Using Partial Monotonicity in Submodular Maximization

Anonymous Author(s)
Affiliation
Address
email

Abstract

Over the last two decades, submodular function maximization has been the workhorse of many discrete optimization problems in machine learning applications. Traditionally, the study of submodular functions was based on *binary* function properties, but recent works began to consider *continuous* function properties such as the submodularity ratio and the curvature. The monotonicity property of set functions plays a central role in submodular maximization. Nevertheless, no continuous version of this property has been suggested to date (as far as we know), which is unfortunate since submoduar functions that are almost monotone often arise in machine learning applications. In this work we fill this gap by defining the *monotonicity ratio*, which is a continues version of the monotonicity property. We then show that for many standard submodular maximization algorithms one can prove new approximation guarantees that depend on the monotonicity ratio; leading to improved approximation ratios for the common machine learning applications of movie recommendation, quadratic programming and image summarization.

1 Introduction

Over the last two decades, submodular function maximization has been the workhorse of many discrete optimization problems in machine learning applications such as data summarization [17, 19, 31, 32, 41, 49], social graph analysis [45], adversarial attacks [36], dictionary learning [15], sequence selection [42, 50], interpreting neural networks [18] and many more. Traditionally, the study of submodular functions was based on *binary* properties of functions. A function can be either submodular or non-submodular, monotone or non-monotone, etc. Such properties are simple, but they have an inherit weakness—if an algorithm assumes functions that have a particular property, then it provides no guarantee for functions that violate this property, even if the violation is slight.

Given the above situation, recent works began to consider *continuous* versions of function properties. Probably the most significant among these continuous versions so far are the submodularity ratio and the curvature. The submodularity ratio (originally defined by Das and Kempe [16]) is a parameter $\gamma \in [0, 1]$ replacing the binary submodularity property that a set function can either have or not have. A value of 1 corresponds to a fully submodular function, and lower values of γ represent some violation of submodularity (the worse the violation, the lower γ). Similarly, the curvature (defined by Conforti and Cornuéjol [13]) is a parameter $c \in [0, 1]$ replacing the binary linearity property that a set function can either have or not have. A value of 1 corresponds to a fully linear function, and lower values of c represent some violation of linearity.

A central conceptual contribution of Das and Kempe [16] was that they were able to demonstrate that continuous function properties further extend the usefulness of submodular maximization to new machine learning applications (such as subset selection for regression and dictionary selection). This has motivated a long list of works on such properties (see [3, 25, 26, 29, 34] for a few examples),

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

including works that combine both the submodularity ratio and the curvature (see, e.g., [3]). However, to the best of our knowledge, no continuous version of the binary monotonicity property has been suggested so far.[1] See Appendix A for additional related work.

We note that the monotonicity property of set functions plays a central role in submodular maximization, and basically every problem in this field has been studied for both monotone and non-monotone objective functions. Naturally, monotone objective functions enjoy improved approximation guarantees compared to general functions, and it is natural to ask how much of this improvement applies also to functions that are almost monotone (in some sense). Since such functions often arise in machine learning applications when a diversity promoting component is added to a basic monotone objective, obtaining better guarantees for them should strongly enhance the usefulness of submodular maximization as a tool for many machine learning applications.

Formally, a non-negative set function $f\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ over a ground set $\mathcal{N}$ is (increasingly) *monotone* if $f(S) \subseteq f(T)$ for every $S \subseteq T \subseteq \mathcal{N}$. Similarly, we define the *monotonicity ratio* of such a function f as the maximum value $m \in [0, 1]$ such that $m \cdot f(S) \leq f(T)$ for every two sets $S \subseteq T \subseteq \mathcal{N}$. Equivalently, one can define the monotonicity ratio m by $m \triangleq \min_{S \subseteq T \subseteq \mathcal{N}}[f(T)/f(S)]$, where the ratio $f(T)/f(S)$ is assumed to be 1 whenever $f(S) = 0$. Intuitively, the monotonicity ratio measures how much of the value of a set S can be lost when additional elements are added to S. One can view m as the distance of f from monotonicity. In particular, $m = 1$ if and only if f is monotone.

Our main contribution in this paper is demonstrating the usefulness of the monotonicity ratio in machine learning applications, which we do in two steps.

- First, we show (in Sections 3, 4 and 5) that for many standard submodular maximization algorithms one can prove new approximation guarantees that depend on the monotonicity ratio. These approximation guarantees interpolate between the known approximation ratios of these algorithms for monotone and non-monotone submodular functions.
- Then, using the above new approximation guarantees, we derive new approximation ratios for the standard applications of movie recommendation, quadratic programming and image summarization. Our guarantees improve over the state-of-the-art for most values of the problems' parameters. See Section 6 for more detail.

Remark. Computing the monotonicity ratio m of a given function seems to be difficult. Thus, the algorithms we analyze avoid assuming access to m, and the value of m is only used in the analyses of these algorithms. Nevertheless, in the context of particular applications, we are able to bound m, and plugging this bound into our general results yields our improved guarantees for these applications.

1.1 Our Results

Given a ground set $\mathcal{N}$, a set function $f\colon 2^{\mathcal{N}} \to \mathbb{R}$ is submodular if $f(S \cup \{u\}) - f(S) \geq f(T \cup \{u\}) - f(T)$ for every two sets $S \subseteq T \subseteq \mathcal{N}$ and element $u \in \mathcal{N} \setminus T$. Submodular maximization problems ask to maximize such functions subject to various constraints. To allow for multiplicative approximation guarantees for these problems, it is usually assumed that the objective function f is non-negative. Accordingly, we consider in this paper the following three basic problems.

- Given a non-negative submodular function $f\colon 2^{\mathcal{N}} \to \mathbb{R}$, find a set $S \subseteq \mathcal{N}$ that (approximately) maximizes f. This problem is termed "unconstrained submodular maximization", and is studied in Section 3.
- Given a non-negative submodular function $f\colon 2^{\mathcal{N}} \to \mathbb{R}$ and an integer parameter $0 \leq k \leq |\mathcal{N}|$, find a set $S \subseteq \mathcal{N}$ of size at most k that (approximately) maximizes f among such sets. This problem is termed "maximizing a submodular function subject to a cardinality constraint", and is studied in Section 4.
- Given a non-negative submodular function $f\colon 2^{\mathcal{N}} \to \mathbb{R}$ and a matroid $\mathcal{M}$ over the same ground set, find a set $S \subseteq \mathcal{N}$ that is independent in $\mathcal{M}$ and (approximately) maximizes f among such sets. This problem is termed "maximizing a submodular function subject to a matroid constraint", and is studied in Section 5 (see Section 5 for the definition of matroids).

[1] Following the appearance of the pre-print version of this paper, we learned that Iyer defined in his Ph.D. thesis [30] such a property, which is identical to the one we define. However, Iyer only used this property to prove the result appearing below as Theorem 4.1; thus, our work is the first to systematically study this property.

We present both algorithmic and inapproximability results for the above problems. Our algorithmic results reanalyze a few standard algorithms, and surprisingly show that almost all these algorithms guarantee an approximation ratio of $m \cdot \alpha_{\text{mon}} + (1-m) \cdot \alpha_{\text{non-mon}}$, where m is the monotonicity ratio, α_{mon} is the approximation ratio known for the algorithm when f is monotone, and $\alpha_{\text{non-mon}}$ is the approximation ratio known for the algorithm when f is a general non-negative submodular function.

While the above mentioned algorithmic results lead to our improved guarantees for applications, our inapproximability results represent our main technical contribution. In general, these results are based on the symmetry gap framework of Vondrák [51]. The original version of this framework is able to deal both with the case of general (not necessarily monotone) submodular functions, and with the case of monotone submodular functions; which in our terms correspond to the cases of $m \geq 0$ and $m \geq 1$, respectively. However, to prove our inapproximability results, we had to show that the framework extends to arbitrary lower bounds on m, which was challenging because the original proof of the framework is highly based on derivatives of continuous functions. From this point of view, submodularity is defined as having non-positive second-order derivatives, and monotonicity is defined as having non-negative first-order derivatives. However, the definition of the monotonicity ratio cannot be easily restated in terms of derivatives; and thus, handling it required us to come up with a different proof approach.

Interestingly, our inapproximability result for unconstrained submodular maximization proves that the optimal approximation ratio for this problem does not exhibit a linear dependence on m. Thus, the nice linear dependence demonstrated by almost all our algorithmic results is probably an artifact of looking at standard algorithms rather than representing the true nature of the monotonicity ratio, and we expect future algorithms tailored to take advantage of the monotonicty ratio to improve over this linear dependence. The reason that we concentrate in this work on reanalyzing standard submodular maximization algorithms rather than inventing new ones is that we want to stress the power obtained by using the new notion of monotonicity ratio, as opposed to power gained via new algorithmic innovations. This is in line with the research history of the submodularity ratio and the curvature. For both of these parameters, the original works concentrated on reanalyzed the standard greedy algorithm in view of the new suggested parameter; and the invention of algorithms tailored to the parameter was deferred to later works (see [48] and [12] for examples of such algorithms for the curvature and submodularity ratio, respectively).

Over the years, the standard submodular maximization algorithms have been extended and improved in various ways. Some works presented accelerated and/or parallelized versions of these algorithms, while other works generalized the algorithms beyond the realm of set functions (for example, to (DR-)submodular functions over lattices or continuous domains). Since our motivation in this paper is related to the monotonicity ratio, which is essentially independent of the extensions and improvements mentioned above, we mostly analyze the vanilla versions of all the algorithms considered. This keeps our analyses relatively simple. However, our experiments are based on more state-of-the-art versions of the algorithms. Similarly, many continuous properties (including the submodularity ratio) have weak versions that only depend on the behavior of the function for nearly feasible sets, and immediately enjoy most of the results that apply to the original strong property. The definition of such weak versions is useful for capturing additional application, but often add little from a theoretical perspective. Therefore, in the theoretical parts of this paper we consider only the monotonicity ratio as it is defined above; but for the sake of one of our applications we later define also the natural corresponding weak property.

2 Preliminaries and Basic Observations

In this section we define the notation used in this paper, and then state some useful basic observations. Given an element $u \in \mathcal{N}$ and a set $S \subseteq \mathcal{N}$, we use $S + u$ and $S - u$ as shorthands for $S \cup \{u\}$ and $S \setminus \{u\}$. Additionally, given a set function $f \colon 2^{\mathcal{N}} \to \mathbb{R}$, we define $f(u \mid S) \triangleq f(S+u) - f(S)$—this value is known as the marginal contribution of u to S with respect to f. Similarly, given an additional set $T \subseteq \mathcal{N}$, we define $f(T \mid S) \triangleq f(S \cup T) - f(S)$. We also use $\mathbf{1}_S$ to denote the characteristic vector of the set S, i.e., a vector in $[0,1]^{\mathcal{N}}$ that has 1 in the coordinates corresponding to elements that appear in S and 0 to the other coordinates. Finally, if f is non-negative, we say that it is m-monotone if its monotonicity ratio is at least m; and given an event $\mathcal{E}$, we denote by $\mathbf{1}[\mathcal{E}]$ the indicator of this event, i.e., a random variable that takes the value 1 when the event happens, 0 otherwise .

Next, we present a well-known continuous extension of set functions. Given a set function $f\colon 2^{\mathcal{N}} \to \mathbb{R}$, its *multilinear extension* is a function $F\colon [0,1]^{\mathcal{N}} \to \mathbb{R}$ defined as follows. For every vector $\mathbf{x} \in [0,1]^{\mathcal{N}}$, let $\mathtt{R}(\mathbf{x})$ to be a random subset of $\mathcal{N}$ that includes every element $u \in \mathcal{N}$ with probability x_u, independently. Then, $F(\mathbf{x}) = \mathbb{E}[f(\mathtt{R}(\mathbf{x}))]$. Another extension of set functions is central to the proof of the next lemma, which generalizes Lemma 2.2 of [6]—see Appendix B for the proof.

Lemma 2.1. *Let $f\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be a non-negative m-monotone submodular function. For every deterministic set $O \subseteq \mathcal{N}$ and random set $D \subseteq \mathcal{N}$, $\mathbb{E}[f(O \cup D)] \geq (1 - (1-m) \cdot \max_{u \in \mathcal{N}} \Pr[u \in D]) \cdot f(O)$.*

We conclude this section with the following observation, which we view as evidence that the class of non-negative m-monotone functions is a natural class for every $m \in [0,1]$.

Observation 2.2. *For every two non-negative m-monotone functions $f, g\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ and constant $c \geq 0$, the following functions are also m-monotone: (i) $h(S) = f(S) + g(S)$, (ii) $h(S) = f(S) + c$, and (iii) $h(S) = c \cdot f(S)$.*

3 Unconstrained Maximization

Recall that in the unconstrained submodular maximization problem, we are given a non-negative submodular function $f\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$, and the objective is to find a set $S \subseteq \mathcal{N}$ that (approximately) maximizes $f(S)$. Buchbinder et al. [7] gave the first $1/2$-approximation algorithm for this problem, known as the double greedy algorithm. The $1/2$-approximation guarantee of double greedy is known to be optimal in general due to a matching inapproximability result due to Feige et al. [21]. Nevertheless, in this section we study the extent to which one can improve over this guarantee as a function of the monotonicity ratio m of f.

Theorem 3.1. *There exists a polynomial time algorithm that obtains an approximation ratio of $\max\{m, (2+m)/4\}$ for unconstrained submodular maximization, but no such algorithm obtains an approximation ratio of $1/(2-m) + \varepsilon$ for any constant $\varepsilon > 0$.*[2]

While Theorem 3.1 does not settle the approximability of unconstrained submodular maximization as a function of m, the gap between its positive and negative results is quite small. More importantly, Theorem 3.1 implies that the optimal approximation ratio for unconstrained submodular maximization cannot have a linear dependence on m.

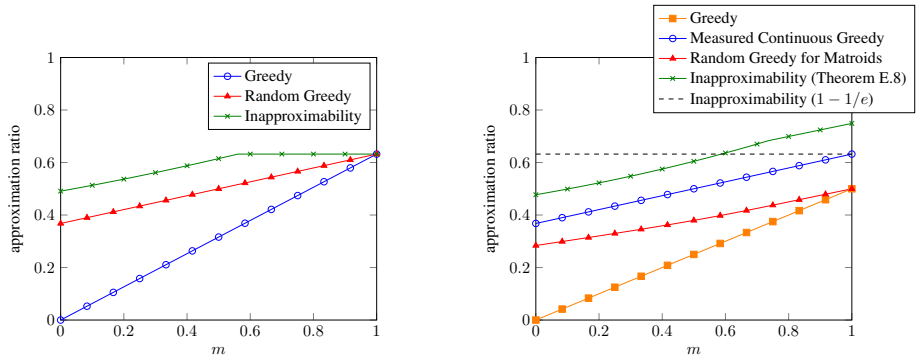
The first part of Theorem 3.1 is proved in Appendix C.1 using an algorithm that outputs the better solution among $\mathcal{N}$ and the output of the double greedy algorithm of Buchbinder et al. [7]. Below, we concentrate on proving the second part of Theorem 3.1. We do this using a generalization of the symmetry gap framework of Vondrák [51] that is informally stated as Theorem 3.2 (see Appendix C.2 for the formal statement of the theorem).

