azar et al. also require the cost functions to have essentially low-degree.

Can we do better by going beyond the worst-case model? In this paper, we focus on the random-order or “secretary” setting, where the set of items is fixed by an adversary but they arrive in random order. In the single-dimensional case $d = 1$, it is easy to see that a solution that learns a “good” threshold $\lambda$ and picks all further items with density $v(e)/s(e)$ at least $\lambda$ essentially gives a constant approximation, much like in the secretary and knapsack secretary problems [Fre83, BIKK07]. The multi-dimensional case is much more challenging. This was studied by Barman et al. [BUCM12], again assuming a separable cost function $g(z) = \sum_{i=1}^d g_i(z_i)$. They give an $O(d)$-competitive algorithm for the unconstrained case, and an $O(d^5\alpha)$-competitive algorithm for the problem with a downward closed constraint set $\mathcal{F}$, where $\alpha$ is the competitive ratio for the $\mathcal{F}$-secretary problem. Their main idea is to perform a clever decomposition of the value of each item into “subvalues” $v_i(e)$ for each of the coordinate cost functions $g_i$’s; this effectively decomposes the problem into $d$ 1-dimension problems with values $v_i$’s and costs $g_i$’s. Unfortunately, since their solution explicitly relies on the decomposability of the cost function, it is unclear how to extend it to general supermodular functions. We note that when the cost function is supermodular, the profit function is a submodular set function (Section 2.1). However, the profit can take negative values, and then existing algorithms for submodular maximization break down.

Our work is then motivated by trying to better understand the multi-dimensional nature of this problem, and provide a more principled algorithmic approach.

1.1 Our Results

We use techniques from convex duality to re-interpret, simplify, and improve the existing results. First, we obtain the first approximation for non-separable supermodular cost functions. (We omit some mild regularity conditions for brevity; see Section 3 for full details.)

**Theorem 1.1** (Unconstrained & Supermodular). *For the unconstrained problem with supermodular convex cost functions $g$, we give an $O(d)$-competitive randomized algorithm in the random-order model.*

This result generalizes the $O(d)$-approximation of Barman et al. [BUCM12] to the non-separable case. The factor $d$ seems unavoidable, since our problem inherits the (offline) $\Omega(d^{1-\varepsilon})$ hardness of the $d$-dimensional knapsack, assuming $NP \neq ZPP$ [DCG05].

Next, we consider the constrained case. For simplicity, we focus on the most interesting case where $\mathcal{F}$ is a matroid constraint; more general results can be obtained from the results and techniques in Section 5.

**Theorem 1.2** (Constrained & Separable). *For the constrained problem with $\mathcal{F}$ being a matroid constraint, and the cost function $g$ being separable, we get an $O(d^2 \log \log \text{rank})$-competitive randomized algorithm in the random-order model.*

This improves by a factor of $d^3$ the $O(d^5 \log \log \text{rank})$-approximation given by [BUCM12]. Finally, we give a general reduction that takes an algorithm for separable functions and produces an algorithm for supermodular functions, both with respect to a matroid constraint. This implies:

**Theorem 1.3** (Constrained & Supermodular). *For the constrained problem with $\mathcal{F}$ being a matroid constraint, and the cost function $g$ being supermodular, we get an $O(d^3 \log \log \text{rank})$-competitive randomized algorithm in the random-order model.*

---

1For example, we can model set packing (which is $\Omega(\sqrt{\# \text{ sets}})$-hard) as follows: for a subcollection $S$ of sets, let $\pi(S) = |\bigcup_{S \in S} S| - \sum_{S \in S} (|S| - 1)$. The function $\pi$ is submodular, and its maximizer is a largest set packing.
On conceptual contributions are in bringing techniques from convex duality to obtain, in a principled way, threshold-based algorithms for non-linear secretary problems. Since this is a classical and heavily used algorithmic strategy for secretary problems [Fre83, BIKK07, Kle05, AWY14, MR14] we hope that the perspectives used here will find use in other contexts.

1.2 Other Related Work

There is a vast literature on secretary problems [Fre83]. Closest to our setting, Agrawal and Devanur study an online convex optimization problem in the random order model, and give a powerful result showing strong regret bounds in this setting [AD15]. They extend this result to give algorithms for online packing LPs with “large” right-hand sides. However, it is unclear how to use their algorithm to obtain results in our setting. Other algorithms solving packing LPs with large right-hand sides appear in [AWY14, DJSW11, MR14, KRTV14, GM16, ESF14].

Feldman and Zenklusen [FZ15] show how to transform any algorithm for (linear) matroid secretary into one for submodular matroid secretary. They give an $O(\log \log \text{rank})$-algorithm for the latter, based on results of [Lac14, FSZ15]. All these algorithms critically assume the submodular function is non-negative everywhere, which is not the case for us, since picking too large a set may cause the profit function to go negative. Indeed, one technical contribution is a procedure for making the profit function non-negative while preserving submodularity (Section 4.1), which allows us to use these results as part of our solution.

1.3 Structure of the paper

Section 3 develops the convex duality perspective used in the paper for the offline version of the unconstrained case, hopefully in an manner accessible to non-experts. Section 4 gives the small changes required to extend this to the constrained case. Section 5 shows how transform these into online algorithms. Section 6 shows how to convert an algorithm for separable functions into one for supermodular functions, both subject to matroid constraints. To improve the presentation, we make throughout convenient assumptions, which are discharged in Appendix C.

Since some familiarity with convex functions and conjugates will be useful, we give basic facts about them and some probabilistic inequalities in Appendix A.

2 Preliminaries

Problem Formulation. Elements from a universe $U$ of size $n$ are presented in random order. Each element $e$ has value $v(e) \in \mathbb{R}_+$ and size $s(e) \in \mathbb{R}_d^+$. We are given a convex cost function $g : \mathbb{R}_d^+ \rightarrow \mathbb{R}_+$. On seeing each element we must either accept or discard it. A downwards-closed collection $\mathcal{F} \subseteq 2^U$ of feasible sets is also given. When $\mathcal{F} = 2^U$, we call it the unconstrained problem. The goal is to pick a subset $A \in \mathcal{F}$ to maximize the profit

$$\pi(A) := \sum_{e \in A} v(e) - g\left(\sum_{e \in A} s(e)\right).$$

(2.2)

We often use vectors in $\{0, 1\}^n$ to denote subsets of $U$; $\chi_A$ denotes the indicator vector for set $A$. Hence, $\mathcal{F} \subseteq \{0, 1\}^n$ is a down-ideal on the Boolean lattice, and we can succinctly write our problem as

$$\max_{x \in \mathcal{F}} \pi(x) := \langle v, x \rangle - g(Sx),$$

(2.3)

where columns of $S \in \mathbb{R}^{d \times n}$ are the item sizes. Let opt denote the optimal value. For a subset $A \subseteq U$, $v(A)$ and $s(A)$ denote $\sum_{e \in A} v(e) = \langle v, \chi_A \rangle$ and $\sum_{e \in A} s(e) = S\chi_A$ respectively.
Definition 2.1 (Exceptional). \textit{Item }e \in U \textit{ is exceptional if }\arg \max_{g \in [0,1]} \{ \theta v(e) - g(\theta s(e)) \} \in (0, 1).

Definition 2.2 (Marginal Function). \textit{Given }g : \mathbb{R}^d \to \mathbb{R}, \text{ define the }i^{th} \text{ marginal function }g_i : \mathbb{R} \to \mathbb{R} \text{ as }g_i(x) := g(xe_i), \text{ where }e_i \text{ is the }i^{th} \text{ standard unit vector.}

2.1 Supermodular Functions

While supermodular functions defined over the Boolean lattice are widely considered, one can define supermodularity for all real-valued functions. Omitted proofs are presented in Appendix B.1.

Definition 2.3 (Supermodular). \textit{Let }X \subseteq \mathbb{R}^d \textit{ be a lattice. A function }f : X \to \mathbb{R} \textit{ is supermodular if for all }x, y \in X, f(x) + f(y) \leq f(x \land y) + f(x \lor y), \textit{ where }x \land y \textit{ and }x \lor y \textit{ are the component-wise minimum and maximum operations.}

This corresponds to the usual definition of (discrete) supermodularity when \(X = \{0, 1\}^d\). For proof of the lemma below and other equivalent definitions, see, e.g., [Top98].

Lemma 2.4 (Supermodularity and Gradients). \textit{A convex function }f : \mathbb{R}_+^d \to \mathbb{R} \textit{ is supermodular if and only if any of the following are true.}

\begin{itemize}
  \item \(\nabla f\) \textit{ is increasing in each coordinate, if }f \textit{ is differentiable.}
  \item \(\frac{\partial^2 f(x)}{\partial x_i \partial x_j} \geq 0 \) for all \(i, j\), if \(f\) is twice-differentiable.
\end{itemize}

Lemma 2.5 (Superadditivity). \textit{If }f : \mathbb{R}_+^d \to \mathbb{R} \textit{ is differentiable, convex, and supermodular, then for }x, x', y \in \mathbb{R}_+^d \textit{ such that }x' \leq x, f(x' + y) - f(x') \leq f(x + y) - f(x). \textit{In particular, if }f(0) = 0, \textit{ setting }x' = 0 \textit{ gives }f(x) + f(y) \leq f(x + y).

Corollary 2.6 (Subadditivity of profit). \textit{The profit function }\pi \textit{ is subadditive.}

The next fact shows that the cost \(g\) is also supermodular when seen in a discrete way.