Theorem 3.2. *Consider a non-negative m-monotone submodular function f and a collection $\mathcal{F} \subseteq 2^{\mathcal{N}}$ of feasible sets such that the problem $\max\{f(S) \mid S \in \mathcal{F}\}$ is symmetric with respect to some group $\mathcal{G}$ of permutations over $\mathcal{N}$. If the best solution for this problem which is symmetric with respect to $\mathcal{G}$ is worse by a factor of γ compared to the optimal solution, then we say that the problem has a symmetry gap of γ. In this case, exponentially many value oracle queries are required to obtain $(1+\varepsilon)\gamma$-approximation for the class of problems $\max\{\tilde{f}(S) \mid S \in \tilde{F}\}$ in which $\tilde{f}$ is a non-negative m-monotone submodular function, and $\tilde{F}$ is some generalization of $\mathcal{F}$ (in particular, if $\mathcal{F}$ is a matroid/cardinality constraint, then so is $\tilde{F}$).*

To use Theorem 3.2, we need to define a submodular maximization problem with a significant symmetry gap. Let us choose $\mathcal{N} = \{u, v\}$, $f(S) = m \cdot \mathbf{1}[S \neq \varnothing] + (1-m) \cdot (|S| \bmod 2)$ and $\mathcal{F} = 2^{\mathcal{N}}$, where m is an arbitrary constant $m \in [0,1]$. One can verify that f is submodular and non-negative, that its monotonicity ratio is exactly m, and that the problem $\max\{f(S) \mid S \in \mathcal{F}\}$ is symmetric with respect to the group $\mathcal{G}$ of the two possible permutations of $\mathcal{N}$. The following lemma calculates the symmetry gap of this problem, and its proof can be found in Appendix C.3. The second part of Theorem 3.1 follows from this lemma and Theorem 3.2.

Lemma 3.3. *The problem $\max\{f(S) \mid S \in \mathcal{F}\}$ has a symmetry gap of $\frac{1}{2-m}$.*

[2] In the second part of Theorem 3.1, like in all the other inapproximability results in this paper, we make the standard assumption that the objective function f can be accessed only through a value oracle that given a set $S \subseteq \mathcal{N}$ returns $f(S)$.



(a) Results for cardinality constraint (Section 4) (b) Results for matroid constraint (Section 5)

Figure 1: Graphical representation of the results of Sections 4 and 5

4 Maximization with a Cardinality Constraint

In this section we consider the problem of maximizing a non-negative submodular function $f\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ subject to a cardinality constraint. In other words, we are given an integer value $1 \leq k \leq |\mathcal{N}|$, and the objective is to output a set $S \subseteq \mathcal{N}$ of size at most k (approximately) maximizing f among such sets. When the objective function f is guaranteed to be monotone, it is long known that a standard greedy algorithm guarantees $(1 - 1/e)$-approximation for the above problem [44], and that this is essentially the best possible for any polynomial time algorithm [43]. However, the greedy algorithm has no constant approximation guarantee when the objective function is not guaranteed to be monotone (see [4] for an example demonstrating this). In Appendix D.1 we prove Theorem 4.1, which generalizes the result of [44], and proves an approximation guarantee for the greedy algorithm that deteriorates gracefully with the monotonicity ratio m.

Theorem 4.1. *The Greedy algorithm (Algorithm 1) has an approximation ratio of at least $m(1 - 1/e)$ for the problem of maximizing a non-negative m-monotone submodular function subject to a cardinality constraint.*

Following a long line of works [35, 22, 46, 51, 6, 20], the state-of-the-art approximation guarantee for the case in which the objective function f is not guaranteed to be monotone is currently 0.385 [5]. However, the algorithm obtaining this approximation ratio is quite involved, which limits its practicality. Arguably, the state-of-the-art approximation ratio obtained by a "simple" algorithm is the $1/e \approx 0.367$-approximation obtained by an algorithm called Random Greedy. Furthermore, this algorithm has the nice property that for monotone objective functions it recovers the optimal $1 - 1/e$ approximation guarantee. In Appendix D.2 we prove Theorem 4.2, which gives an approximation guarantee for Random Greedy that smoothly changes as a function of m and recovers the above mentioned $1/e$ and $1 - 1/e$ guarantees.

Theorem 4.2. *Random Greedy (Algorithm 2) has an approximation ratio of at least $m(1 - 1/e) + (1 - m) \cdot (1/e)$ for the problem of maximizing a non-negative m-monotone submodular function subject to a cardinality constraint.*

There is still a gap between the state-of-the-art 0.385-approximation for non-monotone objectives and the state-of-the-art inapproximability result due to Oveis Gharan and Vondrák [46], which only shows that no polynomial time algorithm can guarantee a better than roughly 0.491-approximation. In Appendix D.3 we give Theorem D.4, which proves an inapproximability result that smoothly depends on m and recovers the above mentioned inapproximability results for $m = 0$ and $m = 1$.

To get an intuitive understanding of Theorem D.4, we numerically evaluated it for various values of m. The plot obtained in this way appears in Figure 1a. For context, this figure also includes all the other results proved in this section. As is evident from Figure 1a, Theorem D.4 improves over the $1 - 1/e$ inapproximability result of Nemhauser and Wolsey [43] only for m that is smaller than roughly 0.56. This is surprising since, intuitively, one would expect the best possible approximation ratio to be strictly worse than $1 - 1/e$ for any $m < 1$. However, we were unable to prove an inapproximability that is even slightly lower than $1 - 1/e$ for any $m > 0.56$. Understanding whether this is an artifact of our proof or a real phenomenon is an interesting question that we leave open.

5 Maximization with a Matroid Constraint

In this section we consider the problem of maximizing a non-negative submodular function subject to a matroid constraint. A matroid $\mathcal{M}$ over the ground set $\mathcal{N}$ is defined as a pair $\mathcal{M} = (\mathcal{N}, \mathcal{I})$, where $\mathcal{I}$ is a non-empty subset of $2^{\mathcal{N}}$ obeying two properties for every two sets $S, T \subseteq \mathcal{N}$: (i) if $S \subseteq T$ and $T \in \mathcal{I}$, then $S \in \mathcal{I}$; and (ii) if $S, T \in \mathcal{I}$ and $|S| < |T|$, then there exists an element $u \in T \setminus S$ such that $S + u \in \mathcal{I}$. A set $S \subseteq \mathcal{N}$ is called *independent* with respect to a matroid $\mathcal{M}$ if it belongs to $\mathcal{I}$ (otherwise, we say that S is dependent with respect to $\mathcal{M}$); and the matroid constraint corresponding to a given matroid $\mathcal{M}$ allows only independent sets with respect to this matroid as feasible solutions. Hence, we can restate the problem we consider in this section in the following more formal way. Given a non-negative submodular function $f\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ and a matroid $\mathcal{M} = (\mathcal{N}, \mathcal{I})$, output an independent set $S \in \mathcal{I}$ (approximately) maximizing f among all such sets.

When the objective function f is guaranteed to be monotone, a standard greedy algorithm guarantees $1/2$-approximation for the above problem [24]. Our first result for this section (proved in Appendix E.1) shows how this approximation guarantee changes as a function of m (the greedy algorithm has no constant guarantee for non-monotone functions in this case as well).

Theorem 5.1. *The Greedy algorithm (Algorithm 3) has an approximation ratio of at least $m/2$ for maximizing a non-negative m-monotone submodular function subject to a matroid constraint.*

The approximation ratio of the greedy algorithm was improved over by the seminal work of Călinescu et al. [9], who described the Continuous Greedy algorithm whose approximation ratio is $1 - 1/e$ when f is monotone; matching the inapproximability result of Nemhauser and Wolsey [43]. In contrast, when f is not guaranteed to be monotone, the approximability of the problem is less well-understood. On the one hand, after a long line of works [35, 22, 46, 51, 20], the state-of-the-art approximation ratio for the problem is 0.385 [5], but on the other hand, it is only known that no polynomial time algorithm for the problem can guarantee 0.478-approximation [46].

Unfortunately, the above mentioned state-of-the-art 0.385-approximation algorithm is quite involved. Therefore, we chose to consider in this work two other algorithms. The first algorithm is Measure Continuous Greedy (due to [22]) which guarantees an approximation ratio of $1/e - o(1) \approx 0.367$. This algorithm performs only slightly worse than the above state-of-the-art, and is a central component of all the currently known algorithms achieving better than $1/e$-approximation. Measured Continuous Greedy is also known to guarantee $(1 - 1/e - o(1))$-approximation when the objective f is monotone, and the next theorem (proved in Appendix E.2) shows that its approximation guarantee changes smoothly with the monotonicity ratio of f.

Theorem 5.2. *Measured Continuous Greedy (Algorithm 4) has an approximation ratio of at least $m(1 - 1/e) + (1 - m) \cdot (1/e) - o(1)$ for maximizing a non-negative m-monotone submodular function subject to a matroid constraint, where the $o(1)$ term diminishes with the ground set's size.*[3]

The other algorithm we consider is an algorithm called Random Greedy for Matroids (due to [6]). Unlike Measured Continuous Greedy and almost all the other algorithms suggested for non-monotone objectives to date, this algorithm is combinatorial, which makes it appealing in practice. Buchbinder et al. [6] proved an approximation ratio of roughly $(1 + e^{-2})/4$ for Random Greedy for Matroids. The next theorem shows how this approximation guarantee improves as a function of the monotonicity ratio. In this theorem we refer to the rank k of the matroid constraint $\mathcal{M}$, which is the size of the largest independent set with respect to this matroid. We also note that the algorithm we analyze is identical to the algorithm of [6] up to two modifications: our algorithm makes more iterations, and it updates the solution in an iteration only when this increases the solution's value.

Theorem 5.3. *For every $\varepsilon \in (0, 1)$, Random Greedy for Matroids (Algorithm 5) has an approximation ratio of at least $\frac{1+m+e^{-2/(1-m)}}{4} - \varepsilon - o_k(1)$ for the problem of maximizing a non-negative m-monotone submodular function subject to a matroid constraint (except in the case of $m = 1$ in which the approximation ratio is $1/2 - \varepsilon - o_k(1)$), where $o_k(1)$ represents a term that diminishes with k.*

Theorem 5.3 is proved in Appendix E.3. Let us also mention Theorem E.8, which appears in Appendix E.4 and generalizes the 0.478 inapproximability result of Oveis Gharan and Vondrák [46].

[3] Technically, Measured Continuous Greedy is an algorithm for maximizing the multilinear extesnion of a non-negative submodular function subject to a general solvable down-closed convex body P constraint, and we prove in Appendix E.2 that it guarantees the approximation ratio stated in Theorem 5.2 for this setting. However, this implies the result stated in Theorem 5.2 using a standard reduction (see Appendix E.2 for further detail).

To get an intuitive understanding of this theorem, we numerically evaluated it for various values of m, and depict the results in Figure 1b. For context, this figure also includes all the other results proved in this section. Somewhat surprisingly, Figure 1b shows that Theorem E.8 does not generalize the $1 - 1/e$ inapproximability result of Nemhauser and Wolsey [43] for monotone functions despite the fact that this inapproximability result holds for every monotonicity ratio $m \in [0, 1]$. This resembles the inability of Theorem D.4 to improve over the same inapproximability result for large values of m.

6 Applications and Experiment Results

Many machine learning applications require optimization of non-monotone submodular functions subject to some constraint. Unfortunately, such functions enjoy relatively low approximation guarantees. Nevertheless, in many cases the non-monotone objective functions have a significant monotone component that can be captured by the monotonicity ratio. In this section, we discuss three concrete applications with non-monotone submodular objective functions. For each application we provide a lower bound on the monotonicity ratio m of the objective function, which translates via our results from the previous sections into an improved approximation guarantee for the application.

To demonstrate the value of our improved guarantees in experiments, we took the following approach. The output of an approximation algorithm provides an upper bound on the value of the optimal solution for the problem (formally, this upper bound is the value of the output over the approximation ratio of the algorithm). Thus, we plot in each experiment the upper bound on the value of the optimal solution obtained with and without taking into account the monotonicity ratio, which gives a feeling of how the magnitude of our improvements compare to other values of interest (such as the gaps between the performances of the algorithms considered). In Appendix F we give a third application (Image summarization) that we study in the same way.

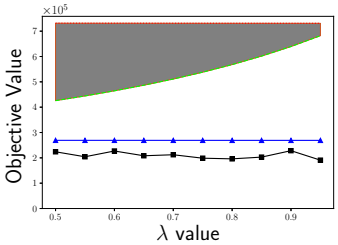
6.1 Personalized Movie Recommendation

The first application we consider is Personalized Movie Recommendation. Consider a movie recommendation system where each user specifies what genres she is interested in, and the system has to provide a representative subset of movies from these genres. Assume that each movie is represented by a vector consisting of users' ratings for the corresponding movie. One challenge here is that each user does not necessarily rate all the movies, hence, the vectors representing the movies do not necessarily have similar sizes. To overcome this challenge, a low-rank matrix completion techniques [10] can be performed on the matrix with missing values in order to obtain a complete rating matrix. Formally, given few ratings from k users to n movies we obtain in this way a rating matrix $\mathbf{M}$ of size $k \times n$. Following [40], to score the quality of a selected subset of movies, we use the function $f(S) = \sum_{u \in \mathcal{N}} \sum_{v \in S} s_{u,v} - \lambda \sum_{u \in S} \sum_{v \in S} s_{u,v}$. Here, $\mathcal{N}$ is the set of n movies, $\lambda \in [0, 1]$ is a parameter and $s_{u,v}$ denotes the similarity between movies u and v (the similarity $s_{u,v}$ can be calculated based on the matrix $\mathbf{M}$ in multiple ways: cosine similarity, inner product, etc). Note that the first term in f's definition captures coverage, while the second term captures diversity. Thus, the parameter λ denotes the importance of diversity in the returned subset.

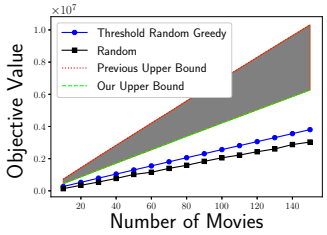
One can verify that the above defined function f is non-negative and submodular. The next theorem, proved in Appendix G.1, analyzes the monotonicity ratio of this function. In this theorem we assume that the similarity scores $s_{u,v}$ are non-positive and obey $s_{u,v} = s_{v,u}$ for every $u, v \in \mathcal{N}$. Note that the above mentioned ways to define these scores have these properties. Interestingly, it turns out that the function f is monotone when λ is small enough despite the fact that this function is traditionally treated as non-monotone (e.g., in [40, 23]). This is a nice unexpected result of the use of the monotonicity ratio, which required us to really understand the degree of non-monotonicity represented by the objective function.

Theorem 6.1. *The objective function f is monotone for $0 \leq \lambda \leq 1/2$ and $2(1 - \lambda)$-monotone for $1/2 \leq \lambda \leq 1$.*

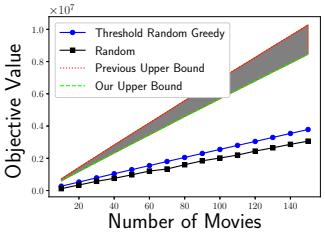
To demonstrate the value of our lower bound on the monotonicity ratio, we followed the experimental setup of [40] and used a subset of movies from the MovieLens data set [28] which includes $10{,}437$ movies. Each movie in this data set is represented by a 25 dimensional feature vector calculated using user ratings, and we used inner products to obtain the similarity values $s_{i,j}$ based on these vectors.



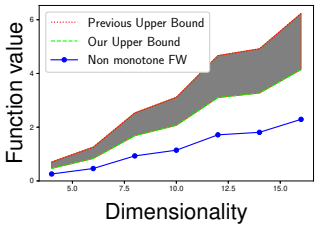
(a) Results when up to 10 movies can be selected for varying λ.



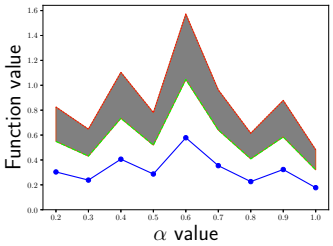
(b) Results for $\lambda = 0.55$ when the number movies in the solution varies.



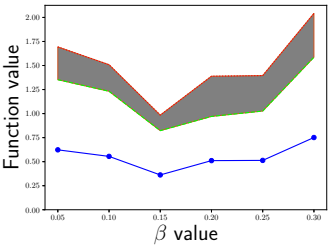
(c) Results for $\lambda = 0.85$ when the number movies in the solution varies.



(d) Varying the dimensionality n for fixed $\alpha = 0.3$ and $\beta = 0.2$.



(e) Varying α for fixed $\beta = 0.2$ and $n = 4$.



(f) Varying β for fixed $\alpha = 0.5$ and $n = 4$.

Figure 2: Experimental results for Personalized Movie Recommendation (a–c) and Quadratic Programming (d–f). Each plot includes the output of the algorithms we consider as well the previous and improved upper bounds on the optimal value (the area between these two bounds is shaded).

In our experiment we employed accelerated versions of the algorithms analyzed in Section 4 for a cardinality constraint. Specifically, instead of the Greedy algorithm we used Threshold Greedy [1] and Sample Greedy [39]; and instead of Random Greedy we used a threshold based version of this algorithm due to [8] that we refer to as Threshold Random Greedy (Algorithm 6 in [8]). All three algorithms had almost identical performance in our experiments (see Appendix H), thus, to avoid confusion, in Figure 2 we draw only the output of Threshold Random Greedy.

Each plot of Figure 2 depicts the outputs of Threshold Random Greedy and a scarecrow algorithm called Random that simply outputs a random subset of movies of the required size. Each point in the plots represents the average value of the outputs of 10 executions of these algorithms. We also depict in each plot the upper bound on the value of the optimal solution based on the general approximation ratio of Random Greedy and the improved approximation ratio implied by Theorems 4.2 and 6.1—the area between the two upper bounds is shaded. In Figure 2a we plot these values for the case in which we asked the algorithms to pick at most 10 movies, and we vary the parameter λ. In Figures 2b amd 2c we plotted the same values for a fixed parameter λ, while varying the maximum cardinality (number of movies) allowed for the output set. Since the height of the shaded area is on the same order of magnitude as the values of the solutions produced by Threashold Random Greedy (especially when λ is close to $1/2$), our results demonstrate that the improved upper bound we are able to prove is much tighter than the state-of-the-art. Furthermore, our improved upper bound shows that the gap between the empirical outputs of Threshold Random Greedy and Random is much more significant as a percentage of the value of the optimal solution than one could believe based on the weaker bound.

6.2 Quadratic Programming

Consider the function

$$F(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{h}^T\mathbf{x} + c \ . \tag{1}$$

By choosing appropriate matrix $\mathbf{H}$, vector $\mathbf{h}$ and scalar c, this function can be made to have various properties. Specifically, we would like to make it non-negative and DR-submodular (DR-submodularity is an extension of submodularity to continuous functions—see Appendix I for

more detail). Our goal in this section is to maximize F under a polytope constraint given by $P = \{\mathbf{x} \in \mathbb{R}^n_{\geq 0} \mid A\mathbf{x} \leq \mathbf{b}, \mathbf{x} \leq \mathbf{u}, A \in \mathbb{R}^{m \times n}_{\geq 0}, \mathbf{b} \in \mathbb{R}^m_{\geq 0}\}$ for some dimensions n and m.

Following Bian et al. [2], we set $m = n$, choose the matrix $\mathbf{H} \in \mathbb{R}^{n \times n}$ to be a randomly generated symmetric matrix whose entries are drawn uniformly at random (and independently) from $[-1, 0]$, and choose $\mathbf{A} \in \mathbb{R}^{m \times n}$ to be a randomly generated matrix whose entries are drawn uniformly at random from $[v, v+1]$ for $v = 0.01$ (this choice of v guarantees that the entries of A are strictly positive). We also set $\mathbf{b} = \bar{1}$ (i.e., $\mathbf{b}$ is the all ones vector), and $\mathbf{u}$ to be the upper bound on P given by $u_j = \min_{j \in [m]} b_i / A_{i,j}$ for every $j \in [n]$. Finally, we set $\mathbf{h} = -\beta \cdot \mathbf{H}^T \mathbf{u}$ for a parameter $\beta > 0$.

The non-positivity of $\mathbf{H}$ guarantees that f is DR-submodular. To make sure that f is also non-negative, the value of c should be at least $-\min_{\bar{0} \leq \mathbf{x} \leq \mathbf{u}} \frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{h}^T \mathbf{x}$ (where $\bar{0}$ is the all zeros vector). This value can be approximately obtained by using QUADPROGIP[4] [52]. Let the value of this minimum be M; then we set $c = -M + \alpha|M|$ for some parameter $\alpha > 0$.

The definition of the monotonicity ratio can be extend to the continuous setting we consider in this section using the formula $m = \inf_{\bar{0} \leq \mathbf{x} \leq \mathbf{y} \leq \mathbf{u}} \frac{F(\mathbf{y})}{F(\mathbf{x})}$, where the ratio $F(\mathbf{y})/F(\mathbf{x})$ is understood as 1 whenever $F(\mathbf{x}) = 0$. The following theorem analyzes the monotonicity ratio of the function F given in Equation (1) based on this definition. The proof of this theorem can be found in Appendix G.2.

Theorem 6.2. *For $\beta \in (0, 1/2)$, the objective function F given by Equation* (1) *is $\frac{(1-2\beta)\cdot\alpha}{1+\alpha}$-monotone. Furthermore, when $\min_{\bar{0} \leq x \leq \mathbf{u}}(\frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{h}\mathbf{x}) \geq 0$, F is even $(1 - 2\beta)$-monotone.*

We applied the Non-monotone Frank-Wolfe algorithm of Bian et al. [2] to the above defined optimization problem (we refer the reader to Appendix I for further detail about this algorithm and its analysis). Figures 2d, 2e and 2f depict the results we obtained. Specifically, Figure 2d shows the value of the solution obtained by Non-monotone Frank-Wolfe for $\alpha = 0.3$ and $\beta = 0.2$ as the dimensionality n varies. The shaded area is the area between the previous upper bound on the optimal value, and our upper bound that takes advantage on the monotonicity ratio bound given by Theorem 6.2. Figures 2e and 2f are similar, but they fix the dimensionality n to be 4, and vary α or β instead. Let us discuss now some properties of Figures 2d, 2e and 2f. (i) Each data point in these figures corresponds to a single instance drawn from the distribution described above. This implies that the plots in these figures vary for different runs of our experiment, but the plots that we give represent a (single) typical run. (ii) The size of the the shaded area depends on α and β, but also on the sign of $\min_{\bar{0} \leq \mathbf{x} \leq \mathbf{u}}(\frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{h}\mathbf{x})$. This is the reason that this size behaves somewhat non-continuously in Figure 2f. Interestingly, the sign of this minimum is mostly a function of β. In other words, there are values of β for which the minimum is non-negative with high probability, and other values for which the minimum is negative with high probability. (iii) One can see that the use of the monotonicity ratio significantly improves the upper bound on the optimal value, especially when $\min_{\bar{0} \leq \mathbf{x} \leq \mathbf{u}}(\frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{h}\mathbf{x})$ happens to be non-negative.

7 Conclusion

In this paper we have defined the monotonicity ratio, analyzed how the approximation ratios of standard submodular maximization algorithms depend on this ratio, and then demonstrated that this leads to improved approximation guarantees for the applications of movie recommendation, image summarization and quadratic programming. We believe that the monotonicity ratio is a natural parameter of submodular maximization problems, refining the binary distinction between monotone and non-monotone objective functions and improving the power of submodular maximization tools in machine learning applications. Thus, we hope to see future work towards understanding the optimal dependence on m of the approximation ratios of various submodular maximization problems.

An important take-home message from our work is that, at least in the unconstrained submodular maximization case, the optimal algorithm has an approximation ratio whose dependence on m is non-linear. Such algorithms are rarely obtained using current techniques, which might be one of the reasons why these techniques have so far failed to obtain tight approximation guarantees for constrained non-monotone submodular maximization.

[4] We have used IBM CPLEX optimization studio `https://www.ibm.com/products/ilog-cplex-optimization-studio`.

References

[1] Ashwinkumar Badanidiyuru and Jan Vondrák. Fast algorithms for maximizing submodular functions. In *SODA*, pages 1497–1514, 2014. doi: 10.1137/1.9781611973402.110. URL https://doi.org/10.1137/1.9781611973402.110.

[2] An Bian, Kfir Yehuda Levy, Andreas Krause, and Joachim M. Buhmann. Non-monotone continuous DR-submodular maximization: Structure and algorithms. In *NeurIPS*, pages 486–496, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/58238e9ae2dd305d79c2ebc8c1883422-Abstract.html.

[3] Andrew An Bian, Joachim M. Buhmann, Andreas Krause, and Sebastian Tschiatschek. Guarantees for greedy maximization of non-submodular functions with applications. In *ICML*, pages 498–507, 2017. URL http://proceedings.mlr.press/v70/bian17a.html.

[4] Niv Buchbinder and Moran Feldman. Submodular functions maximization problems. In Teofilo F. Gonzalez, editor, *Handbook of Approximation Algorithms and Metaheuristics, Second Edition, Volume 1: Methologies and Traditional Applications*, pages 753–788. Chapman and Hall/CRC, 2018. doi: 10.1201/9781351236423-42. URL https://doi.org/10.1201/9781351236423-42.

[5] Niv Buchbinder and Moran Feldman. Constrained submodular maximization via a nonsymmetric technique. *Math. Oper. Res.*, 44(3):988–1005, 2019. doi: 10.1287/moor.2018.0955. URL https://doi.org/10.1287/moor.2018.0955.

[6] Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. Submodular maximization with cardinality constraints. In Chandra Chekuri, editor, *SODA*, pages 1433–1452. SIAM, 2014. doi: 10.1137/1.9781611973402.106. URL https://doi.org/10.1137/1.9781611973402.106.

[7] Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. A tight linear time (1/2)-approximation for unconstrained submodular maximization. *SIAM J. Comput.*, 44(5):1384–1402, 2015. doi: 10.1137/130929205. URL https://doi.org/10.1137/130929205.

[8] Niv Buchbinder, Moran Feldman, and Roy Schwartz. Comparing apples and oranges: Query trade-off in submodular maximization. *Math. Oper. Res.*, 42(2):308–329, 2017. doi: 10.1287/moor.2016.0809. URL https://doi.org/10.1287/moor.2016.0809.

[9] Gruia Călinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.*, 40(6):1740–1766, 2011. doi: 10.1137/080733991. URL https://doi.org/10.1137/080733991.

[10] Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

[11] Chandra Chekuri, Jan Vondrák, and Rico Zenklusen. Dependent randomized rounding via exchange properties of combinatorial structures. In *FOCS*, pages 575–584, 2010. doi: 10.1109/FOCS.2010.60. URL https://doi.org/10.1109/FOCS.2010.60.

[12] Lin Chen, Moran Feldman, and Amin Karbasi. Weakly submodular maximization beyond cardinality constraints: Does randomization help greedy? In *ICML*, pages 803–812, 2018. URL http://proceedings.mlr.press/v80/chen18b.html.

[13] Michele Conforti and Gérard Cornuéjols. Submodular set functions, matroids and the greedy algorithm: Tight worst-case bounds and some generalizations of the rado-edmonds theorem. *Discret. Appl. Math.*, 7(3):251–274, 1984. doi: 10.1016/0166-218X(84)90003-9. URL https://doi.org/10.1016/0166-218X(84)90003-9.

[14] Min Cui, Donglei Du, Dachuan Xu, and Ruiqi Yang. Approximation algorithm for maximizing nonnegative weakly monotonic set functions. In *CSoNet*, pages 50–58, 2021. doi: 10.1007/978-3-030-91434-9_5. URL https://doi.org/10.1007/978-3-030-91434-9_5.

[15] Abhimanyu Das and David Kempe. Submodular meets spectral: greedy algorithms for subset selection, sparse approximation and dictionary selection. In *ICML*, pages 1057–1064, 2011.

[16] Abhimanyu Das and David Kempe. Approximate submodularity and its applications: Subset selection, sparse approximation and dictionary selection. *J. Mach. Learn. Res.*, 19:3:1–3:34, 2018. URL http://jmlr.org/papers/v19/16-534.html.

[17] Anirban Dasgupta, Ravi Kumar, and Sujith Ravi. Summarization through submodularity and dispersion. In *ACL*, pages 1014–1022. The Association for Computer Linguistics, 2013. URL https://aclanthology.org/P13-1100/.

[18] Ethan R. Elenberg, Alexandros G. Dimakis, Moran Feldman, and Amin Karbasi. Streaming weak submodularity: interpreting neural networks on the fly. In *NeurIPS*, pages 4047–4057, 2017.

[19] Ehsan Elhamifar and M Clara De Paolis Kaluza. Online summarization via submodular and convex optimization. In *CVPR*, pages 1783–1791, 2017.