Fact 2.7 (Continuous vs. Discrete Supermodularity). \textit{Given a convex supermodular function }g : \mathbb{R}^d \to \mathbb{R} \textit{ and }n \textit{ items with sizes }s_1, \ldots, s_n \in \mathbb{R}_+^d, \textit{ define the function }h : \{0, 1\}^n \to \mathbb{R} \textit{ as }h(v) = g(\sum_i s_i v_i) = g(Sv). \textit{Then }h(\cdot) \textit{ is a (discrete) supermodular function.}

3 The Offline Unconstrained Problem

We first present an offline algorithm for supermodular functions in the \textbf{unconstrained} case (where \(\mathcal{F} = \{0, 1\}^n\)). We focus on the main techniques and defer some technicalities and all computational aspects for now. Just for this section, we assume item sizes are “infinitesimal”. We make the following assumptions on the cost function \(g\) and the elements.

Assumption 3.1. \textit{We assume that cost function }g \textit{ is non-negative, strictly convex, closed, and differentiable. We assume }g(0) = 0, g \textit{ is supermodular, and that gradients of }g \textit{ go to }\infty \textit{ along every positive direction. We assume elements are in general position\footnote{There are no non-trivial linear dependencies, see Lemma C.2 for a formal definition} \textit{ and that there are no exceptional items. We also assume that every individual item has profit at most }M := \frac{\text{opt}}{\eta d} \textit{ for }\eta \geq 10^4. \textit{ (See Section C.2 on how to remove these assumptions on elements.)}

Classifiers. The offline algorithm will be based on \textit{linear classifiers}, where a set of weights is used to aggregates the multidimensional size of an item into a scalar, and the algorithm picks all items that have high-enough value/aggregated-size ratio.
Proof. For any solution $x$.

Claim 3.3. where the second inequality holds since, by definition, $x$ coordinates with the "0 marginal value"

The existence of such good classifiers is not enough, since we need to

Restricting the Set of Classifiers. The existence of such good classifiers is not enough, since we need to find them online. This is difficult not only because of $d$ degrees of freedom and no control over the magnitude of the values/sizes (to be exploited in concentration inequalities), but also because picking too few or too many items could lead to low profits. So we restrict the set of candidate classifiers to be a monotone $d$-dimensional curve $C \subseteq \mathbb{R}^d_+$, satisfying additional properties given below. The main motivation is that it imposes a total ordering on the set of items

$^3$A curve $C$ is monotone if for every pair $\lambda, \lambda' \in C$, one is coordinate-wise smaller than the other.
picked by the classifiers: given \( \lambda \leq \mu \) on such a curve \( C \), the sets of items picked satisfy the inclusion \( U_\lambda \supseteq U_\mu \). This allows us to select a “minimally good” classifier in \( C \) in a robust way, avoiding classifiers that select too many items.

To design the curve \( C \) so it contains a classifier with profit \( \approx \frac{\text{OPT}}{2} \), we relax the condition \( \nabla g(\text{occ}_\lambda) = \lambda \) from Claim 3.3 (too much to ask) and require the existence of \( \lambda \in C \) satisfying:

(P1) (don’t pick too many items) \( \nabla g(\text{occ}_\lambda) \leq \lambda \).

(P2) (partial gradient equality) There is a coordinate \( i^* \) where \( (\nabla g(\text{occ}_\lambda))_{i^*} = \lambda_{i^*} \).

(P3) (balanced curve) \( g_i^*(\lambda_i) = g_j^*(\lambda_j) \forall i, j \in [d] \) (see also Claim A.6).

Property (P1) enforces half of the equality in Claim 3.3 and (P2) guarantees that equality holds for some coordinate. Now for property (P3). Since \( \lambda \neq \nabla g(\text{occ}_\lambda) \) the optimality proof of Claim 3.3 does not go through, since \( g(\text{occ}_\lambda) \neq \langle \lambda, \text{occ}_\lambda \rangle - g^*(\lambda) \). As we prove later, the difference between these terms can be at most \( g^*(\lambda) \) (see Figure 1 for an illustration), and the superadditivity of \( g \) gives us \( g^*(\lambda) \leq \sum_i g_i^*(\lambda_i) \) (see Claim A.7). Property (P3) is used to control this sum, by charging it to the coordinate \( i^* \) where we know we have “the right linearization” (by property (P2)). Reinterpreting the construction of \( \text{BUCM12} \) in our setting, we then define \( C \) as any monotone curve where every \( \lambda \in C \) satisfies (P3).

**Lemma 3.4.** The curve \( C \) exists and contains a \( \lambda \) satisfying properties (P1)-(P3).

**Proof.** We first show existence, that is, the set \( \{ \lambda \in \mathbb{R}_d^+ \mid g_i^*(\lambda_i) = g_j^*(\lambda_j) \forall i, j \} \) contains a monotone curve. Notice that this set is the union of the box \( \{ \lambda \in \mathbb{R}_d^+ \mid g_i^*(\lambda_i) = 0 \forall i \} = \prod_i [0, g_i^*(0)] \) (range of slopes where we can swivel around \( g_i(0) = 0 \)) and a monotone curve \( \{ \lambda(\tau) \mid \tau > 0 \} \), where \( \lambda(\tau) \) is the unique vector satisfying \( g_i^*(\lambda_i(\tau)) = \tau \); uniqueness follows from the fact \( g_i^* \) stays at value zero in the interval \([0, g_i^*(0)]\), but after that is strictly increasing due to its convexity, and monotonicity of this curve also follows from monotonicity of the \( g_i^* \)s. Thus, \( C \) is this curve plus any monotone curve extending it to the origin.

To see that \( C \) satisfies properties (P1) and (P2), we note that since the \( g_i^* \)s are increasing and not identically 0, \( C \) is unbounded in all coordinates. Thus, a sufficiently large \( \lambda \in C \) satisfies (P1), and we can start with such \( \lambda \) and move down the curve (decreasing in each coordinate) until we obtain \( \lambda' \in C \) with \( \lambda' = \nabla g(\text{occ}_{\lambda'}) \), since the \( g \) has increasing gradients. (The equality in this final step uses the assumption that item sizes are infinitesimal, which we made for simplicity in this section).

Making the above discussion formal, we show that \( C \) has a high-value classifier. Recall that \( U_\lambda \) is the set of items picked by \( \lambda \) (Definition 3.2).

**Theorem 3.5.** Given Assumption 3.7 let \( \lambda^* \) be a classifier in \( C \) satisfying properties (P1)-(P3). Then for all \( x' \in [0, 1]^n \) we have \( \pi(U_{\lambda^*}) \geq \frac{1}{d+1} \cdot \pi(x') \).

**Proof.** Let \( x^* = \chi_{U_{\lambda^*}} \) be the solution picked by the classifier \( \lambda^* \), and note that \( \text{occ}_{\lambda^*} = Sx^* \). Let \( L(y, \mu) := \langle v, y \rangle - \langle \mu, S y \rangle - g^*(\mu) \rangle \) be the linearization of \( \pi(y) \) at some slope \( \mu \). From (3.4) we know \( g(y) \geq L(y, \mu) \) for all \( \mu \geq 0 \). Since \( x^* \) is optimal for the linearization \( L(y, \lambda^*) \) (because \( x_i^* = 1 \) iff \( v_i - \langle \lambda^*, S v \rangle \geq 0 \)), we have

\[
L(x^*, \lambda^*) \geq L(x', \lambda^*) \geq \pi(x') \quad \text{for all } x' \in [0, 1]^n. 
\]

(3.5)

Now we relate the true profit \( \pi(x^*) \) to this linearized value. Observe that

\[
\pi(x^*) = \langle v, x^* \rangle - g(Sx^*) = \langle v, x^* \rangle - [\langle \nabla g(Sx^*), Sx^* \rangle - g^*(\nabla g(Sx^*))]
\]

(by Claim A.2)

\[
\geq \langle v, x^* \rangle - \langle \lambda^*, Sx^* \rangle + g^*(\nabla g(Sx^*)), \quad \geq 0
\]

(3.6)
where the inequality uses that $\lambda^* \geq \nabla g(Sx^*)$ by property (P1) and $Sx^* \geq 0$. The first term is non-negative because we only pick items for which $v_i - \langle \lambda, S^i \rangle \geq 0$. The second term is non-negative due to Claim [A.5]a. We can now prove three lemmas that imply the theorem.

**Lemma 3.6.** For any $x' \in [0, 1]^n$, $\pi(x^*) \geq L(x^*, \lambda^*) - g^*(\lambda^*) \geq \pi(x') - g^*(\lambda^*)$.

**Proof.** Drop the second term from (3.6), then use the definition of $L(\cdot, \cdot)$ and (3.5). $\square$

**Lemma 3.7.** $g^*(\lambda^*) \leq d \cdot g^*_{\pi}(\lambda^*_\pi)$.

**Proof.** Using the superadditivity of $g$ and Claim [A.7] we get $g^*(\lambda^*) \leq \sum_i g^*_i(\lambda^*_i)$. Now from property (P3) of the classifier $\lambda^*$, all the terms in the sum are equal. $\square$

**Lemma 3.8.** $\pi(x^*) \geq g^*_\pi(\lambda^*_{\pi})$.

**Proof.** We claim that $g^*(\nabla g(Sx^*)) \geq g^*_{\pi}(\lambda^*_\pi)$; plugging this into (3.6) proves the lemma. For the claim, define $\lambda' = \nabla g(Sx^*)$. By Property (P2), $\lambda'_{\pi} = \lambda^*_\pi$, so we want to show $g^*(\lambda') \geq g^*_{\pi}(\lambda'_{\pi}) = g^*(\lambda'_{\pi}, e_{i^*})$. This follows because $g^*$ is monotone (Claim [A.5]b)). $\square$

This completes the proof of Theorem 3.5.