[20] Alina Ene and Huy L. Nguyen. Constrained submodular maximization: Beyond $1/e$. In Irit Dinur, editor, *FOCS*, pages 248–257. IEEE Computer Society, 2016. doi: 10.1109/FOCS.2016.34. URL https://doi.org/10.1109/FOCS.2016.34.

[21] Uriel Feige, Vahab S. Mirrokni, and Jan Vondrák. Maximizing non-monotone submodular functions. *SIAM J. Comput.*, 40(4):1133–1153, 2011. doi: 10.1137/090779346. URL https://doi.org/10.1137/090779346.

[22] Moran Feldman, Joseph (Seffi) Naor, and Roy Schwartz. A unified continuous greedy algorithm for submodular maximization. In Rafail Ostrovsky, editor, *FOCS*, pages 570–579. IEEE Computer Society, 2011. doi: 10.1109/FOCS.2011.46. URL https://doi.org/10.1109/FOCS.2011.46.

[23] Moran Feldman, Christopher Harshaw, and Amin Karbasi. Greed is good: Near-optimal submodular maximization via greedy optimization. In *COLT*, pages 758–784, 2017. URL http://proceedings.mlr.press/v65/feldman17b.html.

[24] M. Fisher, G. Nemhauser, and L. Wolsey. An analysis of approximations for maximizing submodular set functions–II. *Mathematical Programming*, 8:73–87, 1978.

[25] Mehrdad Ghadiri, Richard Santiago, and F. Bruce Shepherd. A parameterized family of meta-submodular functions. *CoRR*, abs/2006.13754, 2020. URL https://arxiv.org/abs/2006.13754.

[26] Mehrdad Ghadiri, Richard Santiago, and F. Bruce Shepherd. Beyond submodular maximization via one-sided smoothness. In Dániel Marx, editor, *SODA*, pages 1006–1025. SIAM, 2021. doi: 10.1137/1.9781611976465.63. URL https://doi.org/10.1137/1.9781611976465.63.

[27] Anupam Gupta, Aaron Roth, Grant Schoenebeck, and Kunal Talwar. Constrained non-monotone submodular maximization: Offline and secretary algorithms. In *WINE*, pages 246–257, 2010. doi: 10.1007/978-3-642-17572-5_20. URL https://doi.org/10.1007/978-3-642-17572-5_20.

[28] F. Maxwell Harper and Joseph A. Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (TiiS)*, 5(4):1–19, 2015.

[29] Rishabh K. Iyer, Stefanie Jegelka, and Jeff A. Bilmes. Curvature and optimal algorithms for learning and minimizing submodular functions. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2742–2750, 2013.

[30] Rishabh Krishnan Iyer. *Submodular optimization and machine learning: Theoretical results, unifying and scalable algorithms, and applications*. PhD thesis, University of Washington, 2015.

[31] Ehsan Kazemi, Shervin Minaee, Moran Feldman, and Amin Karbasi. Regularized submodular maximization at scale. In *ICML*, pages 5356–5366. PMLR, 2021.

[32] Katrin Kirchhoff and Jeff Bilmes. Submodularity for data selection in machine translation. In *EMNLP*, pages 131–141, 2014.

[33] Alex Krizhevsky. Learning multiple layers of features from tiny images. Master's thesis, University of Toronto, 2009.

[34] Alan Kuhnle, J. David Smith, Victoria G. Crawford, and My T. Thai. Fast maximization of non-submodular, monotonic functions on the integer lattice. In Jennifer G. Dy and Andreas Krause, editors, *ICML*, pages 2791–2800. PMLR, 2018. URL `http://proceedings.mlr.press/v80/kuhnle18a.html`.

[35] Jon Lee, Vahab S. Mirrokni, Viswanath Nagarajan, and Maxim Sviridenko. Non-monotone submodular maximization under matroid and knapsack constraints. In Michael Mitzenmacher, editor, *STOC*, pages 323–332. ACM, 2009. doi: 10.1145/1536414.1536459. URL `https://doi.org/10.1145/1536414.1536459`.

[36] Qi Lei, Lingfei Wu, Pin-Yu Chen, Alex Dimakis, Inderjit S. Dhillon, and Michael J. Witbrock. Discrete adversarial attacks and submodular optimization with applications to text classification. In Ameet Talwalkar, Virginia Smith, and Matei Zaharia, editors, *MLSys*. mlsys.org, 2019. URL `https://proceedings.mlsys.org/book/284.pdf`.

[37] Hui Lin and Jeff Bilmes. Multi-document summarization via budgeted maximization of submodular functions. In *Human Language Technologies (HLT)*, pages 912–920, 2010.

[38] László Lovász. Submodular functions and convexity. In A. Bachem, M. Grötschel, and B. Korte, editors, *Mathematical Programming: the State of the Art*, pages 235–257. Springer, 1983.

[39] Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrák, and Andreas Krause. Lazier than lazy greedy. In *AAAI*, pages 1812–1818, 2015. URL `http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9956`.

[40] Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, and Amin Karbasi. Fast constrained submodular maximization: Personalized data summarization. In *ICML*, pages 1358–1367. PMLR, 2016.

[41] Marko Mitrovic, Ehsan Kazemi, Morteza Zadimoghaddam, and Amin Karbasi. Data summarization at scale: A two-stage submodular approach. In *ICML*, pages 3596–3605. PMLR, 2018.

[42] Marko Mitrovic, Ehsan Kazemi, Moran Feldman, Andreas Krause, and Amin Karbasi. Adaptive sequence submodularity. *NeurIPS*, 32:5352–5363, 2019.

[43] G. L. Nemhauser and L. A. Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Mathematics of Operations Research*, 3(3):177–188, 1978.

[44] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions–I. *Mathematical Programming*, 14:265–294, 1978.

[45] Ashkan Norouzi-Fard, Jakub Tarnawski, Slobodan Mitrovic, Amir Zandieh, Aidasadat Mousavifar, and Ola Svensson. Beyond 1/2-approximation for submodular maximization on massive data streams. In *ICML*, pages 3829–3838. PMLR, 2018.

[46] Shayan Oveis Gharan and Jan Vondrák. Submodular maximization by simulated annealing. In Dana Randall, editor, *SODA*, pages 1098–1116. SIAM, 2011. doi: 10.1137/1.9781611973082.83. URL `https://doi.org/10.1137/1.9781611973082.83`.

[47] Alexander Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*. Springer, 2003.

[48] Maxim Sviridenko, Jan Vondrák, and Justin Ward. Optimal approximation for submodular and supermodular optimization with bounded curvature. *Math. Oper. Res.*, 42(4):1197–1218, 2017. doi: 10.1287/moor.2016.0842. URL `https://doi.org/10.1287/moor.2016.0842`.

[49] Sebastian Tschiatschek, Rishabh K Iyer, Haochen Wei, and Jeff A Bilmes. Learning mixtures of submodular functions for image collection summarization. In *NeurIPS*, pages 1413–1421, 2014.

[50] Sebastian Tschiatschek, Adish Singla, and Andreas Krause. Selecting sequences of items via submodular maximization. In Satinder P. Singh and Shaul Markovitch, editors, *AAAI*, pages 2667–2673. AAAI Press, 2017. URL `http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14898`.

[51] Jan Vondrák. Symmetry and approximability of submodular maximization problems. *SIAM J. Comput.*, 42(1):265–304, 2013. doi: 10.1137/110832318. URL `https://doi.org/10.1137/110832318`.

[52] Wei Xia, Juan-Carlos Vera, and Luis F. Zuluaga. Globally solving nonconvex quadratic programs via linear integer programming techniques. *INFORMS J. Comput.*, 32(1):40–56, 2020. doi: 10.1287/ijoc.2018.0883. URL `https://doi.org/10.1287/ijoc.2018.0883`.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] Most of them in the Appendix
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] in the Appendix
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Additional Related Work

Lin and Bilmes [37] described an algorithm that takes advantages of a continuous partial monotonicity property, but unlike the monotonicity ratio, their property was defined in terms of the particular submodular objective they were interested in. More recently, Cui et al. [14] considered a weaker, but still binary, version of monotonicity called weak-monotonicity.

B Proof of Lemma 2.1

In this section we prove Lemma 2.1.

Lemma 2.1. *Let $f\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ be a non-negative m-monotone submodular function. For every deterministic set $O \subseteq \mathcal{N}$ and random set $D \subseteq \mathcal{N}$, $\mathbb{E}[f(O \cup D)] \geq (1 - (1-m) \cdot \max_{u \in \mathcal{N}} \Pr[u \in D]) \cdot f(O)$.*

To prove this lemma we first have to define the Lovász extension of set functions. The *Lovász extension* of a set function $f\colon 2^{\mathcal{N}} \to \mathbb{R}$ is a function $\hat{f}\colon [0,1]^{\mathcal{N}} \to \mathbb{R}$ defined as follows. For every vector $\mathbf{x} \in [0,1]^{\mathcal{N}}$,

$$\hat{f}(\mathbf{x}) = \int_0^1 f(T_\lambda(\mathbf{x}))d\lambda \ ,$$

where $T_\lambda(\mathbf{x}) \triangleq \{u \in \mathcal{N} \mid x_u \geq \lambda\}$. The Lovász extension of a submodular function is known to be convex. More important for us is the following known lemma regarding this extension. This lemma stems from an equality, proved by Lovász [38], between the Lovász extension of a submodular function and another extension known as the convex closure.

Lemma B.1. *Let $f\colon 2^{\mathcal{N}} \to \mathbb{R}$ be a submodular function, and let $\hat{f}$ be its Lovász extension. For every $\mathbf{x} \in [0,1]^{\mathcal{N}}$ and random set $D_{\mathbf{x}} \subseteq \mathcal{N}$ obeying $\Pr[u \in D_{\mathbf{x}}] = x_u$ for every $u \in \mathcal{N}$ (i.e., the marginals of $D_{\mathbf{x}}$ agree with $\mathbf{x}$), $\hat{f}(\mathbf{x}) \leq \mathbb{E}[f(D_{\mathbf{x}})]$.*

Using the last lemma, we can now prove Lemma 2.1.

Proof of Lemma 2.1. Let $\mathbf{x}$ be the vector of marginals of $O \cup D$, i.e., $x_u = \Pr[u \in O \cup D]$ for every $u \in \mathcal{N}$. Then, by Lemma B.1,

$$\begin{aligned}
\mathbb{E}[f(O \cup D)] \geq \hat{f}(\mathbf{x}) &= \int_0^1 f(T_\lambda(\mathbf{x}))d\lambda \\
&= \int_0^{\max_{u \in \mathcal{N}} \Pr[u \in D]} f(T_\lambda(\mathbf{x}))d\lambda + \int_{\max_{u \in \mathcal{N}} \Pr[u \in D]}^1 f(T_\lambda(\mathbf{x}))d\lambda \\
&= \int_0^{\max_{u \in \mathcal{N}} \Pr[u \in D]} f(O \cup T_\lambda(\mathbf{x}))d\lambda + (1 - \max_{u \in \mathcal{N}} \Pr[u \in D]) \cdot f(O) \ ,
\end{aligned}$$

where the last equality holds since the elements of O appear in $T_\lambda(\mathbf{x})$ for every $\lambda \in [0,1]$, and no other element appears in $T_\lambda(\mathbf{x})$ when $\lambda > \Pr[u \in D]$. Using the definition of the monotonicity ratio, the expression $f(O \cup T_\lambda(\mathbf{x}))$ on the rightmost side of the previous equation can be lower bounded by $m \cdot f(O)$, which yields

$$\begin{aligned}
\mathbb{E}[f(O \cup D)] &\geq \int_0^{\max_{u \in \mathcal{N}} \Pr[u \in D]} m \cdot f(O)d\lambda + (1 - \max_{u \in \mathcal{N}} \Pr[u \in D]) \cdot f(O) \\
&= m \cdot \max_{u \in \mathcal{N}} \Pr[u \in D] \cdot f(O) + (1 - \max_{u \in \mathcal{N}} \Pr[u \in D]) \cdot f(O) \\
&= (1 - (1-m) \cdot \max_{u \in \mathcal{N}} \Pr[u \in D]) \cdot f(O) \ .
\end{aligned}$$

□

C Proofs of Section 3

In this section we give the proofs of Section 3.

C.1 Proof of the first part of Theorem 3.1

In this section we prove the first part of Theorem 3.1, which is restated by the following proposition.

Proposition C.1. *There exists a polynomial time algorithm that obtains an approximation ratio of* $\max\{m, (2+m)/4\}$ *for unconstrained submodular maximization.*

Proof. The analysis of the double greedy algorithm by Buchinder et al. [7] shows that this algorithm outputs a solution set of expected value at least

$$\frac{2f(OPT) + f(\varnothing) + f(\mathcal{N})}{4} \ ,$$

where OPT is an arbirary optimal solution. In general, this only shows $1/2$-approximation, as is claimed by [7]. However, when the monotonicity ratio is taken into account,

$$\begin{aligned}\frac{2f(OPT) + f(\varnothing) + f(\mathcal{N})}{4} &\geq \frac{2f(OPT) + f(\mathcal{N})}{4} \\ &\geq \frac{2f(OPT) + m \cdot f(OPT)}{4} = \frac{2+m}{4} \cdot f(OPT) \ ,\end{aligned}$$

where the first inequality follows from the non-negativity of f, and the second inequality holds since OPT is a subset of $\mathcal{N}$. Hence, the double greedy algorithm of Buchinder et al. [7] is a polynomial time algorithm guaranteeing $(2+m)/4$-approximation.

To complete the proof of the proposition, we note that the trivial algorithm that always outputs the set $\mathcal{N}$ has an approximation ratio of at least m because $f(\mathcal{N}) \geq m \cdot f(OPT)$. Hence, outputting the better solution among $\mathcal{N}$ and the output of double greedy guarantees $\max\{m, (2+m)/4\}$-approximation. □

C.2 Proof of Theorem 3.2

In this section, we show how the proof of the symmetry gap technique due to Vondrák [51] can be adapted to prove Theorem 3.2. Let us begin the section by stating some definitions that are required in order to formally state Theorem 3.2.

Definition C.2. [Strong symmetry] Consider a non-negative submodular function f and a collection $\mathcal{F} \subseteq 2^{\mathcal{N}}$ of feasible sets. The problem $\max\{f(S) \mid S \in \mathcal{F}\}$ is strongly symmetric with respect to a group of permutations $\mathcal{G}$ on $\mathcal{N}$, if (1) $f(S) = f(\sigma(S))$ for all $S \subseteq \mathcal{N}$ and $\sigma \in \mathcal{G}$, and (2) $S \in \mathcal{F} \iff S' \in \mathcal{F}$ whenever $\mathbb{E}_{\sigma\in\mathcal{G}}[\mathbf{1}_{\sigma(S)}] = \mathbb{E}_{\sigma\in\mathcal{G}}[\mathbf{1}_{\sigma(S')}]$, where $\mathbb{E}_{\sigma\in\mathcal{G}}$ represents the expectation over picking σ uniformly at random out of $\mathcal{G}$.

Definition C.3. [Symmetry gap] Consider a non-negative submodular function f and a collection $\mathcal{F} \subseteq 2^{\mathcal{N}}$ of feasible sets. Let $F(x)$ be the multilinear extension of f and $P(\mathcal{F}) \subseteq [0,1]^{\mathcal{N}}$ be the convex hull of $\mathcal{F}$. Then, if the problem $\max\{f(S) \mid S \in \mathcal{F}\}$ is strongly symmetric with respect to a group $\mathcal{G}$ of permutation, then its symmetry is defined as

$$\frac{\max\{F(\bar{\mathbf{x}}) \mid \mathbf{x} \in P(\mathcal{F})\}}{\max\{F(\mathbf{x}) \mid \mathbf{x} \in P(\mathcal{F})\}} \ ,$$

where $\bar{\mathbf{x}} \triangleq \mathbb{E}_{\sigma\in\mathcal{G}}[\sigma(\mathbf{x})]$.

Definition C.4. [Refinement] Consider a set $\mathcal{F} \subseteq 2^{\mathcal{N}}$, and let X be some set. We say that $\tilde{\mathcal{F}} \subseteq 2^{\mathcal{N}\times X}$ is a refinement of $\mathcal{F}$ if

$$\tilde{F} = \left\{ S \subseteq \mathcal{N} \times X \;\middle|\; \mathbf{x} \in P(\mathcal{F}), \text{ where } x_u = \tfrac{|S \cap (\{u\}\times X)|}{|X|} \text{ for every } u \in \mathcal{N} \right\} \ .$$

Using the above definitions, we can now formally state the theorem that we want to prove.

Theorem 3.2. *Consider a non-negative m-monotone submodular function f and a collection $\mathcal{F} \subseteq 2^{\mathcal{N}}$ of feasible sets such that the problem $\max\{f(S) \mid S \in \mathcal{F}\}$ is strongly symmetric with respect to some group $\mathcal{G}$ of permutations over $\mathcal{N}$ and has a symmetry gap γ. Let $\mathcal{C}$ be the class of problems $\max\{\tilde{f}(S) \mid S \in \tilde{F}\}$ in which $\tilde{f}$ is a non-negative m-monotone submodular function, and $\tilde{F}$ is a refinement of F. Then, for every $\varepsilon > 0$, any (even randomized) $(1+\varepsilon)\gamma$-approximation algorithm for the class $\mathcal{C}$ would require exponentially many value queries to $\tilde{f}$.*

The crux of the symmetry gap technique is two lemmata due to [51] that we restate below. Lemma C.5 shows that given a non-negative set function f, one can obtain from it two continuous versions: a continuous version $\hat{F}$ that resembles f itself, and a continuous version $\hat{G}$ that resembles a symmetrized version of f. Distinguishing between $\hat{F}$ and $\hat{G}$ is difficult, however, this does not translate into an hardness for discrete problems since $\hat{F}$ and $\hat{G}$ are continuous. Therefore, Vondrák [51] proved also Lemma C.6, which shows how these continuous functions can be translated back into set functions with appropriate properties.

Lemma C.5 (Lemma 3.2 of [51])**.** *Consider a function $f\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ invariant under a group of permutations $\mathcal{G}$ on the ground set $\mathcal{N}$. Let $F(\mathbf{x})$ be the multilinear extension of F, define $\bar{x} = \mathbb{E}_{\sigma\in\mathcal{G}}[\mathbf{1}_{\sigma(\mathbf{x})}]$ and fix any $\varepsilon > 0$. Then, there is $\delta > 0$ and functions $\hat{F}, \hat{G}\colon [0,1]^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (which are also symmetric with respect to $\mathcal{G}$), satisfying the following:*

1. *For all $\mathbf{x} \in [0,1]^{\mathcal{N}}$, $\hat{G}(\mathbf{x}) = \hat{F}(\bar{\mathbf{x}})$.*
2. *For all $\mathbf{x} \in [0,1]^{\mathcal{N}}$, $|\hat{F}(\mathbf{x}) - F(\mathbf{x})| \leq \varepsilon$.*
3. *Whenever $\|\mathbf{x} - \bar{\mathbf{x}}\|_2 \leq \delta$, $\hat{F}(\mathbf{x}) = \hat{G}(\mathbf{x})$ and the value depends only on $\bar{\mathbf{x}}$.*
4. *The first partial derivatives of $\hat{F}$ and $\hat{G}$ are absolutely continuous.*
5. *If f is monotone, then, for every element $u \in \mathcal{N}$, $\frac{\partial \hat{F}}{\partial x_u} \geq 0$ and $\frac{\partial \hat{G}}{\partial x_u} \geq 0$ everywhere.*
6. *If f is submodular then, for every two elements $u, v \in \mathcal{N}$, $\frac{\partial^2 \hat{F}}{\partial x_u \partial x_v} \leq 0$ and $\frac{\partial^2 \hat{G}}{\partial x_u \partial x_v} \leq 0$ almost everywhere.*

Lemma C.6 (Lemma 3.1 of [51])**.** *Let n be a positive integer, and let $F\colon [0,1]^{\mathcal{N}} \to \mathbb{R}$ and $X = [n]$. If we define $f\colon 2^{\mathcal{N}\times X} \to \mathbb{R}_{\geq 0}$ so that $f(S) = F(\mathbf{x})$, where $x_u = \frac{1}{n}|S \cap (\{u\} \times X)|$. Then,*

1. *if $\frac{\partial F}{\partial x_u} \geq 0$ everywhere for each element $u \in \mathcal{N}$, then f is monotone,*
2. *and if the first partial derivatives of F are absolutely continuous and $\frac{\partial^2 F}{\partial x_u \partial x_v} \leq 0$ almost everywhere for all elements $u, v \in \mathcal{N}$, then f is submodular.*

One can note that the above lemmata have the property that if the function f plugged into Lemma C.5 is monotone, then the discrete functions obtained by applying Lemma C.6 to the functions $\hat{F}$ and $\hat{G}$ are also monotone. This is the reason that the framework of [51] applies to monotone functions (as well as general, not necessarily monotone, functions). Therefore, to get the proof of [51] to yield Theorem 3.2, it suffices to prove the following two modified versions of Lemmata C.5 and C.6. These modified versions preserve m-monotonicity for any $m \in [0,1]$, rather than just standard monotonicity.

Lemma C.7 (modified version of Lemma C.5)**.** *Consider a function $f\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ that is m-monotone and invariant under a group of permutations $\mathcal{G}$ on the ground set $\mathcal{N}$. Let $F(\mathbf{x})$ be the multilinear extension of F, define $\bar{x} = \mathbb{E}_{\sigma\in\mathcal{G}}[\mathbf{1}_{\sigma(\mathbf{x})}]$ and fix any $\varepsilon > 0$. Then, there is $\delta > 0$ and functions $\hat{F}, \hat{G}\colon [0,1]^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ (which are also symmetric with respect to $\mathcal{G}$), satisfying the following:*

1. *For all $\mathbf{x} \in [0,1]^{\mathcal{N}}$, $\hat{G}(\mathbf{x}) = \hat{F}(\bar{\mathbf{x}})$.*
2. *For all $\mathbf{x} \in [0,1]^{\mathcal{N}}$, $|\hat{F}(\mathbf{x}) - F(\mathbf{x})| \leq \varepsilon$.*
3. *Whenever $\|\mathbf{x} - \bar{\mathbf{x}}\|_2 \leq \delta$, $\hat{F}(\mathbf{x}) = \hat{G}(\mathbf{x})$ and the value depends only on $\bar{\mathbf{x}}$.*
4. *For every two vectors $\mathbf{x}, \mathbf{y} \in [0,1]^{\mathcal{N}}$ obeying $\mathbf{x} \leq \mathbf{y}$, $m \cdot F(\mathbf{x}) \leq F(\mathbf{y})$.*
5. *If f is submodular then, for every two elements $u, v \in \mathcal{N}$, $\frac{\partial^2 \hat{F}}{\partial x_u \partial x_v} \leq 0$ and $\frac{\partial^2 \hat{G}}{\partial x_u \partial x_v} \leq 0$ almost everywhere.*

Lemma C.8 (modified version of Lemma C.6)**.** *Let n be a positive integer, and let $F\colon [0,1]^{\mathcal{N}} \to \mathbb{R}$ and $X = [n]$. If we define $f\colon 2^{\mathcal{N}\times X} \to \mathbb{R}_{\geq 0}$ so that $f(S) = F(\mathbf{x})$, where $x_u = \frac{1}{n}|S \cap (\{u\} \times X)|$. Then,*

1. *if for some value $m \in [0,1]$ the inequality $m \cdot F(\mathbf{x}) \leq F(\mathbf{y})$ holds for any two vectors $\mathbf{x}, \mathbf{y} \in [0,1]^{\mathcal{N}}$ that obey $\mathbf{x} \leq \mathbf{y}$, then f is m-monotone,*

2. *and if the first partial derivatives of F are absolutely continuous and $\frac{\partial^2 F}{\partial x_u \partial x_v} \leq 0$ almost everywhere for all elements $u, v \in \mathcal{N}$, then f is submodular.*

The proof of Lemma C.7 is quite long and appears below. However, before getting to this proof, let first give the much simpler proof of Lemma C.8.

Proof of Lemma C.8. The second point in Lemma C.8 follows immediately from Lemma C.6, so we concentrate on proving the first point. In other words, we assume that $m \cdot F(\mathbf{x}) \leq F(\mathbf{y})$ for every two vectors $\mathbf{x}, \mathbf{y} \in [0,1]^{\mathcal{N}}$ obeying $\mathbf{x} \leq \mathbf{y}$, and we need to show that $m \cdot f(S) \leq f(T)$ for every two sets $S \subseteq T \subseteq \mathcal{N}$.

Let us define two vectors $\mathbf{x}^{(S)}, \mathbf{x}^{(T)} \subseteq [0,1]^{\mathcal{N}}$ as follows. For every $u \in \mathcal{N}$,

$$x_u^{(S)} = \frac{1}{n}|S \cap (\{u\} \times X)| \quad \text{and} \quad x_u^{(T)} = \frac{1}{n}|T \cap (\{u\} \times X)| \ .$$

Since $S \subseteq T$, we get $\mathbf{x}^{(S)} \leq \mathbf{x}^{(T)}$, which implies $m \cdot F(\mathbf{x}^{(S)}) \leq F(\mathbf{x}^{(T)})$; and the last inequality proves the lemma since $f(S) = F(\mathbf{x}^{(S)})$ and $f(T) = F(\mathbf{x}^{(T)})$ by the definition of f. $\square$

We now get to the proof of Lemma C.7. We use in this proof functions $\hat{F}$ and $\hat{G}$ that are similar to the ones constructed by Vondrák [51] in the proof of Lemma C.5. Specifically, like in the proof of [51], we define

$$\hat{G}(\mathbf{x}) = G(\mathbf{x}) + 256M|\mathcal{N}|\alpha J(\mathbf{x}) \ ,$$

where M is the maximum value that the function f takes on any set, G is a symmetrized version of the multilinear extension F of f defined as $G(\mathbf{x}) = F(\bar{\mathbf{x}})$, $J(\mathbf{x}) \triangleq |\mathcal{N}|^2 + 3|\mathcal{N}| \cdot \sum_{u \in \mathcal{N}} x_u - \left(\sum_{u \in \mathcal{N}} x_u\right)^2$, and α is a positive value that is independent of $\mathbf{x}$. Similarly, the function $\hat{F}$ was defined by Vondrák [51] as

$$\hat{F}(\mathbf{x}) = \tilde{F}(\mathbf{x}) + 256M|\mathcal{N}|\alpha J(\mathbf{x}) \ ,$$

where the function $\tilde{F}$ interpolates between the multilinear extension F of f and its symmetrized version G, and is given by

$$\tilde{F}(\mathbf{x}) = (1 - \phi(D(\mathbf{x}))) \cdot F(\mathbf{x}) + \phi(D(\mathbf{x})) \cdot G(\mathbf{x}) \ .$$

Here, $D(\mathbf{x}) \triangleq \|\mathbf{x} - \bar{\mathbf{x}}\|_2^2$, and $\phi \colon \mathbb{R}_{\geq 0} \to [0,1]$ is a function which is defined using the following lemma.

Lemma C.9 (Lemma 3.7 of [51])**.** *For any $\alpha, \beta > 0$, there is $\delta > (0, \beta)$ and a function $\phi \colon \mathbb{R}_{\geq 0} \to [0,1]$ with an absolutely continuous first derivative such that*

- *For $t \in [0, \delta]$, $\phi(t) = 1$.*
- *For $t \geq \beta$, $\phi(t) < e^{-1/\alpha}$.*
- *For all $t \geq 0$, $|t\phi'(t)| \leq 4\alpha$.*
- *For almost all $t \geq 0$, $|t^2\phi''(t)| \leq 10\alpha$.*

Vondrák [51] proved that the above functions $\hat{F}$ and $\hat{G}$ have all the properties guaranteed by Lemma C.5 for the δ whose existence is guaranteed by Lemma C.9 when the values of α and β are set to be $\alpha = \frac{\varepsilon}{2000M|\mathcal{N}|^3}$ and $\beta = \frac{\varepsilon}{16M|\mathcal{N}|}$. Moreover, the proof of [51] continues to work as long as $\alpha \leq \frac{\varepsilon}{2000M|\mathcal{N}|^3}$ and $\beta \leq \frac{\varepsilon}{16M|\mathcal{N}|}$. Therefore, we assume below that $\alpha = \min\{1, \frac{\varepsilon}{2000M|\mathcal{N}|^3}\}$ and $\beta = \min\{\alpha^2, \frac{\varepsilon}{16M|\mathcal{N}|}\}$, and we prove only the part of Lemma C.7 that is not stated in the guarantees of Lemma C.5, which is Property 4 of the lemma. We begin by showing that the function $\hat{G}$ indeed has this property.

Lemma C.10. *For every two vectors $\mathbf{x}, \mathbf{y} \in [0,1]^{\mathcal{N}}$ obeying $\mathbf{x} \leq \mathbf{y}$, $m \cdot \hat{G}(\mathbf{x}) \leq \hat{G}(\mathbf{y})$.*

Proof. Consider the random sets $\mathtt{R}(\bar{\mathbf{x}})$ and $\mathtt{R}(\bar{\mathbf{y}})$. Since $\bar{\mathbf{x}} \leq \bar{\mathbf{y}}$, the set $\mathtt{R}(\bar{\mathbf{y}})$ stochastically dominates $\mathtt{R}(\bar{\mathbf{x}})$. In other words, one can correlate the randomness of these sets in a way that does not alter their distributions, but guarantees that the inclusion $\mathtt{R}(\bar{\mathbf{x}}) \subseteq \mathtt{R}(\bar{\mathbf{y}})$ holds deterministically. Assuming this done, we get

$$m \cdot G(\mathbf{x}) = m \cdot F(\bar{\mathbf{x}}) = m \cdot \mathbb{E}[f(\mathtt{R}(\bar{\mathbf{x}}))] \leq \mathbb{E}[f(\mathtt{R}(\bar{\mathbf{y}}))] = F(\bar{\mathbf{y}}) = G(\mathbf{y}) \ , \tag{2}$$

where the inequality follows from the linearity of the expectation and the m-monotonicity of f.