### 4 The Offline Constrained Case

Having built up tools and intuition in the unconstrained case, we turn to the case where there is a downwards-closed constraint $\mathcal{F} \subseteq \{0, 1\}^n$, and the goal is to maximize the profit subject to $x \in \mathcal{F}$. We again work with Assumption [3.1], but do not assume anything about items sizes. We discuss computational aspects at the end of this section.

The general idea is again to use classifiers $\lambda \in \mathbb{R}_+^d$, and only consider items in $U_\lambda$, namely those with “high-enough” value $v_i \geq \langle \lambda, S^i \rangle$. However, because of the constraints $\mathcal{F}$ we may no longer be able to pick all these items. Thus, we need to consider the most profitable solution from $\mathcal{F}$ in this filtered feasible set $U_\lambda$ (whose quality is less clear how to analyze).

Again we restrict to the 1-dimensional curve $\mathcal{C}$ defined in the previous section; however, it only satisfies slightly modified versions of properties (P1)-(P2), since we do not assume the item sizes to be infinitesimal anymore. To make this precise, define the “open” set $U^\circ_\lambda := \{e \in U \mid v(e) \geq \langle \lambda, s(e) \rangle\}$; note the strict inequality. Under the assumption of items being in general position, there is at most one “threshold” item with $v_i = (\lambda, S^i)$, i.e., $|U_\lambda \setminus U^\circ_\lambda| \leq 1$. Now a “good” classifier is one that satisfies the following:

**(P1’)** For all binary $x$ with support$(x) \subseteq U^\circ_\lambda$ and $x \in \mathcal{F}$, $\nabla g(Sx) \leq \lambda$.

**(P2’)** There exists a binary $x^{\text{occ}}$ with support$(x^{\text{occ}}) \subseteq U_\lambda$ and $x^{\text{occ}} \in \mathcal{F}$, and index $i^*$ such that $(\nabla g(Sx^{\text{occ}}))_{i^*} \geq \lambda_{i^*}$. (Note that if support$(x^{\text{occ}}) \subseteq U^\circ_\lambda$, then by property (P1’) the above inequality holds at equality; else $x^{\text{occ}}$ contains the unique element in $U_\lambda \setminus U^\circ_\lambda$.)

**(P3’)** This is the same as before: $g^*_i(\lambda_i) = g^*_j(\lambda_j) \quad \forall i, j \in [d]$.

The arguments of Lemma 3.4 show the following.

**Lemma 4.1.** Given Assumption [3.1] the curve $\mathcal{C}$ defined in the previous section contains a $\lambda$ satisfying properties (P1’)-(P3’).
Next, we show that for a good classifier $\lambda \in \mathcal{C}$, the maximum profit solution from $\mathcal{F}$ contained within $U_\lambda^\circ$ essentially gives an $O(1/d)$-approximation.

**Theorem 4.2** (Offline Approach). Suppose Assumption 3.7 holds. Let $\lambda^*$ be a classifier in $\mathcal{C}$ satisfying properties (P1')–(P3'). Then the better of the two solutions: (a) the maximum profit solution in $\mathcal{F}$ containing elements only from $U_{\lambda^*}^\circ$, and (b) the optimal single element in $U_{\lambda^*}$, has profit at least $\pi(x')/(2d + 1)$ for any vector $x' \in \text{Conv}(\mathcal{F}) \subseteq [0, 1]^n$.

**Proof.** The idea is to follow the development in Theorem 3.5. There same solution only from (P1')–(P3'). Then the better of the two solutions: (a) the maximum profit solution in $\mathcal{F}$ intersect $\pi$ bounds of Lemmas 3.6 and 3.8; to satisfy the first lemma, we needed the solution to be optimal for the linearization of “slope” $\lambda^*$; to satisfy the second, we needed to satisfy (P2). Here, we construct two solutions in $\mathcal{F}$ intersect $U_{\lambda^*}$ to satisfy these lemmas separately:

$$x^{\text{lin}} := \text{argmax}\{ (v, y) - (\lambda^*, Sy) \mid y \subseteq U_{\lambda^*}^\circ, y \in \mathcal{F} \}$$

$$x^{\text{occ}} := \text{the solution promised by property (P2')} \text{.}$$

Since property (P1') and (P3') holds for $x^{\text{lin}}$, Lemmas 3.6 and 3.7 hold essentially unchanged, and thus for any vector $x' \in \text{Conv}(\mathcal{F})$ we have

$$\pi(x^{\text{lin}}) \geq \pi(x') - d \cdot g^{\star}_i(\lambda^*_i). \quad (4.7)$$

The solution $x^{\text{occ}}$ may not belong to the set $U_{\lambda^*}^\circ$, since it may contain the threshold item $e^0 = (\lambda^*, s(e^0))$, if it exists (let $x^0 = \chi_{\{e^0\}}$ be its characteristic vector, all 0’s vector if does not exists). Let $x^{\text{rest}} = x^{\text{occ}} - x^0$.

**Lemma 4.3.** These solutions satisfy $\pi(x^{\text{rest}}) + \pi(x^0) \geq g^{\star}_i(\lambda^*_i)$.

**Proof.** Property (P1') gives $\nabla g(Sx^{\text{rest}}) \leq \lambda^*$, and Property (P2') implies $\nabla g(S(x^{\text{rest}} + x^0)) = \nabla g(Sx^{\text{occ}})$ is at least $\lambda^*$ at some coordinate $i^*$. Since $g$ is convex and differentiable, the gradients are continuous [HUL01], so there is $\delta \in [0, 1]$ where the vector $\tilde{x} := x^{\text{rest}} + \delta x^0$ satisfies $\nabla g(S\tilde{x}) \leq \lambda^*$ and $\nabla g(S\tilde{x})_{i^*} = \lambda^*_i$, for some coordinate $i^*$. Due to these properties, the proof of Lemma 3.8 holds for $\tilde{x}$ and shows $\pi(\tilde{x}) \geq g_i^\star(\lambda^*_i)$.

The assumption of no exceptional items gives $\pi(\delta x^0) \leq \pi(x^0)$. From subadditivity of profit $\pi, g_i^\star(\lambda^*_i) \leq \pi(\tilde{x}) \leq \pi(x^{\text{rest}}) + \pi(\delta x^0) \leq \pi(x^{\text{rest}}) + \pi(x^0)$. This concludes the proof.\hfill $\Box$

Combining Lemma 4.3 with inequality (4.7), for any $x' \in \mathcal{F}$ we have

$$\pi(x') \leq \pi(x^{\text{lin}}) + d \pi(x^{\text{rest}}) + d \pi(x^0).$$

Since $x^{\text{lin}}, x^{\text{rest}}$ are feasible for problem (a) in the theorem statement, and $x^0$ is feasible for problem (b), the best of them gives a $(2d + 1)$-approximation. This proves Theorem 4.2.\hfill $\Box$

Picking the most profitable singleton is trivial offline, and well-approximable online by the secretary algorithm [Fre83]. Moreover, we need to approximately optimize the submodular function $\pi$ (Fact 2.7) over $\mathcal{F}|_{U_{\lambda^*}^\circ}$ (i.e., the sets in $\mathcal{F}$ with only elements of $U_{\lambda^*}^\circ$). For several constraint structures (e.g., matroids, $p$-systems), there are known algorithms for approximately optimizing non-negative (and sometimes also monotone) submodular functions. Unfortunately, our profit function $\pi$ may take negative values, so we cannot directly use these algorithms. Simply considering the truncated function $\max\{\pi(z), 0\}$ does not work because it may be non-submodular. In the next section, when $g$ is separable, we introduce a way of making our profit function non-negative everywhere, while maintaining submodularity and preserving the values at the region of interest $\mathcal{F}|_{U_{\lambda^*}^\circ}$.
4.1 Making the Profit Function \( \pi \) Non-negative

We first show that \( \pi \) already satisfies the desired properties over the sets in \( \mathcal{F}|_{U_{\lambda^*}^c} \).

**Lemma 4.4.** The profit function \( \pi \) is non-negative monotone over \( \mathcal{F}|_{U_{\lambda^*}^c} \).

**Proof.** Since \( \pi(\emptyset) = 0 \) it suffices to show monotonicity. Consider \( x \in \mathcal{F}|_{U_{\lambda^*}^c} \) and let \( \chi_e \) be the indicator os an item in \( x \). Comparing the costs with and without \( e \) we have

\[
g(Sx) \leq g(S(x - \chi_e)) + \langle \nabla g(Sx), S\chi_e \rangle \leq g(S(x - \chi_e)) + \langle \lambda^*, s(e) \rangle.
\]

Since \( x \in U_{\lambda^*}^c \), we have \( v(e) > \langle \lambda^*, s(e) \rangle \) and thus \( \pi(x) > \pi(x - \chi_e) \), i.e., monotonicity. \( \square \)

However, to run algorithms that approximately optimize \( \pi \) over \( \mathcal{F}|_{U_{\lambda^*}^c} \) in a black-box fashion, non-negativity over the feasible sets \( \mathcal{F}|_{U_{\lambda^*}^c} \) is not enough, even if the algorithm only probes \( \pi \) over these sets, since their proof of correctness may require this property outside of feasible sets. Thus, we need to modify \( \pi \) to ensure non-negativity outside of \( \mathcal{F}|_{U_{\lambda^*}^c} \).