Observe now that for every element $u \in \mathcal{N}$, the partial derivative of J with respect to z_u at any point $\mathbf{z} \in [0,1]^{\mathcal{N}}$ is

$$\frac{\partial J(\mathbf{z})}{\partial z_u} = 3|\mathcal{N}| - 2\sum_{v \in \mathcal{N}} z_v \geq |\mathcal{N}| \geq 0 \ .$$

Hence, the inequality $\mathbf{x} \leq \mathbf{y}$ implies $m \cdot J(\mathbf{x}) \leq J(\mathbf{x}) \leq J(\mathbf{y})$. Together with Inequality (2), this implies the lemma. □

One can observe that the arguments used to prove Inequality(2) in the proof of the last lemma also show that $m \cdot F(\mathbf{x}) \leq F(\mathbf{y})$, which is a fact that we use below. However, proving that $\hat{F}$ also has this property (and therefore, obeys Property 4 of Lemma C.7) is more involved. As a first step towards this goal, we bound the gradient of

$$\tilde{F}(\mathbf{x}) - F(\mathbf{x}) = \phi(D(\mathbf{x})) \cdot [G(\mathbf{x}) - F(\mathbf{x})] \ .$$

The following lemma does that in the regime in which $D(\mathbf{x})$ is small, and the next lemma handles the other regime.

Lemma C.11. *For every element $u \in \mathcal{N}$ and vector $\mathbf{x} \in [0,1]^{\mathcal{N}}$ obeying $D(\mathbf{x}) \leq \beta$, the absolute value of the partial derivative $\frac{\partial\{\phi(D(\mathbf{x}))\cdot[G(\mathbf{x})-F(\mathbf{x})]\}}{\partial x_u}$ is at most $72\sqrt{\beta}M|\mathcal{N}| \leq 72\alpha M|\mathcal{N}|$.*

Proof. Observe that

$$\begin{aligned}\frac{\partial\{\phi(D(\mathbf{x})) \cdot [G(\mathbf{x}) - F(\mathbf{x})]\}}{\partial x_u} &= \phi'(D(\mathbf{x})) \cdot \frac{\partial D(\mathbf{x})}{\partial x_u} \cdot [G(\mathbf{x}) - F(\mathbf{x})] \\ &\quad + \phi(D(\mathbf{x})) \cdot \left[\frac{\partial G(\mathbf{x})}{\partial x_u} - \frac{\partial F(\mathbf{x})}{\partial x_u}\right] \ .\end{aligned}$$

To use this equation to bound the absolute value of the left hand side, we need to make some observations. First, Lemma 3.6 of [51] shows that $\|\nabla D(\mathbf{x})\|_2 = 2\sqrt{D(\mathbf{x})}$, which implies

$$\frac{\partial D(\mathbf{x})}{\partial x_u} \leq \|\nabla D(\mathbf{x})\|_2 = 2\sqrt{D(\mathbf{x})} \ .$$

Additionally, Lemma 3.5 of [51] shows that $|G(\mathbf{x}) - F(\mathbf{x})| \leq 8M|\mathcal{N}| \cdot D(\mathbf{x})$, and therefore,

$$\begin{aligned}\left|\phi'(D(\mathbf{x})) \cdot \frac{\partial D(\mathbf{x})}{\partial x_u} \cdot [G(\mathbf{x}) - F(\mathbf{x})]\right| &\leq |\phi'(D(\mathbf{x}))| \cdot \left|\frac{\partial D(\mathbf{x})}{\partial x_u}\right| \cdot |G(\mathbf{x}) - F(\mathbf{x})| \\ &\leq |\phi'(D(\mathbf{x}))| \cdot 2\sqrt{D(\mathbf{x})} \cdot 8M|\mathcal{N}| \cdot D(\mathbf{x}) \\ &= |D(\mathbf{x}) \cdot \phi'(D(\mathbf{x}))| \cdot 16M|\mathcal{N}| \cdot \sqrt{D(\mathbf{x})} \\ &\leq 64\alpha\sqrt{\beta}M|\mathcal{N}| \ ,\end{aligned}$$

where the second inequality follows from Lemma C.9 and our assumption that $D(\mathbf{x}) \leq \beta$.

We now observe that

$$\begin{aligned}&\left|\phi(D(\mathbf{x})) \cdot \left[\frac{\partial G(\mathbf{x})}{\partial x_u} - \frac{\partial F(\mathbf{x})}{\partial x_u}\right]\right| = \phi(D(\mathbf{x})) \cdot \left|\frac{\partial G(\mathbf{x})}{\partial x_u} - \frac{\partial F(\mathbf{x})}{\partial x_u}\right| \\ &\leq \phi(D(\mathbf{x})) \cdot \|\nabla G(\mathbf{x}) - \nabla F(\mathbf{x})\|_2 \leq \phi(D(\mathbf{x})) \cdot 8M|\mathcal{N}| \cdot \sqrt{D(\mathbf{x})} \leq 8\sqrt{\beta}M|\mathcal{N}| \ ,\end{aligned}$$

where the second inequality holds since Lemma 3.5 of [51] shows that $\|\nabla G(\mathbf{x}) - F(\mathbf{x})\|_2 \leq 8M|\mathcal{N}| \cdot \sqrt{D(\mathbf{x})}$; and the last inequality holds by our assumption that $D(\mathbf{x}) \leq \beta$ and by recalling that the range of ϕ is $[0,1]$.

Combining all the above yields

$$\left|\frac{\partial\{\phi(D(\mathbf{x}))\cdot[G(\mathbf{x})-F(\mathbf{x})]\}}{\partial x_u}\right| \leq 64\alpha\sqrt{\beta}M|\mathcal{N}| + 8\sqrt{\beta}M|\mathcal{N}| \leq 72\sqrt{\beta}M|\mathcal{N}| \ ,$$

where the second inequality holds since $\alpha \leq 1$. □

Lemma C.12. *For every element $u \in \mathcal{N}$ and vector $\mathbf{x} \in [0,1]^{\mathcal{N}}$ obeying $D(\mathbf{x}) \geq \beta$, the absolute value of the partial derivative $\frac{\partial\{\phi(D(\mathbf{x}))\cdot[G(\mathbf{x})-F(\mathbf{x})]\}}{\partial x_u}$ is at most $72\alpha M|\mathcal{N}|^{3/2}$.*

Proof. Repeating the proof of Lemma C.11, except for the use of the inequality $D(\mathbf{x}) \leq \beta$ (which does not hold in the current lemma) and the inequality $\phi(\mathbf{x}) \leq 1$ (which too weak for our current purpose), we get

$$\left|\phi(D(\mathbf{x}))\cdot\left[\frac{\partial G(\mathbf{x})}{\partial x_u} - \frac{\partial F(\mathbf{x})}{\partial x_u}\right]\right| \leq 64\alpha M|\mathcal{N}|\cdot\sqrt{D(\mathbf{x})} + |\phi(D(\mathbf{x}))|\cdot 8M|\mathcal{N}|\cdot\sqrt{D(\mathbf{x})} \ .$$

The expression $\phi(D(\mathbf{x}))$ can be upper bounded by $e^{-1/\alpha} \leq \alpha$ by Lemma C.9. Also, $D(\mathbf{x}) = \|\mathbf{x}-\bar{\mathbf{x}}\|_2^2 \leq |\mathcal{N}|$. The lemma now follows by plugging these two upper bounds into the previous inequality. □

Corollary C.13. *For every element $u \in \mathcal{N}$ and vector $\mathbf{x} \in [0,1]^{\mathcal{N}}$, the absolute value of the partial derivative $\frac{\partial\{\phi(D(\mathbf{x}))\cdot[G(\mathbf{x})-F(\mathbf{x})]\}}{\partial x_u}$ is at most $72\alpha M|\mathcal{N}|^{3/2}$.*

The last corollary implies that $\tilde{F}$ can be presented as the sum of F and a component that changes slowly. Therefore, if we add to $\tilde{F}$ a function that increases quickly enough (as is done to define $\hat{F}$), then we should get a function that can be represented as F plus a monotone component. This is the intuition formalized in the proof of the next lemma.

Lemma C.14. *The function $\hat{F}(\mathbf{x}) - F(\mathbf{x})$ has non-negative partial derivatives for every $\mathbf{x} \in [0,1]^{\mathcal{N}}$.*

Proof. By the definition of $\hat{F}(\mathbf{x})$,

$$\hat{F}(\mathbf{x}) - F(\mathbf{x}) = \tilde{F}(\mathbf{x}) - F(\mathbf{x}) + 256M|\mathcal{N}|\alpha J(\mathbf{x}) \ .$$

By Corollary C.13 and the observation that all the partial derivatives of $J(\mathbf{x})$ are at least $|\mathcal{N}|$ (see the proof of Lemma C.10), the last equality implies, for every element $u \in \mathcal{N}$,

$$\frac{\partial[\hat{F}(\mathbf{x}) - F(\mathbf{x})]}{\partial x_u} \geq -72\alpha M|\mathcal{N}|^{3/2} + 256\alpha M|\mathcal{N}|^2 \geq 0 \ .$$ □

We are now ready to show that $\hat{F}$ obeys Property 4 of Lemma C.7.

Lemma C.15. *For every two vectors $\mathbf{x}, \mathbf{y} \in [0,1]^{\mathcal{N}}$ obeying $\mathbf{x} \leq \mathbf{y}$, $m \cdot \hat{F}(\mathbf{x}) \leq \hat{F}(\mathbf{y})$.*

Proof. Note that $\overline{\mathbf{1}_\varnothing} = \mathbf{1}_\varnothing$, which implies that $G(\mathbf{1}_\varnothing) = F(\mathbf{1}_\varnothing)$, and therefore,

$$\tilde{F}(\mathbf{1}_\varnothing) - F(\mathbf{1}_\varnothing) = \phi(D(\mathbf{1}_\varnothing))\cdot[G(\mathbf{1}_\varnothing) - F(\mathbf{1}_\varnothing)] = 0 \ .$$

Plugging this observation into the definition of $\hat{F}$ now gives

$$\hat{F}(\mathbf{1}_\varnothing) - F(\mathbf{1}_\varnothing) = \tilde{F}(\mathbf{1}_\varnothing) - F(\mathbf{1}_\varnothing) + 256M|\mathcal{N}|\alpha J(\mathbf{1}_\varnothing) = 256M|\mathcal{N}|\alpha J(\mathbf{1}_\varnothing) \ .$$

Since all the first partial derivatives of $\hat{F}(\mathbf{z}) - F(\mathbf{z})$ are non-negative by Lemma C.14, the last inequality implies

$$\hat{F}(\mathbf{y}) - F(\mathbf{y}) \geq \hat{F}(\mathbf{x}) - F(\mathbf{x}) \geq 256M|\mathcal{N}|\alpha J(\mathbf{x}) \geq 0 \ .$$

Hence,

$$m\cdot\hat{F}(\mathbf{x}) \leq m\cdot[F(\mathbf{x})+\hat{F}(\mathbf{y})-F(\mathbf{y})] \leq F(\mathbf{y})+m\cdot[\hat{F}(\mathbf{y})-F(\mathbf{y})] \leq F(\mathbf{y})+[\hat{F}(\mathbf{y})-F(\mathbf{y})] = \hat{F}(\mathbf{y}) \ ,$$

where the second inequality holds by the discussion immediately after the proof of Lemma C.10, and the last inequality holds since $m \leq 1$ and $\hat{F}(\mathbf{y}) - F(\mathbf{y}) \geq 0$. □

C.3 Proof of Lemma 3.3

Lemma 3.3. *The problem* $\max\{f(S) \mid S \in \mathcal{F}\}$ *has a symmetry gap of* $\frac{1}{2-m}$.

Proof. Observe that our definition of $\mathcal{F}$ implies that $P(\mathcal{F}) = [0,1]^{\mathcal{N}}$. Therefore,

$$\max\{F(\mathbf{x}) \mid \mathbf{x} \in P(\mathcal{F})\} = \max\{F(\mathbf{x}) \mid \mathbf{x} \in [0,1]^{\mathcal{N}}\} = \max\{f(S) \mid S \subseteq \mathcal{N}\} = 1 \ , \quad (3)$$

where the second equality holds since, for every vector $\mathbf{x}$, $F(\mathbf{x})$ is a convex combination of values of f for subsets of $\mathcal{N}$; and on the other hand, for every set $S \subseteq \mathcal{N}$, $f(S) = F(\mathbf{1}_S)$.

Observe now that the definition of f implies that

$$\begin{aligned} F(\mathbf{x}) &= m[1-(1-x_u)(1-x_v)] + (1-m)\cdot[x_u(1-x_v) + x_v(1-x_u)] \\ &= x_u + x_v - x_u x_v (2-m) \ . \end{aligned}$$

Since $\bar{\mathbf{x}}$ is a vector that has the value $(x_u + x_v)/2$ in both its coordinates, if we we use the shorthand $y = (x_u + x_v)/2$, then we get

$$F(\bar{\mathbf{x}}) = 2y - (2-m)y^2 \ .$$

This expression is maximized for $y = 1/(2-m)$, and the maximum attained for this y is

$$\frac{2}{2-m} - \frac{(2-m)}{(2-m)^2} = \frac{1}{2-m} \ .$$

Since the value $y = 1/(2-m)$ is obtained, for example, when $\mathbf{x} = (y,y) \in [0,1]^{\mathcal{N}}$, the above implies

$$\max\{F(\bar{\mathbf{x}}) \mid \mathbf{x} \in P(\mathcal{F})\} = \frac{1}{2-m} \ .$$

Together with Equation (3), this implies the lemma. □

D Inapproximability and Proofs of Section 4

In this section we state and analyze the algorithms used to prove the results given in Section 4. We also state and prove in Section D.3 the inapproximability result mentioned in Section 4.

D.1 Analysis of the Greedy Algorithm

In this section we prove Theorem 4.1, which we repeat here for convenience.

Theorem 4.1. *The Greedy algorithm (Algorithm 1) has an approximation ratio of at least* $m(1 - 1/e)$ *for the problem of maximizing a non-negative* m*-monotone submodular function subject to a cardinality constraint.*

The greedy algorithm starts with an empty solution, and then augments this solution in k iterations (recall that k is the maximum cardinality allowed for a feasible solution). Specifically, in iteration i, the algorithm adds to the current solution the element u_i with the best (largest) marginal contribution with respect to the current solution—but only if this addition does not decrease the value of the solution. A formal description of the greedy algorithm appears as Algorithm 1. Note that in this description the solution of the algorithm after i iterations, for every integer $0 \le i \le n$, is denoted by A_i.