For that, the idea is to truncate the gradient of the cost \( g \) so \( \nabla g(Sx) \) becomes at most \( \lambda^* \) for all subsets \( x \subseteq U_{\lambda^*} \) (i.e., so Property (P1') holds for all subsets); this was the crucial element for the monotonicity (and hence non-negativity) proof above. Notice that since Property (P1') guarantees already \( \nabla g(Sx) \leq \lambda^* \) for all \( x \in \mathcal{F}|_{U_{\lambda^*}^c} \), this does not change the value of \( \pi \) over these points. The proof of the lemma is given in Appendix B.

**Lemma 4.5.** If \( g \) is separable, there is a submodular function \( \pi^+ \) satisfying the following:

i. \( \pi^+ \) is non-negative and monotone over all subsets of \( U_{\lambda^*} \), and

ii. \( \pi^+(x) = \pi(x) \) for every \( x \in \mathcal{F}|_{U_{\lambda^*}^c} \).

4.2 The Offline Algorithm: Wrap-up

Using this non-negativization procedure, we get an \( O(d) \)-approximation offline algorithm for constrained profit maximization for separable cost functions \( g \); this is an offline analog of Theorem 1.2. For the unconstrained case, Lemma 4.4 implies that the profit function \( \pi \) it itself monotone, so we get an \( O(d) \)-approximation offline algorithm for the supermodular case. In the next section we show how to convert these algorithms into online algorithms.

One issue we have not discussed is the computational cost of finding \( \lambda^* \) satisfying (P1')–(P3'). In the full version of the paper, we show that for any \( \varepsilon > 0 \) we can efficiently find a \( \lambda^* \) satisfying (P1'), (P2'), and a slightly weaker condition: \( |g^*_i(\lambda^*_i) - g^*_j(\lambda^*_j)| \leq 2\varepsilon \) for all \( i, j \in [d] \). Using this condition in Theorem 4.2 means we get a profit of at least \( \frac{\text{opt} - 2d\varepsilon}{2d+1} \geq [\frac{\text{opt}}{2d+1}] - \varepsilon \); the running time depends on \( \log \varepsilon^{-1} \) so we can make this loss negligible.

5 The Online Algorithm

In the previous sections we were working offline: in particular, in computing the “good” classifier \( \lambda \in \mathcal{C} \), we assumed knowledge of the entire element set. We now present the online framework for the setting where elements come in random order. Recall the definition of the curve \( \mathcal{C} \) from §3 and the fact that there is a total order among all \( \lambda \in \mathcal{C} \). Recall that for simplicity we restrict the constraints \( \mathcal{F} \) to be matroid constraints.
For a subset of elements $A \subseteq U$, let $\text{opt}(A)$ and $\text{fopt}(A)$ denote the integer and fractional optimal profit for $\mathcal{F}|_A$, the feasible solutions restricted to elements in $A$. Note that in the fractional case this means the best solution in the convex hull $\text{Conv}(\mathcal{F}|_A)$. Clearly, $\text{fopt}(A) \geq \text{opt}(A)$. We use $\text{opt}$ and $\text{fopt}$ to denote $\text{opt}(U)$ and $\text{fopt}(U)$ for the entire instance $U$.

Again we work under Assumption 3.1. We will also make use of any algorithm for maximizing submodular functions over $\mathcal{F}$ in the random-order model satisfying the following.

**Assumption 5.1.** Algorithm SubmodMS takes a nonnegative monotone submodular function $f$ with $f(\emptyset) = 0$, and a number $N$. When run on a sequence $X$ of $N$ elements presented in random order, it returns a (random) subset $X_{\text{alg}} \in \mathcal{F}$ with expected value $\mathbb{E}[f(X_{\text{alg}})] \geq \frac{1}{\alpha} \max_{X' \in \mathcal{F}} f(X)$. Moreover, it only evaluates the function $f$ on feasible sets.

Our algorithm is very simple:

**Algorithm 5.1 Online Algorithm for Profit Maximization**

1. $L \leftarrow$ first Binomial$(n, 1/2)$ items.
2. $\mu \leftarrow$ largest vector on curve $C$ s.t. $\text{fopt}(L_\mu) \geq \frac{1}{12d} \text{fopt}(L)$.
3. $R \leftarrow$ remaining instance, namely the last $n - |L|$ items.
4. $R_{\mu}^o \leftarrow \{ e \in R \mid v(e) > (\mu, s(e)) \}$ be the (strictly) “filtered” remaining instance.
5. **Un-constrained:** Select items in $R_{\mu}^o$ not decreasing the current value of the solution.

**Constrained:** Run algorithm SubmodMS on $R_{\mu}^o$ using the profit function $\pi$, selecting items according to this algorithm. However, do not add any items that decrease the current value of the solution.

Note that $L_\mu$ denotes the set of items in the sample $L$ picked by $\mu$ (Definition 3.2). In Step 2, we can use the Ellipsoid method to find $\text{fopt}$, i.e., to maximize the concave profit function $\pi$ over the matroid polytopes $\text{Conv}(\mathcal{F}|_{L_\mu})$ and $\text{Conv}(\mathcal{F}|_{L})$, within negligible error. Moreover, we must do this for several sets $L_\mu$ and pick the largest one on $C$ using a binary-search procedure. We defer the technical details to the full version of the paper.

### 5.1 Analysis

To analyze the algorithm, we need to show that the classifier $\mu$ learned in Step 2 is large enough that we do not waste space with useless items, but low enough that we admit enough useful items. Along the way we frequently use the concentration bound from Fact A.9. For this we need the profit function $\pi$ to satisfy a Lipschitz-type condition (A.10) on the optimal solutions of any given sub-instance. To facilitate this, let us record a useful lemma, proved in Appendix B. For a vector $y \in \mathbb{R}^n$, and a subset $A \subseteq U$, define $y_A$ to be the same as $y$ on $A$, and zero outside $A$.

**Claim 5.2.** Consider any $U' \subseteq U$, and let $y$ be an optimal fractional solution on $\mathcal{F}|_{U'}$ (so $\pi(y) = \text{fopt}(U')$). Then for any $B \subseteq A \subseteq U'$ with $|A \setminus B| = 1$, we have $|\pi(y_A) - \pi(y_B)| \leq M$, where $M$ is an upper bound on the profit from any single item.

From Section 4, recall $\lambda^* \in \mathbb{R}^d_+$ is a classifier that satisfies properties (P1')–(P3').

**Lemma 5.3** (Goldilocks Lemma). Given Assumption 3.1, the classifier $\mu$ computed in Line 2 of Algorithm 5.1 satisfies:

- (a) (Not too small) $\mu \geq \lambda^*$, with probability at least $19/20$.
- (b) (Not too big) $\text{fopt}(U_\mu) \geq \frac{\text{fopt}}{48d}$ with probability at least $1 - 1/20d \geq 19/20$. 

10
Proof sketch. (See Appendix [B] for full proof.) For the first part, we show that the classifier $\lambda^*$ satisfies the properties needed in Line 2 of Algorithm 4.1 with probability $1 - 1/20$; since $\mu$ is the largest such vector, we get $\mu \geq \lambda^*$. Using Theorem 4.2 and the assumption that no item has large profit, we have $\text{fopt}(U_{\lambda^*}) \geq \text{fopt}(M_{\lambda^*})$. Moreover, the sample $\lambda$ obtains at least half of this profit in expectation, i.e., $\mathbb{E}[\text{fopt}(L_{\lambda})] \geq \text{fopt}(M_{\lambda^*}) \cdot \delta$. Then using concentration (Fact A.9) with the Lipschitz property of Claim 5.2 and the no-high-profit-item assumption, we have $\text{fopt}(L_{\lambda^*}) \geq \text{fopt}(L_{\lambda})$ (which is at least $\text{fopt}(L_2)$) with probability at least $19/20$. Thus, with this probability $\lambda^*$ satisfies the properties needed in Line 2 of the algorithm, as desired.

For the part (b) of the lemma, notice that for each scenario $\text{fopt}(U_{\mu}) \geq \text{fopt}(L_{\mu})$, since feasible solutions for the sample are feasible for the whole instance. Next, by definition of $\mu$, $\text{fopt}(L_{\mu}) \geq \text{fopt}(L_\lambda)$. Finally, if $x$ is the fractional optimal solution on $U$ with $\pi(x) = \mu$, then $\mathbb{E}[\pi(x_L)] \geq \text{fopt}(L_\lambda)/2$, since $g$ is super-additive. Again using the concentration bound Fact A.9, the profit $\pi(x_L)$ is at least $\text{fopt}(L_{\lambda^*})$ with probability at least $(1 - 1/20d)$. Of course, $\text{fopt}(L_\lambda) \geq \pi(x_L)$. Chaining these inequalities, $\text{fopt}(U_{\mu}) \geq \text{fopt}(L_{\lambda^*})/12d$ with this probability. \hfill $\square$

In view of Theorem 4.2, we show the filtered out-of-sample instance $R_{\mu}^\circ$ behaves like $U_{\lambda^*}$.