Algorithm 1: The Greedy Algorithm (f, k)

```
Let A_0 ← ∅.
for i = 1 to k do
    Let u_i be the element of N \ A_{i-1} maximizing f(u_i | A_{i-1}).
    if f(u_i | A_{i-1}) ≥ 0 then Let A_i ← A_{i-1} + u_i.
    else Let A_i ← A_{i-1}.
return A_k.
```

Our first step towards proving Theorem 4.1 is the following lemma, which lower bounds the increase in the value of $f(A_i)$ as a function of i. Specifically, the lemma shows that this increase is significant as long as there is a significant gap between between $f(A_{i-1})$ and $m \cdot f(OPT)$, where OPT is an arbitrary optimal solution.

Lemma D.1. *For every integer* $1 \leq i \leq k, f(A_i) - f(A_{i-1}) \geq k^{-1}[m \cdot f(OPT) - f(A_{i-1})]$.

Proof. We need to distinguish between two cases. Consider first the case in which $f(u_i \mid A_{i-1}) \geq 0$. In this case,

$$\begin{aligned} f(A_i) - f(A_{i-1}) = f(u_i \mid A_{i-1}) &\geq \frac{|OPT \setminus A_{i-1}|}{k} \cdot f(u_i \mid A_{i-1}) \\ &\geq \frac{|OPT \setminus A_{i-1}|}{k} \cdot \max_{u \in OPT \setminus A_{i-1}} f(u \mid A_{i-1}) \geq \frac{\sum_{u \in OPT \setminus A_{i-1}} f(u \mid A_{i-1})}{k} \\ &\geq \frac{f(OPT \cup A_{i-1}) - f(A_{i-1})}{k} \geq \frac{m \cdot f(OPT) - f(A_{i-1})}{k} \ , \end{aligned}$$

where the first inequality holds since $|OPT \setminus A_{i-1}| \leq |OPT| \leq k$ because OPT is a feasible solution, the second inequality is due to the way used by the greedy algorithm to choose the element u_i, the penulatimate inequality follows from the submodularity of f, and the last inequality holds since f is m-monotone.

Consider now the case in which $f(u_i \mid A_{i-1}) < 0$. In this case, $f(A_i) - f(A_{i-1}) = 0$ because $A_i = A_{i-1}$. Furthermore, repeating the arguments used to prove the above inequality yields

$$\begin{aligned} m \cdot f(OPT) - f(A_{i-1}) &\leq |OPT \setminus A_{i-1}| \cdot \max_{u \in OPT \setminus A_{i-1}} f(u \mid A_{i-1}) \\ &\leq |OPT \setminus A_{i-1}| \cdot \max_{u \in \mathcal{N} \setminus A_{i-1}} f(u \mid A_{i-1}) \leq 0 \ . \end{aligned}$$

□

Rearranging the last lemma, we get the following inequality.

$$m \cdot f(OPT) - f(A_i) \leq (1 - 1/k) \cdot [m \cdot f(OPT) - f(A_{i-1})] \ . \tag{4}$$

This inequality bounds the rate in which the gap between $m \cdot f(OPT)$ reduces as a function of i. This allows us to prove Theorem 4.1.

Proof of Theorem 4.1. Combining Inequality (4) for every integer $1 \leq i \leq k$ yields

$$m \cdot f(OPT) - f(A_k) \leq (1 - 1/k)^k \cdot [m \cdot f(OPT) - f(A_0)] \ .$$

Rearranging this inequality, we get

$$f(A_k) \geq m \cdot f(OPT) - m \cdot (1 - 1/k)^k \cdot [f(OPT) - f(A_0)] \geq m \cdot \left(1 - \frac{1}{e}\right) \cdot f(OPT) \ ,$$

where the last inequality follows from the non-negativity of f and the inequality $(1 - 1/k)^k \leq \frac{1}{e}$. □

D.2 Analysis of Random Greedy

In this section we prove Theorem 4.2, which we repeat here for convenience.

Theorem 4.2. *Random Greedy (Algorithm 2) has an approximation ratio of at least* $m(1 - 1/e) + (1 - m) \cdot (1/e)$ *for the problem of maximizing a non-negative* m*-monotone submodular function subject to a cardinality constraint.*

Like the standard greedy algorithm from Section D.1, the Random Greedy algorithm starts with an empty solution, and then augments it in k iterations. Specifically, in iteration i the algorithm finds a set M_i of at most k elements whose total marginal contribution with respect to the current solution is maximal. Then, at most one element of M_i is added to the algorithm's current solution in a random way guaranteeing that every element of M_i is added to the solution with probability exactly $1/k$. A formal presentation of the Random Greedy algorithm appears as Algorithm 2. Note that in this presentation the solution of the algorithm after i iterations is denoted by A_i.

We start the analysis of the Random Greedy algorithm with the following lemma.

Algorithm 2: Random Greedy (f, k)

Let $A_0 \leftarrow \varnothing$.
for $i = 1$ **to** k **do**
  Let $M_i \leftarrow \arg\max_{B \subseteq \mathcal{N} \setminus A_{i-1}, |B| \leq k} \{\sum_{u \in B} f(u \mid A_{i-1})\}$.
  with *probability* $(1 - |M_i|/k)$ **do**
    $A_i \leftarrow A_{i-1}$.
  otherwise
    Let u_i be a uniformly random element of M_i.
    Set $A_i \leftarrow A_{i-1} + u_i$.
return A_k.

Lemma D.2. *For every integer* $0 \leq i \leq k$ *and element* $u \in \mathcal{N}$, $\Pr[u \in A_i] \leq 1 - (1 - 1/k)^i$.

Proof. Note that in each iteration i of Algorithm 2, any element $u \in \mathcal{N} \setminus A_{i-1}$ is added to the current solution with probability of at most $1/k$. Hence,

$$\Pr[u \in A_i] = 1 - \Pr[u \notin A_i] = 1 - \prod_{j=1}^{i} \Pr[u \notin A_j \mid u \notin A_{j-1}] \leq 1 - (1 - 1/k)^i \ . \qquad \square$$

Plugging the guarantee of the last lemma into Lemma 2.1 yields the following lower bound on the expected value of $A_i \cup OPT$.

Corollary D.3. *For every integer* $0 \leq i \leq k$, $\mathbb{E}[f(A_i \cup OPT)] \geq [1 - (1 - m) \cdot (1 - (1 - \frac{1}{k})^i)] \cdot f(OPT) = m \cdot f(OPT) + (1 - m)(1 - \frac{1}{k})^i \cdot f(OPT)$.

Using the last corollary we are now ready to prove Theorem 4.2.

Proof of Theorem 4.2. Let $\mathcal{E}_{i-1}$ be an arbitrary possible choice for the random decisions of Random Greedy during its first $i - 1$ iterations. Observe that, conditioned on $\mathcal{E}_{i-1}$ happening,

$$\begin{aligned}\mathbb{E}[f(A_i) - f(A_{i-1})] &= \frac{\sum_{u \in M_i} f(u \mid A_{i-1})}{k} \\ &\geq \frac{\sum_{u \in OPT \setminus A_{i-1}} f(u \mid A_{i-1})}{k} \geq \frac{f(A_{i-1} \cup OPT) - f(A_{i-1})}{k} \ ,\end{aligned}$$

where the first inequality follows from the choice of M_i by the algorithm, and the second inequality follows from submodularity. Taking now expectation over the choice $\mathcal{E}_{i-1}$ that realized, the last inequality yields

$$\begin{aligned}\mathbb{E}[f(A_i) - f(A_{i-1})] &\geq \frac{\mathbb{E}[f(A_{i-1} \cup OPT)] - \mathbb{E}[f(A_{i-1})]}{k} \\ &\geq \frac{m \cdot f(OPT) + (1 - m)(1 - \frac{1}{k})^{i-1} \cdot f(OPT) - \mathbb{E}[f(A_{i-1})]}{k} \ ,\end{aligned} \tag{5}$$

where the second inequality is due to Corollary D.3.

The last inequality lower bounds the expected increase in the value of the solution of Random Greedy in every iteration. This implies also a lower bound on the expected value of $f(A_i)$. To complete the proof of the theorem, we need to prove a closed form for this implied lower bound, which we do by induction. Specifically, let us prove by induction on i that

$$\mathbb{E}[f(A_i)] \geq \left[m \cdot \left(1 - \left(1 - \frac{1}{k} \right)^i \right) + (1 - m) \cdot \frac{i}{k} \cdot \left(1 - \frac{1}{k} \right)^{i-1} \right] \cdot f(OPT) \tag{6}$$

for every integer $0 \leq i \leq k$, which implies the theorem by plugging $i = k$ because $(1 - 1/k)^k \leq 1/e \leq (1 - 1/k)^{k-1}$.

For $i = 0$, Inequality (6) holds since the non-negativity of f guarantees that $f(A_0) \geq 0 = [(1-m) \cdot (\frac{0}{k}) \cdot (1-\frac{1}{k})^{-1} + m \cdot (1-(1-\frac{1}{k})^0)] \cdot f(OPT)$. Consider now some integer $0 < i \leq k$, and let us prove Inequality (6) for this value of i assuming that its holds for $i-1$. By Inequality (5),

$$\begin{aligned}
\mathbb{E}[f(A_i)] &= \mathbb{E}[f(A_{i-1})] + \mathbb{E}[f(A_i) - f(A_{i-1})] \\
&\geq \mathbb{E}[f(A_{i-1})] + \frac{m \cdot f(OPT) + (1-m)(1-\frac{1}{k})^{i-1} \cdot f(OPT) - \mathbb{E}[f(A_{i-1})]}{k} \\
&= \left(1 - \frac{1}{k}\right) \cdot \mathbb{E}[f(A_{i-1})] + \frac{m + (1-m)(1-\frac{1}{k})^{i-1}}{k} \cdot f(OPT) \ .
\end{aligned}$$

Plugging the induction hypothesis into the last inequality, we get

$$\begin{aligned}
\mathbb{E}[f(A_i)] \geq \ & \left(1 - \frac{1}{k}\right) \cdot \left[m \cdot \left(1 - \left(1 - \frac{1}{k}\right)^{i-1}\right) + (1-m) \cdot \frac{i-1}{k} \cdot \left(1 - \frac{1}{k}\right)^{i-2}\right] \cdot f(OPT) \\
& + \frac{m + (1-m)(1-\frac{1}{k})^{i-1}}{k} \cdot f(OPT) \\
= \ & \left[m \left(1 - \left(1 - \frac{1}{k}\right)^{i}\right) + (1-m) \cdot \frac{i}{k} \cdot \left(1 - \frac{1}{k}\right)^{i-1}\right] \cdot f(OPT) \ . \qquad \square
\end{aligned}$$

D.3 Inapproximability for a Cardinality Constraint

In this section we state and prove the inapproximability result stated in Section 4.

Theorem D.4. *For any constant $\varepsilon > 0$, no polynomial time algorithm can obtain an approximation ratio of*

$$\min_{\alpha \in [0,1]} \frac{\max_{x \in [0,1]} \{\alpha(mx^2 + 2x - 2x^2) + 2(1-\alpha)(1-e^{x-1})(1-(1-m)x)\}}{\max\{1, 2(1-\alpha)\}} + \varepsilon$$

for the problem of maximizing a non-negative m-monotone submodular function subject to a cardinality constraint.

We prove Theorem D.4 using the symmetry gap technique, and specifically, via our extension of this technique proved in Theorem 3.2. To use this theorem, we need to construct an instance of our problem in which there is a large gap between the values of the best (general) solution and the best symmetric solution. Our instance is based on an instance constructed by Oveis Gharan and Vondrák [46]. However, the objective function in the original instance of [46] is not m-monotone for any $m > 0$, and therefore, we need to modify it so that it becomes m-monotone for a value $m \in [0,1]$ of our choosing.

Fix some positive integer value r to be determined later and some value $\alpha \in [0,1]$. The ground set of the instance we construct is $\mathcal{N} = \{a, b\} \cup \{a_i, b_i \mid i \in [r]\}$, and the constraint of the instance is a cardinality constraint allowing a feasible solution to include up to 2 elements. The objective function of our instance is the function $f \colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ defined by $f(S) = \alpha \cdot f_1(S) + (1-\alpha)[f_2(S) + f_3(S)]$, where

$$\begin{aligned}
f_1(S) &= m \cdot \mathbf{1}[S \cap \{a,b\} \neq \varnothing] + (1-m) \cdot (|S \cap \{a,b\}| \bmod 2) \ , \\
f_2(S) &= \mathbf{1}[S \cap \{a_i \mid i \in [r]\} \neq \varnothing] \cdot (1 - (1-m) \cdot \mathbf{1}[a \in S])
\end{aligned}$$

and

$$f_3(S) = \mathbf{1}[S \cap \{b_i \mid i \in [r]\} \neq \varnothing] \cdot (1 - (1-m) \cdot \mathbf{1}[b \in S]) \ .$$

Let us denote the above described instance of submodular maximization subject to a cardinality constraint by $\mathcal{I}$. We begin the analysis of $\mathcal{I}$ by proving some properties of its objective function.

Lemma D.5. *The objective function f of $\mathcal{I}$ is non-negative, m-monotone and submodular.*

Proof. We prove below that the functions f_1, f_2 and f_3 have the properties stated in the lemma. This implies that f also has these properties by Observation 2.2 and the well-known closure of the class of submodular functions to multiplication by a non-negative constant and addition (see, e.g., Lemma 1.2

of [4]). The function f_1 is identical to the function proved in Section 3 to have the properties stated in the lemma, and the functions f_2 and f_3 are identical to each other up to switching the roles of a with b and a_i with b_i. Therefore, to prove that both f_2 and f_3 have the properties stated by the lemma it suffices to show that f_2 has these properties, which we do in the rest of this proof.

Clearly, f_2 is non-negative. To see that f_2 is a submodular function, note that

- For every set $S \subseteq \mathcal{N} - a$, $f_2(a \mid S) = -\mathbf{1}[S \cap \{a_i \mid i \in [r]\} \neq \varnothing] \cdot (1 - m)$.
- For every integer $1 \leq i \leq r$ and set $S \subseteq \mathcal{N} - a_i$, $f_2(a_i \mid S) = \mathbf{1}[S \cap \{a_i \mid i \in [r]\} = \varnothing] \cdot (1 - (1 - m) \cdot \mathbf{1}[a \in S])$.
- For every element $u \in (\mathcal{N} - a) \setminus \{a_i \mid i \in [r]\}$ and set $S \subseteq \mathcal{N} - u$, $f_2(u \mid S) = 0$.