**Lemma 5.4.** The filtered out-of-sample instance $R_{\mu}^\circ$ satisfies the following w.p. $19/20$:

(a) For all $e \in R_{\mu}^\circ$, $v(e) \geq \langle \lambda^*, s(e) \rangle$.

(b) For all $x$ with support $x_{\mathcal{F}} \subseteq R_{\mu}^\circ$ such that $x \in \mathcal{F}$, $\nabla g(Sx) \leq \lambda^*$.

(c) $\text{fopt}(R_{\mu}^\circ) \geq \text{fopt}(L_{\lambda^*})/12d$.

**Proof.** By Lemma 5.3(a), threshold $\mu \geq \lambda^*$ with probability $19/20$. When that happens, $U_{\mu}^\circ \subseteq U_{\lambda^*}^\circ$. Since the first two properties hold for $U_{\lambda^*}^\circ$, they also hold for $U_{\mu}^\circ$, and by downward-closedness, also for $R_{\mu}^\circ$.

For the third part, let $\lambda^+$ be the largest threshold in $\mathcal{C}$ such that $\text{fopt}(U_{\lambda^+}) \geq \text{fopt}(L_{\lambda^+})/200d$. From Lemma 5.3(b), with good probability we have $\mu \leq \lambda^+$. Since $\mu$ is a smaller threshold, the instance $U_{\lambda^+}$ is contained in the instance $U_{\mu}$, which implies that for every scenario $\text{fopt}(R_{\mu}^\circ) \geq \text{fopt}(R_{\lambda^+})$. Next we will show that with good probability $\text{fopt}(R_{\lambda^+}) \geq \text{fopt}(L_{\lambda^+})/12d$, and hence get the same lower bound for $\text{fopt}(R_{\mu})$. If $y$ is the optimal fractional solution for $U_{\lambda^+}$, then $y_R$ is feasible for $R_{\lambda^+}$ with $\mathbb{E}[\pi(y_R)] = \frac{1}{2} \text{fopt}(U_{\lambda^+}) \geq \text{fopt}(L_{\lambda^+})/12d$. Moreover, using the concentration bound again, we get that $\pi(y_R) \geq \text{fopt}(L_{\lambda^+})/12d$ with probability at least $19/20$. Finally, by the assumption of general position, there is at most one item in $R_{\mu} \setminus R_{\mu}^\circ$. Dropping this item from the solution $y$ to get $y^\circ$ reduces the value by at most $M = \text{fopt}(L_{\lambda^+})/100d$; here we use subadditivity of the profit, and that there are no exceptional items. Hence, with probability at least $19/20$:

$$\text{fopt}(R_{\mu}^\circ) \geq \text{fopt}(R_{\lambda^+}^\circ) \geq \pi(y_R^\circ) \geq \frac{\text{fopt}(L_{\lambda^+})}{196d} - M \geq \frac{\text{fopt}(L_{\lambda^*})}{200d}.$$

Finally, we are ready to prove the main theorems in the online setting.

**Theorem 5.5 (Unconstrained Case: Supermodular Cost Functions).** Algorithm 5.1 gives an $O(d)$-approximation in expectation for the unconstrained case, if the cost function is supermodular.

**Proof.** Define the event $\mathcal{E}$ that Lemmas 5.3 and 5.4 hold; $\text{Pr}(\mathcal{E}) \geq 17/20$. Now, by Lemma 5.4(c), the optimal fractional solution for $R_{\mu}^\circ$ has profit at least $\text{fopt}(L_{\mu})/200d$. Moreover, since there are no constraints, the profit function is monotone submodular over all of $U_{\lambda^*}^\circ$, by Lemma 4.4. Conditioning on the good event $\mathcal{E}$, Lemma 5.3(a) gives that $R_{\mu} \subseteq U_{\lambda^*}^\circ$, so the algorithm to maximize the monotone submodular function (both integrally and fractionally) is to pick all elements. Hence, conditioned on $\mathcal{E}$, the profit we get is at least $\frac{\text{fopt}(L_{\mu})}{200d}$. In the other case, we never pick an item that gives negative marginal value, so our solution is always non-negative. Hence our expected profit is at least $\text{Pr}(\mathcal{E}) \cdot \text{opt}(R_{\mu}) = \Omega(\text{fopt}(d)) \geq \Omega(\text{opt}(d))$. \hfill $\square$
The analysis of the algorithm for the constrained separable-cost case is similar, only using the constrained offline guarantees of Theorem 4.2 and the non-negativization Lemma 4.4 to argue that SubmodMS maintains its guarantees. Details are provided in Appendix B.

**Theorem 5.6 (Constrained Case: Separable Cost Functions).** Suppose algorithm SubmodMS satisfies Assumption 5.1 and is $\alpha$-competitive in expectation. Then Algorithm 5.1 gives a $O(\alpha d^2)$-approximation in expectation.

### 6 Separability versus Supermodularity

In this section, we show that an $\beta$-approximation algorithm for the separable-cost case gives a $O(\beta_+ 2e d)$-approximation for a slight generalization of the supermodular-cost case. Consider the problem of picking a set $A$ to solve

$$
\pi(A) := \max_{A \in \mathcal{F}} \left( v(A) - g(\sum_{e \in A} s(e)) \right),
$$

where $v(A)$ is a (discrete) submodular function over $\{0,1\}^n$ with $v(\emptyset) = 0$, $g$ is a convex, (continuous) supermodular function over $\mathbb{R}^d$, and $\mathcal{F}$ is some downward-closed constraint set. We show that for the case of matroid constraints, this problem can be reduced to the setting where the cost function is separable over its $d$ coordinates, suffering a loss of $O(d)$.

**Theorem 6.1 (Reduction).** Given an $\beta$-approximation algorithm for profit-maximization for separable convex cost functions under matroid constraints, we can get an $d(\beta + 2ed)$-approximation algorithm for the profit-maximization problem with supermodular costs $g$, submodular values $v$, and $\mathcal{F}$ being a matroid constraint.

The reduction is the following:

1. Define separable costs $\overline{g}(y) := 1/d \sum_{i=1}^d g_i(dy_i)$, where $g_i$ are marginal functions for $g$.
2. W.p. $p = \frac{\beta}{\beta + 2ed}$, run single-secretary algorithm to return element with maximum profit.
3. W.p. $1 - p = \frac{2ed}{\beta + 6d}$, run algorithm for value function $v(\cdot)$ and separable cost fn. $\overline{g}(\cdot)$.

This reduction relies on the following simple but perhaps surprising observation that relates separability with supermodularity, which may find other applications.

**Lemma 6.2.** Given a monotone convex superadditive function $g$ with $g(0) = 0$, let $g_i$ be the marginal functions. Then for all $y \in \mathbb{R}^d_+$:

1. $g(y) \geq \sum_i g_i(y_i)$
2. $g(y) \leq \frac{1}{d} \sum_i g_i(dy_i) = \overline{g}(y)$.

**Proof.** The first property follows from the superadditivity of $g$, and the second follows from Jensen’s inequality. 

While the full proof of Theorem 6.1 is deferred to Appendix B, the main idea is clean. Given an optimal integer solution $x^*$ for the original problem (with the original cost function), we use Lemma 6.2 and the Lovász (convex) extension of submodular functions to show that $x^*/d$ is a good fractional solution for the separable cost function. Now using polyhedral properties of $d$-dimensional faces of the matroid polytope, and other properties of the Lovász extension, we show the existence of a good integer solution to the separable problem. Combining this reduction with Theorem 1.2 proves Theorem 1.3.
References

[ABC+16] Yossi Azar, Niv Buchbinder, T-H. Hubert Chan, Shahar Chen, Ilan R. Cohen, Anupam Gupta, Zhiyi Huang, Ning Kang, Viswanath Nagarajan, Joseph (Seffi) Naor, and Debmalya Panigrahi, Online algorithms for covering and packing problems with convex objectives, 57th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2016, New Brunswick, NJ, USA, October 9-11, 2016, 2016.

[AD15] Shipra Agrawal and Nikhil R. Devanur, Fast algorithms for online stochastic convex programming, Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, PA, 2015, pp. 1405–1424. MR 3451117

[AWY14] Shipra Agrawal, Zizhuo Wang, and Yinyu Ye, A dynamic near-optimal algorithm for online linear programming, Oper. Res. 62 (2014), no. 4, 876–890. MR 3246084

[BGMS11] Avrim Blum, Anupam Gupta, Yishay Mansour, and Ankit Sharma, Welfare and profit maximization with production costs, FOCS, Nov 2011, pp. 77–86.

[BIKK07] Moshe Babaioff, Nicole Immorlica, David Kempe, and Robert Kleinberg, A knapsack secretary problem with applications, APPROX-RANDOM, 2007, pp. 16–28.

[BLM13] S. Boucheron, G. Lugosi, and P. Massart, Concentration inequalities: A nonasymptotic theory of independence, OUP Oxford, 2013.

[BUCM12] Siddharth Barman, Seeun Umboh, Shuchi Chawla, and David Malec, Secretary problems with convex costs, Automata, languages, and programming. Part I, Lecture Notes in Comput. Sci., vol. 7391, Springer, Heidelberg, 2012, pp. 75–87. MR 2995298

[DGV05] Brian C. Dean, Michel X. Goemans, and Jan Vondrak, Adaptivity and approximation for stochastic packing problems, Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2005, Vancouver, British Columbia, Canada, January 23-25, 2005, 2005, pp. 395–404.

[DJSW11] Nikhil R. Devanur, Kamal Jain, Balasubramanian Sivan, and Christopher A. Wilkens, Near optimal online algorithms and fast approximation algorithms for resource allocation problems, ACM Conference on Electronic Commerce (Yoav Shoham, Yan Chen, and Tim Roughgarden, eds.), ACM, 2011, pp. 29–38.

[EF16] Reza Eghbali and Maryam Fazel, Designing smoothing functions for improved worst-case competitive ratio in online optimization, Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain, 2016, pp. 3279–3287.

[ESF14] Reza Eghbali, Jon Swenson, and Maryam Fazel, Exponentiated subgradient algorithm for online optimization under the random permutation model, CoRR abs/1410.7171 (2014).

[Fre83] P. R. Freeman, The secretary problem and its extensions: a review, Internat. Statist. Rev. 51 (1983), no. 2, 189–206. MR 715534

[FSZ15] Moran Feldman, Ola Svensson, and Rico Zenklusen, A simple $O(\log \log (\text{rank}))$-competitive algorithm for the matroid secretary problem, Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, PA, 2015, pp. 1189–1201. MR 3451103

[FZ15] Moran Feldman and Rico Zenklusen, The submodular secretary problem goes linear, 2015 IEEE 56th Annual Symposium on Foundations of Computer Science—FOCS 2015, IEEE Computer Soc., Los Alamitos, CA, 2015, pp. 486–505. MR 3473324

[GM16] Anupam Gupta and Marco Molinaro, How the experts algorithm can help solve lps online, Math. Oper. Res. 41 (2016), no. 4, 1404–1431.

[GRSZ14] Fabrizio Grandoni, R. Ravi, Mohit Singh, and Rico Zenklusen, New approaches to multi-objective optimization, Math. Program. 146 (2014), no. 1-2, 525–554.

[HK15] Zhiyi Huang and Anthony Kim, Welfare maximization with production costs: a primal dual approach, 26th SODA, SIAM, Philadelphia, PA, 2015, pp. 59–72. MR 3451030
A Facts about Convex Functions and Useful Inequalities

A.1 Convex Functions

To avoid degenerate conditions, we assume that the convex cost functions $g$ we consider are closed, not identically $+\infty$ and there is an affine function minorizing $g$ on $\mathbb{R}^d$.

**Definition A.1** (Convex Dual). For any function $g : \mathbb{R}^d \to \mathbb{R}$, its convex dual is the function $g^* : \mathbb{R}^d \to \mathbb{R}$ given by

$$g^*(y) := \sup_x \left[ \langle y, x \rangle - g(x) \right].$$

Figure 1: The overshoot gap is upper bounded by $g^*(\lambda)$.
Claim A.2 (Linearization). \((HUL01\text{,} \text{Theorem E.1.4.1})\) For every convex function \(g : \mathbb{R}^d \to \mathbb{R}\), any (sub)gradient at the point \(x\) gives the “right linearization”:

\[
g(x) = \langle x, u \rangle - g^*(u) \iff u \in \partial g(x).
\]  
(A.8)

Claim A.3 (Double Dual). \((HUL01\text{,} \text{Corollary E.1.3.6})\) Let \(g : \mathbb{R}^d \to \mathbb{R}\) be a convex function. If its epigraph \(\{(x, r) \in \mathbb{R}^d \times \mathbb{R} : r \geq g(x)\}\) is closed, then \(g^{**} = g\).

Claim A.4 (Fenchel-Young Inequality). For every convex function \(g : \mathbb{R}^d \to \mathbb{R}\), linearizing using any vector gives us an underestimate on the value of \(g\):

\[
g(x) \geq \langle x, u \rangle - g^*(u).
\]  
(A.9)

Claim A.5 (Dual Function Facts). Let \(g : \mathbb{R}^d_+ \to \mathbb{R}\) be a convex function on the positive orthant.

\(a.\) If \(g(0) = 0\), then \(g^*(\lambda) \geq 0\) for all \(\lambda\).

\(b.\) If \(\lambda \preceq \lambda'\) then \(g^*(\lambda) \geq g^*(\lambda')\).

\(c.\) \(g^*\) is a closed convex function.

Proof. For property \((a)\), \(g^*(\lambda) \geq \langle \lambda, 0 \rangle - g(0) = 0\) using Claim A.4.

For property \((b)\), take any \(x\) in \(\mathbb{R}^d_+\) (the domain of \(g\)) and observe

\[
\langle \lambda, x \rangle - g(x) \geq \langle \lambda', x \rangle - g(x).
\]

Take the supremum over all such \(x\)'s in the left-hand side and use Definition A.1 to get

\[
g^*(\lambda) \geq \langle \lambda', x \rangle - g(x)
\]

for all \(x \in \mathbb{R}^d_+\). To complete the argument, take the supremum on the right-hand side.

For property \((c)\), see \([HUL01, \text{Theorem E.1.1.2}]\). \(\square\)

Claim A.6 (Duals and Marginals Commute). Given a monotone convex \(g : \mathbb{R}^d \to \mathbb{R}\), \((g^*)_i(z) = (g_i)^*(z)\) for all \(z \in \mathbb{R}\). I.e., the marginal of the dual is the same as the dual of the marginal.

Proof. \((g^*)_i(z) = g^*(ze_i) = \max_x (\langle ze_i, x \rangle - g(x)) \overset{\text{increas.}}{=} \max_{x_i} (zx_i - g_i(x_i)) = \max_{x_i} (zx_i - g_i(x_i)) = (g_i)^*(z)\). This means there are no concerns of ambiguity when we write \(g^*_i(z)\). \(\square\)

Claim A.7 (Subadditivity over Coordinates). Given a superadditive convex function \(g : \mathbb{R}^d \to \mathbb{R}\),

\[
g^*(\lambda) \leq \sum_i g^*_i(\lambda_i) \quad \forall \lambda.
\]

Proof. From the definition of convex dual, we have

\[
g^*(\lambda) = \max_x (\langle x, \lambda \rangle - g(x)) \leq \max_x (\langle x, \lambda \rangle - \sum_i g_i(x_i)) = \sum_i \max_{x_i} (x_i \lambda_i - g_i(x_i)) = \sum_i g^*_i(\lambda_i).
\]

The inequality uses the superadditivity of \(g\). \(\square\)
A.2 Probabilistic inequalities

Fact A.8. Consider a vector \( x \in \{0, 1\}^n \) and let \( X \) be the random vector obtained by setting each coordinate of \( x \) to 0 with probability \( \frac{1}{2} \). If cost function \( g \) is supermodular then
\[
\mathbb{E}[\pi(X)] \geq \frac{1}{2} \pi(x).
\]

Proof. Function \( g \) is superadditive due to Lemma 2.5, and so \( \pi \) is subadditive: \( \pi(y + z) \leq \pi(y) + \pi(z) \). Writing \( x = X + (x - X) \) and applying subadditivity, we get
\[
\pi(x) \leq \pi(X) + \pi(x - X).
\]
But \( X \) and \( x - X \) have the same distribution, so taking expectations gives
\[
2\mathbb{E}[\pi(X)] \geq \pi(x).
\]

Fact A.9. Consider a submodular function \( f : 2^U \rightarrow \mathbb{R} \). Consider a set \( Y \subseteq U \) such that \( f \) is non-negative over all of its subsets and we also have the following Lipschitz condition for some \( M \):
\[
\text{For all } Y' \subseteq Y \text{ and element } e \in Y', \ |f(Y') - f(Y' - e)| \leq M. \tag{A.10}
\]

Let \( Y \) be the random subset obtained from picking each element from \( Y \) independently with some probability (which can be different for each item). Then
\[
\text{Pr}(|f(Y) - \mathbb{E}[f(Y)]| \geq t) \leq \frac{2M \mathbb{E}[f(Y)]}{t^2} \tag{A.11}
\]

Proof. Vondrák showed that \( M \)-Lipschitz non-negative submodular functions are weakly \((2M,0)\)-self-bounding \[Von10\]. By the Efron-Stein inequality, such functions have \( \text{Var}(f(Y)) \leq 2M \mathbb{E}[f(Y)] \) \[BLM13\]. Now Chebychev’s inequality gives the result.

B Missing Proofs

B.1 Proofs for Section 2.1

Proof of Lemma 2.5. By integrating gradients we have
\[
f(x' + y) - f(x') = \int_{t=0}^{1} \langle \nabla f(x' + ty), y \rangle \, dt \leq \int_{t=0}^{1} \langle \nabla f(x + ty), y \rangle \, dt = f(x + y) - f(x),
\]
where the inequality uses Lemma 2.4(1), the monotone gradients property.

Proof of Fact 2.7. Given \( x' \leq x \) in the Boolean cube and \( i \) not in \( x 
\[
h(x' + e_i) - h(x') = g(S(x' + e_i)) - g(Sx') \leq g(S(x + e_i)) - g(Sx) = h(x + e_i) - h(x),
\]
where the inequality uses Lemma 2.5 and \( S \geq 0 \).
B.2 Proof of Lemma 4.5

Since $g$ is separable, it has the form $g(z) = \sum_i g_i(z_i)$. Thus, it will suffice to perform the gradient truncation on each of the 1-dimensional functions $g_i$, which is accomplished by the following lemma.

Claim B.1 (Gradient truncation). Given a 1-dimensional convex function $f : \mathbb{R} \to \mathbb{R}$ and a real value $\gamma \in \mathbb{R}_+$, there is another function $f^+ : \mathbb{R} \to \mathbb{R}$ satisfying the following:

i. $f^+$ is convex.
ii. For all $z$, all subgradients $u \in \partial f^+(z)$, and satisfy $u \leq \gamma$.
iii. If $z$ is such that some subgradient $z \in \partial f(z)$ satisfies $u \leq \gamma$, then $f^+(z) = f(z)$.

Proof. Define $f^+$ as follows:

$$f^+(z) = \sup_{u \leq \gamma} [u \cdot z - f^*(u)];$$

notice the constraint $u \leq \gamma$, and that the dual $f^*$ is the usual Fenchel dual for $f$. Properties (i) and (ii) follow from the fact $f^+$ is the point-wise supremum of linear functions with slopes at most $\gamma$.

For Property (iii), by the double dual property in Claim A.3, $f(z) = \sup_u [u \cdot z - f^*(u)]$, and hence $f \geq f^+$. Moreover, take a point $z$ such that some subgradient $u \in \partial f(z)$ satisfies $u \leq \gamma$. Then we have

$$f(z) \overset{(\text{Claim A.3})}{=} \pi \cdot z - f^*(u) \leq \sup_{u \leq \lambda^*} [u \cdot z - f^*(u)] \overset{\text{(by defn.)}}{=} f^+(z) \leq f(z).$$

This concludes the proof. \hfill \Box

Now define $g_i^+$ as the function obtained by applying the truncation lemma above with $f = g_i$ and $\gamma = \lambda_i^*$. Also define the truncated cost function $g^+$ as $g^+(z) = \sum_i g_i^+(z_i)$, and the truncated profit function $\pi^+$ as $\pi^+(x) = \langle v, x \rangle - g^+(Sx)$.

We claim that $\pi^+$ satisfies all properties required by the lemma. First, it is discrete submodular: $g^+$ is convex and supermodular, since it is the sum of 1-dimensional convex functions, which are trivially supermodular; discrete submodularity of $\pi^+$ then follows from Fact 2.7.

Moreover, we claim $\pi^+$ has the same value as $\pi$ over solutions in $\mathcal{F}_{U_{\lambda^*}^o}$. This follows from the fact that every solution $x$ in this family has $\nabla g(Sx) \leq \lambda^*$ (by Property (P1')), and hence Claim B.1(iii) guarantees that $g(Sx) = g^+(Sx)$.

Finally, we claim that $\pi^+$ is non-negative monotone. Since $\pi^+(\emptyset) = \pi(\emptyset) = 0$, it again suffices to show monotonicity. For that, Claim B.1(ii) guarantees that all solutions $x \subseteq U_{\lambda}^o$ now satisfy $\nabla g^+(Sx) \leq \lambda^*$.

B.3 Proof of Claim 5.2

Say $A \mid B = \{i\}$. Then by supermodularity of the cost function $g$ (Lemma 2.5) and the absence of exceptional items,

$$\pi(y_A) - \pi(y_B) = v_iy_i - (g(Sy_A) - g(Sy_B)) \leq v_iy_i - g(S^i y_i) \leq \pi(e_i) \leq M.$$ 

For the lower bound, we also use the optimality of $y$.

$$\pi(y_A) - \pi(y_B) = v_iy_i - (g(Sy_A) - g(Sy_B)) \geq v_iy_i - (g(Sy_U) - g(Sy_{U \setminus \{i\}})) = \pi(y) - \pi(y_{U \setminus \{i\}}) \geq 0,$$

where the middle inequality uses supermodularity of $g$, and the last one uses the optimality of $y$. This concludes the proof.
B.4 Proof of Lemma 5.3

Part (b) was already proved in details, so we provide only a proof for part (a).

Let \( f_{\text{opt}}' = f_{\text{opt}}(U_\lambda) \). Using Theorem 4.2 and the assumption that no item has profit more than \( \text{opt}/\eta d \), we know \( f_{\text{opt}}' \geq \text{opt}(U_\lambda^d) \geq \frac{f_{\text{opt}}}{2d+1} \geq \frac{f_{\text{opt}}}{3d} \). Let \( y \) be an optimal fractional solution for \( U_\lambda^* \), so that \( \pi(y) = f_{\text{opt}}' \geq \frac{f_{\text{opt}}}{3d} \). By downward-closedness of \( \mathcal{F} \), \( y_L \) is a feasible fractional solution only containing items in \( L_\lambda^* \). Therefore, \( \mathbb{E}[f_{\text{opt}}(L_\lambda^*)] \geq \mathbb{E}[\pi(y_L)] = f_{\text{opt}}'/2 \). Finally, using the concentration inequality of Fact A.9 with \( t = f_{\text{opt}}/12d \) (where the Lipschitz-type condition is satisfied due to Claim 5.2), we get

\[
\Pr \left( \left| \pi(y_L) - \frac{f_{\text{opt}}}{2} \right| \leq \frac{f_{\text{opt}}}{12d} \right) \geq 1 - \frac{2M (fopt'/2)}{(fopt/12d)^2} \geq \frac{19}{20}.
\]

The last inequality follows using \( M = \frac{\text{opt}}{\eta d} \leq \frac{\text{opt}}{10^4d} \). Hence, w.p. at least \( 19/20 \), \( f_{\text{opt}}(L_\lambda^*) \geq \pi(y_L) \geq \frac{f_{\text{opt}}}{2} - \frac{f_{\text{opt}}}{12d} \geq \frac{f_{\text{opt}}}{12d} \). This is at least \( \frac{1}{12d} f_{\text{opt}}(L) \) since \( L \subset U \). Hence \( \lambda^* \) is a candidate in the definition of \( \mu \), thus by maximality \( \mu \geq \lambda^* \), proving the part (a) of the lemma.

B.5 Proof of Theorem 5.6

Recall the definition of modified profit function \( \pi^+ \) from Lemma 4.5. Again, let \( \mathcal{E} \) be the event that Lemmas 5.3 and 5.4 hold; by a union bound \( \Pr[\mathcal{E}] \geq 17/20 \). We claim that under this event, the functions \( \pi^+ \) and \( \pi \) coincide on feasible sets of \( R^*_\mu \). Indeed, \( \mu \geq \lambda^* \), so the elements \( R^*_\mu \subset U_\lambda^* \), and then Lemma 4.5(iii) tells us that \( \pi^+ \) and \( \pi \) agree on all feasible subsets of \( U_\lambda^* \).

Using Assumption 5.1, the algorithm \texttt{SubmodMS} applied over \( R^*_\mu \) cannot distinguish between \( \pi \) and \( \pi^+ \). Thus, the solution \( X \in \mathcal{F} \) returned by our algorithm using profit function \( \pi \) is the same as one returned by running algorithm \texttt{SubmodMS} over instance \( R^*_\mu \) with the \textit{non-negative monotone submodular} function \( \pi^+ \). This algorithm is \( \alpha \)-competitive, and moreover conditioning on the sample still leaves the out-of-sample items in random order, so the guarantee of the algorithm still holds. Hence,

\[
\mathbb{E}[\pi(X) \mid \mathcal{E}] = \mathbb{E}[\pi^+(X) \mid \mathcal{E}] \overset{\alpha\text{-approx}}{=} \frac{1}{\alpha} \mathbb{E}[\text{opt}(R^*_\mu) \mid \mathcal{E}],
\]

where the first equality follows from \( \pi^+ \) and \( \pi \) agreeing on \( R^*_\mu \). Since we are assuming no item has large profit, by Theorem 4.2 the integer optimum is at least a \( 1/(2d+1) \)-fraction of the fractional optimum,

\[
\frac{1}{\alpha} \mathbb{E}[\text{opt}(R^*_\mu) \mid \mathcal{E}] \geq \frac{1}{\alpha(2d+1)} \mathbb{E}[f_{\text{opt}}(R^*_\mu) \mid \mathcal{E}] \geq \frac{f_{\text{opt}}}{200\alpha d(2d+1)},
\]

the last inequality using that event \( \mathcal{E} \) guarantees Lemma 5.4(c). Since the algorithm does not include items with negative marginals, it always produces solutions with non-negative values. Therefore,

\[
\mathbb{E}[\pi(X)] \geq \mathbb{E}[\pi(X) \mid \mathcal{E}] \Pr(\mathcal{E}) \geq \frac{\text{opt}}{O(\alpha d^2)}.
\]

The last inequality follows since \( \Pr(\mathcal{E}) \) is a constant. This concludes the proof.

B.6 Proof of Theorem 6.1

Let \( x^* \) be the optimal solution with value \( \text{opt} := \pi(x^*) \) for the problem of maximizing profit with the supermodular cost function \( g \). Since our proof deals with fractional allocations, define \( \hat{v}(\cdot) \) to be the convex extension (or
Lovász extension) of the submodular value function \(v(\cdot)\) \[Sch03\]. Since \(\tilde{v}\) is an extension of \(v\), \(v(x) = \tilde{v}(x)\) for all points \(x\) in the domain of \(v\), i.e., for \(x \in \{0, 1\}^n\). Define for all \(x \in [0, 1]^n\),

\[
\pi(x) := \tilde{v}(x) - \bar{g}(Sx).
\]

Now,

\[
\pi(x^*/d) = \tilde{v}(x^*/d) - \bar{g}(Sx^*/d) \geq 1/d \left( v(x^*) - g(Sx^*) \right) = \pi(x^*)/d = \text{opt}/d. \tag{B.12}
\]

The inequality uses that for the fractional point \(x^*/d\), the Lovász extension value is \(\tilde{v}(x^*/d) = (1 - 1/d) \cdot v(0) + 1/d \cdot v(x^*)\), and that by Lemma 6.2(1), \(\bar{g}(Sx^*/d) \leq g(Sx^*)\).

So the separable problem has a good fractional solution \(x^*/d\), and we want to “round” it to a near-integral solution. Indeed, take the matroid polytope \(P\) corresponding to the matroid constraint \(\mathcal{F}\), and intersect \(P\) with the subspace \(\{x \mid Sx = S(x^*/d)\}\). Clearly \(x^*/d\) belongs to this intersection. Now consider maximizing the linear function \(\langle \nabla \tilde{v}(x^*/d), x - x^*/d \rangle\) over this polytope, and let \(\tilde{x}\) be a basic feasible solution to this linear optimization problem. Since at most \(d\) of the tight constraints come from the subspace restriction, the point \(\tilde{x}\) lies on some face of the matroid polytope of dimension at most \(d\). By \[GRSZ14, Theorem 4.3\], \(\tilde{x}\) has at most \(2d\) fractional coordinates. Moreover, since \(\tilde{x}\) is the maximizer of the linear function and \(x^*/d\) is a feasible point, the inner product \(\langle \nabla \tilde{v}(x^*/d), \tilde{x} - x^*/d \rangle \geq 0\). The convexity of the Lovász extension now implies \(\tilde{v}(\tilde{x}) \geq \tilde{v}(x^*/d)\). Because \(S\tilde{x} = Sx^*/d\), the cost remains unchanged and we get

\[
\pi(\tilde{x}) \geq \pi(x^*/d) \geq \text{opt}/d.
\]

Let \(x^\text{int}\) be the \(\tilde{x}\) restricted to the integral coordinates, and let \(F \subseteq [n]\) be the set of fractional coordinates in \(\tilde{x}\). Then by subadditivity of the \(\pi\) function, we get the following, where \(\chi_e\) is an indicator vector of element \(e\).

\[
\text{opt}/d \leq \pi(\tilde{x}) \leq \pi(x^\text{int}) + \sum_{e \in F} \pi(\tilde{x}_e \chi_e) \leq \pi(x^\text{int}) + \sum_{e \in F} \left( \tilde{v}(\tilde{x}_e \chi_e) - g(\tilde{x}_e S\chi_e) \right). \tag{Definition of \(\pi\), and Lemma 6.2(2)}
\]

Moreover, for each individual item \(e\),

\[
\tilde{v}(\tilde{x}_e \chi_e) - g(\tilde{x}_e S\chi_e) = \tilde{x}_e \tilde{v}(\chi_e) - g(\tilde{x}_e S\chi_e) \leq \tilde{v}(\chi_e) - g(S\chi_e) = \pi(\chi_e).
\]

The first equality above uses that the Lovász extension acts linearly on single items. The inequality follows since there are no exceptional items and \(\tilde{x}_e \in (0, 1)\). Hence, we get \(\pi(x^\text{int}) + \sum_{e \in F} \pi(\chi_e) \geq \text{opt}/d\).

We can use the algorithm for the separable problem (which is part of the theorem assumption) to find \(x^\text{sep}\) with value \(\pi(x^\text{sep}) \geq (1/\beta) \pi(x^\text{int})\). Using Lemma 6.2(2) again, \(\pi(x^\text{sep}) \geq \pi(x^\text{sep})\). Also, using the well-known 1/\(e\)-approximation for the most profitable item returns an item \(e^*\) with profit \(\pi(e^*) \geq 1/e^* \sum_{e \in F} \pi(\chi_e)\). Returning \(x^\text{sep}\) with probability \(\frac{1}{\beta + 2ed}\) and the single element \(\tilde{e}\) otherwise gives expected value at least

\[
\frac{1}{\beta + 2ed} \left( \beta \pi(x^\text{sep}) + 2ed \pi(x^\text{opt}) \right) \geq \frac{1}{\beta + 2ed} \left( \pi(x^\text{int}) + \sum_{e \in F} \pi(\chi_e) \right) \geq \frac{\text{opt}}{d(\beta + 2ed)}.
\]

### C Other Loose Ends

#### C.1 Conjugates over the Positive Orthant

The following lemma justifies why it is enough to consider only non-negative \(\lambda_s\) for our setting.
Lemma C.1. Given a convex, non-decreasing, non-negative function $g : \mathbb{R}^d_+ \to \mathbb{R}_+$, $\forall z \in \mathbb{R}^d_+$, we have $g(z) = \max_{\lambda \in \mathbb{R}^d_+} \langle \lambda, z \rangle = \max_{\lambda \in \mathbb{R}^d_+} \langle \lambda, z \rangle - g^*(\lambda)$.

Proof. Let $\tilde{g}$ be a function that is same as $g$ on positive orthant and is $\infty$ everywhere else. Then for $\lambda \in \mathbb{R}^d$,

$$
\tilde{g}^*(\lambda) = \sup_{z \in \mathbb{R}^d} \{ \langle \lambda, z \rangle - \tilde{g}(z) \} = \sup_{z \geq 0} \{ \langle \lambda, z \rangle - \tilde{g}(z) \} = g^*(\max(\lambda, 0)).
$$

Here, the first equality is by definition. The second is because $\tilde{g}(z) = \infty$ if $z$ is not non-negative. The third is because if some coordinate of $\lambda$ is negative, zeroing out the corresponding coordinate of $z$ increases $\langle \lambda, z \rangle$ and decreases $\tilde{g}(z)$, because $\tilde{g}$ is non-decreasing in positive orthant. Here vector $\max(\lambda, 0)$ is the coordinate-wise maximum.

Now for any $z \in \mathbb{R}^d_+$, $g(z) = \tilde{g}(z) = \max_{\lambda \in \mathbb{R}^d_+} \langle \lambda, z \rangle = \max_{\lambda \in \mathbb{R}^d_+} \langle \lambda, z \rangle - g^*(\max(\lambda, 0)) = \max_{\lambda \in \mathbb{R}^d_+} \langle \lambda, z \rangle - g^*(\lambda)$.

\section*{C.2 Removing Assumptions on the Elements}

Let $\text{opt}$ denote the profit of the optimal integer solution to the problem \eqref{eq:ip}. To discharge the conditions on elements in Assumption 3.1, we show the following reduction.

Lemma C.2. Suppose $A$ is algorithm that works for instances that have no exceptional items, where each item has profit $\pi(e) \leq \text{opt}/\eta d$, and where items are in general position, that guarantees a profit of $\text{opt}/\bar{\beta}$. Then we can get another algorithm that requires none of these assumptions, and guarantees a profit of $\frac{\text{opt}}{O(\beta + \eta d)}$.

Proof. The general position argument is simplest: we essentially need that for some fixed $\lambda \in \mathcal{C}$, there is at most one element such that $v(e) = \langle \lambda, s(e) \rangle$. This can be achieved by subtracting from each $v(e)$ some random noise picked uniformly from the interval $[0, (\delta/n) \pi(e)]$ for some tiny $\delta$; this can change the optimal profit most by a $(1 - \delta)$-factor.

Recall that item $e$ is called exceptional if $\arg \max_{\theta \in [0,1]} \{ \theta v(e) - g(\theta s(e)) \}$ is achieved at $\theta \in (0, 1)$: i.e., it is optimal to take a fraction of the item. E.g., in the 1-dimensional case, $v(e) = s(e) = 1$, and $g(s) = 0.99 s^2$.

The following claim is a minor variation of \cite[Lemma 5.1]{BUCM12}:

Claim C.3 (Few Exceptional Items). If $g$ is supermodular, then any optimal solution contains at most $d$ exceptional items.

Proof. Fix an optimal solution $O^*$, and for each $i = 1, \ldots, d$, let $o_i := \arg \max_{o \in O^*} s(o)_i$ be an item for which the $i$th-coordinate of the size vector is maximized. Let $L$ denote the set of these “large” items. If $O^*$ contains strictly more than $d$ exceptional items, let $o^* \in O^*$ be any exceptional item not in $L$, and let $x' := \chi_{O^* \setminus \{o^*\}}$ be the characteristic vector for the elements in the optimal set without $o^*$. By construction, $s(o^*) \leq Sx'$ component-wise. Since $o^*$ is exceptional, $v(o^*) < \langle \nabla g(s(o^*)), s(o^*) \rangle$; moreover, the latter is at most $\langle \nabla g(Sx'), s(o^*) \rangle$ due to $g$ having monotone gradients. But this implies that dropping $o^*$ would increase the profit, which contradicts our choice of $O^*$.

Moreover, since the profit function is subadditive, there can be at most $\eta d$ high-valued items. Now the reduction procedure: with probability $\frac{1}{2}$ run the single-item secretary problem (with competitive ratio $1/e$), and with the remaining probability run algorithm $A$. If the instance has a high-valued item then we get expected value at least $(1/e) \cdot \text{opt}/(\eta d)$. If not, divide the optimal solution $x^*$ into the solution restricted to the non-exceptional items $x^1$, and to the (at most $d$) exceptional items $x^2$. By the subadditivity of the profit, $\pi(x^*) \leq \pi(x^1) + \pi(x^2)$. Again the secretary algorithm gives a $1/(de)$-approximation for the profit $\pi(x^2)$, so it suffices to get a good approximation for the non-exceptional items.