Since all the above marginal contributions are down-monotone functions of S (i.e., functions whose value can only decrease when elements are added to S), the function f_2 is submodular.

It remains to argue why f_2 is m-monotone. Consider any two sets $S \subseteq T \subseteq \mathcal{N}$. If $f_2(S) = 0$, then the inequality $m \cdot f(S) \leq f(T)$ follows from the non-negativity of f_2. Therefore, consider the case in which $f_2(S) > 0$, which implies that $S \cap \{a_i \mid i \in [r]\} \neq \varnothing$; and therefore, $f_2(S) = (1 - (1 - m) \cdot \mathbf{1}[a \in S]) \leq 1$. Since S is a subset of T, we also get $f_2(T) = (1 - (1 - m) \cdot \mathbf{1}[a \in T]) \geq m$, and hence, $m \cdot f_2(S) \leq m \cdot 1 = m \leq f_2(T)$. □

A cardinality constraint is symmetric in the sense that the feasibility of a set depends only on the number of elements in it, and is completely independent of the identity of these elements. Let us now denote by $\mathcal{G}$ the group of permutations of $\mathcal{N}$ that are equivalent to applying any number of the following two steps: (1) switching a with b and a_i with b_i for every $i \in [r]$, or (2) switching a_i with a_j for two integers $i, j \in [r]$. The first step preserves the value of f because it simply switches the values of f_2 and f_3, while leaving the value of f_1 unaffected; and the second step preserves the value of f since it deals with elements that both f_1 and f_3 ignore, and f_2 treats in the same way. Hence, for every set $S \subseteq \mathcal{N}$ and permutation $\sigma \in \mathcal{G}$, we have $f(S) = f(\sigma(S))$, which implies the following observation.

Observation D.6. *The instance $\mathcal{I}$ is strongly symmetric with respect to $\mathcal{G}$.*

To use Theorem 3.2, we still need to bound the symmetry gap of $\mathcal{I}$, which we do next.

Lemma D.7. *The symmetry gap of $\mathcal{I}$ is at most*

$$\frac{\max_{x \in [0,1]}\{\alpha(mx^2 + 2x - 2x^2) + 2(1-\alpha)[1 - (1 - (1-x)/r)^r](1 - (1-m)x)\}}{\max\{1, 2(1-\alpha)\}}$$
$$\leq \frac{\max_{x \in [0,1]}\{\alpha(mx^2 + 2x - 2x^2) + 2(1-\alpha)(1 - e^{x-1})(1 - (1-m)x)\}}{\max\{1, 2(1-\alpha)\}} + 2/r \ .$$

Proof. Two possible feasible solutions for $\mathcal{I}$ are the sets $\{a, b_1\}$ and $\{a_1, b_1\}$ whose values according to f are 1 and $2(1-\alpha)$, respectively. Therefore, the value of the optimal solution for $\mathcal{I}$ is at least $\max\{1, 2(1-\alpha)\}$. Since the symmetry gap is the ratio between the value of the best symmetric solution and the value of the best solution, to prove the lemma it remains to argue that the best symmetric solution for $\mathcal{I}$ has a value of $\max_{x \in [0,1]}\{\alpha(mx^2 + 2x - 2x^2) + 2(1-\alpha)[1 - (1 - (1 - x)/r)^r](1 - (1-m)x)\}$.

We remind the reader that a symmetric solution for $\mathcal{I}$ is $\bar{\mathbf{y}} = \mathbb{E}_{\sigma \in \mathcal{G}}[\mathbf{y}]$ for some vector $\mathbf{y} \in [0,1]^{\mathcal{N}}$ obeying $\|\mathbf{y}\|_1 \leq 2$. Since a and b can be exchanged with each other by the permutations of $\mathcal{G}$, the values of the coordinates of a and b in $\bar{\mathbf{y}}$ must be equal to each other. Similarly, every two elements of $\{a_i, b_i \in i \in [r]\}$ can be exchanged by the permutations of $\mathcal{G}$, and therefore, the values of the coordinates of these elements in $\bar{\mathbf{y}}$ must all be identical. Thus, any symmetric solution $\bar{\mathbf{y}}$ can be represented as

$$\bar{y}_u = \begin{cases} x & \text{if } u = a \text{ or } u = b \\ z & \text{if } u \in \{a_i, b_i \mid i \in [r]\} \end{cases} ,$$

for some values $x, z \in [0,1]$ obeying $2x + 2rz \leq 2$ (or equivalently, $z \leq (1-x)/r$). The value of this solution (according to the multilinear extension F of f) is

$$\begin{aligned}&\alpha[m(1-(1-x)^2) + 2(1-m)x(1-x)] + 2(1-\alpha)(1-(1-z)^r)(1-(1-m)x)\\ =\ &\alpha(mx^2 + 2x - 2x^2) + 2(1-\alpha)(1-(1-z)^r)(1-(1-m)x)\ .\end{aligned}$$

Since this expression is a non-decreasing function of z, the maximum value of any symmetry solution for $\mathcal{I}$ is

$$\begin{aligned}&\max_{\substack{x,z\in[0,1]\\ z\leq(1-x)/r}} \{\alpha(mx^2 + 2x - 2x^2) + 2(1-\alpha)(1-(1-z)^r)(1-(1-m)x)\}\\ =\ &\max_{x\in[0,1]} \{\alpha(mx^2 + 2x - 2x^2) + 2(1-\alpha)[1-(1-(1-x)/r)^r](1-(1-m)x)\}\ . \qquad \square\end{aligned}$$

Since any refinement of a cardinality constraint is a cardinality constraint over a larger ground set, plugging Lemma D.5, Observation D.6 and Lemma D.7 into Theorem 3.2 yields the following corollary.

Corollary D.8. *For every constant $\varepsilon' > 0$, no polynomial time algorithm for maximizing a non-negative m-monotone submodular function subject to a cardinality contraint obtains an approximation ratio of*

$$\frac{\max_{x\in[0,1]}\{\alpha(mx^2 + 2x - 2x^2) + 2(1-\alpha)(1-e^{x-1})(1-(1-m)x)\}}{\max\{1, 2(1-\alpha)\}} + 2/r + \varepsilon'\ .$$

Theorem D.4 now follows from the last corollary by choosing $\varepsilon' = \varepsilon/2$, $r = \lceil 4/\varepsilon \rceil$ and

$$\alpha = \operatorname*{arg\,min}_{\alpha'\in[0,1]} \frac{\max_{x\in[0,1]}\{\alpha'(mx^2 + 2x - 2x^2) + 2(1-\alpha')(1-e^{x-1})(1-(1-m)x)\}}{\max\{1, 2(1-\alpha')\}}\ .$$

E Inapproximability and Proofs of Section 5

In this section we state and analyze the algorithms used to prove the results given in Section 5. We also state and prove in Section E.4 the inapproximability result mentioned in Section 5.

E.1 Analysis of the Greedy algorithm

A version of the greedy algorithm designed for matroid constraints appears as Algorithm 3. This algorithm starts with an empty solution, and then iteratively adds elements to this solution, where the element added in each iteration is the element with the largest marginal contrition with respect to the current solution among all the elements whose addition to the solution does not violate feasibility. The algorithm terminates when no additional elements can be added to the solution without decreasing its value.

Algorithm 3: The Greedy Algorithm (for a Matroid Constraint) $(f, \mathcal{M} = (\mathcal{N}, \mathcal{I}))$

```
Let A_0 ← ∅, and i ← 0.
while true do
    Let u_{i+1} be the element of {v ∈ N \ A_i | A_i + v ∈ I} maximizing f(u_{i+1} | A_i).
    if f(u_{i+1} | A_i) ≥ 0 then Let A_{i+1} ← A_i + u_{i+1}, and then, increase i by 1.
    else return A_i.
```

Theorem 5.1. *The Greedy algorithm (Algorithm 3) has an approximation ratio of at least $m/2$ for maximizing a non-negative m-monotone submodular function subject to a matroid constraint.*

Proof. Lemma 3.2 of [27] shows that the greedy algorithm outputs a solution S of value at least $f(S \cup OPT)/2$ for the problem of maximizing a non-negative submodular function f subject to a matroid constraint, where OPT is an optimal solution for the problem.[5] The theorem now follows since for an m-monotone function f we are guaranteed to have $f(S \cup OPT) \geq m \cdot f(OPT)$. $\square$

[5] In fact, Lemma 3.2 of [27] proves a more general result for p-set systems, but it implies the stated result since matroids are 1-set systems.

E.2 Analysis of Measured Continuous Greedy

In this section, we reanalyze the Measured Continuous Greedy algorithm of [22] in view of the monotonicity ratio. Given a non-negative submodular function $f\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ and a down-closed solvable[6] convex body $P \subseteq [0,1]^{\mathcal{N}}$, Measured Continuous Greedy is an algorithm designed to find a vector $\mathbf{x} \in P$ that approximately maximize $F(\mathbf{x})$, where F is the multilinear extension of f. Specifically, we prove the following theorem.

Theorem E.1. *Given a non-negative m-monotone submodular function $f\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$, a solvable down-close convex body $P \subseteq [0,1]^{\mathcal{N}}$ and a parameter $T \geq 0$, Measured Continuous Greedy outputs a vector $\mathbf{x} \in [0,1]^{\mathcal{N}}$ obeying $F(\mathbf{x}) \geq [m(1-e^{-T}) + (1-m)Te^{-T}] \cdot f(OPT)$, where F is the multilinear extension of f and OPT is the set maximizing f among all sets whose characteristic vectors belong to P. Furthermore, $\mathbf{x} \in P$ whenever $T \in [0,1]$.*

We note that Feldman et al. [22] discussed conditions that guarantee that $\mathbf{x}$ belongs to P also for some values of T that are larger than 1. However, the above stated form of Theorem E.1 already suffices to prove Theorem 5.2. Let us explain why this is the case. When P is the matroid polytope $P_{\mathcal{M}}$ of a matroid $\mathcal{M}$, there are algorithms called Pipage Rounding [9] and Swap Rounding [11] that, given a vector $\mathbf{x} \in P$ produce a set S that is independent in $\mathcal{M}$ and also obeys $\mathbb{E}[f(S)] \geq F(\mathbf{x}) - o(1) \cdot f(OPT)$. Therefore, one can obtain an algorithm for maximizing f subject to the matroid $\mathcal{M}$ by executing Measured Continuous Greedy with $P = P_{\mathcal{M}}$ and $T = 1$, and then applying either Pipage Rounding or Swap Rounding to the resulting vector; which yields an algorithm with the properties specified by Theorem 5.2.

We now describe the version of Measured Continuous Greedy that we analyze (given as Algorithm 4). For simplicity, we chose to analyze a continuous version of this algorithm that assumes direct access to the multilinear extension F of the objective function rather than just to the objective function itself. We refer the reader to [22] for details about discretizing the algorithm and avoiding the assumption of direct access to F. We also note that the $o(1)$ error term in the approximation guarantee stated in Theorem E.1 is due to these issues. Our description of Measured Continuous Greedy requires some additional notation, namely, given two vectors $\mathbf{x}$ and $\mathbf{y}$, we denote by $\mathbf{x} \vee \mathbf{y}$ their coordinate-wise maximum and by $\mathbf{x} \odot \mathbf{y}$ their coordinate-wise multiplication.

Measured Continuous Greedy starts at "time" 0 with the empty solution, and improves this solution during the time interval $[0,T]$. We denote the solution of the algorithm at time t by $\mathbf{y}(t)$. At every time $t \in [0,T]$, the algorithm calculates a vector $\mathbf{w}$ whose u-coordinate is the gain that can be obtained by increasing this coordinates in the solution $\mathbf{y}(t)$ to be 1 (i.e., $w_u(t) = F(\mathbf{y}(t) \vee 1_{\{u\}}) - F(\mathbf{y}(t))$). Then, the algorithm finds a vector $\mathbf{x}(t) \in P$ that maximizes the objective function $\mathbf{w}(t) \cdot \mathbf{x}(t)$, and adds to the solution $\mathbf{y}(t)$ an infinitesimal part of $(\mathbf{1}_{\mathcal{N}} - \mathbf{y}(t)) \odot \mathbf{x}(t)$ (to understand where the last expression comes from, we note that when $\mathbf{x}$ is integral, fully adding $(\mathbf{1}_{\mathcal{N}} - \mathbf{y}(t)) \odot \mathbf{x}(t)$ to $\mathbf{y}(t)$ sets to 1 all the coordinates that are 1 in $\mathbf{x}(t)$, which matches the "spirit" of the definition of $\mathbf{w}$).

Algorithm 4: Measured Continuous Greedy(f, P, T)

Let $\mathbf{y}(0) \leftarrow 1_{\varnothing}$.
foreach $t \in [0,T)$ **do**
  For each $u \in \mathcal{N}$, let $w_u(t) \leftarrow F(\mathbf{y}(t) \vee 1_{\{u\}}) - F(\mathbf{y}(t))$.
  Let $\mathbf{x}(t) \leftarrow \arg\max_{\mathbf{x} \in P}\{\mathbf{w}(t) \cdot \mathbf{x}\}$.
  Increase $\mathbf{y}(t)$ at a rate of $\frac{d\mathbf{y}(t)}{dt} = (\mathbf{1}_{\mathcal{N}} - \mathbf{y}(t)) \odot \mathbf{x}(t)$.
return $y(T)$.

The first step in the analysis of Measured Continuous Greedy is bounding the maximum value of the coordinates of the solution $\mathbf{y}(t)$.

Lemma E.2. *For every $t \in [0,T]$, $\|\mathbf{y}(t)\|_\infty \leq 1 - e^{-t}$.*

[6] A body $P \subseteq [0,1]^{\mathcal{N}}$ is *solvable* if one can efficiently optimize linear functions subject to it, and is *down-closed* if $\mathbf{y} \in P$ implies $\mathbf{x} \in P$ for every vector $\mathbf{x} \in [0,1]^{\mathcal{N}}$ obeying $\mathbf{x} \leq \mathbf{y}$ (this inequality should be understood to hold coordinate-wise).

Proof. Fix an arbitrary element $u \in \mathcal{N}$, and let us explain why $y_u(t) \leq 1 - e^{-t}$. By Line 5 of Algorithm 4, $y_u(t)$ obeys the differential inequality

$$\frac{dy_u(t)}{dt} = (1 - y_u(t)) \cdot x_u(t) \leq 1 - y_u(t) \ ,$$

and the solution of this differential inequality for the initial condition $y_u = 0$ is

$$y_u(t) \leq 1 - e^{-t} \ . \qquad \square$$

We are now ready to prove Theorem E.1

Proof of E.1. Recall that $\mathbf{x}(t)$ is a vector inside P for every time $t \in [0, T]$, and since P is down-closed, $(\mathbf{1}_{\mathcal{N}} - \mathbf{y}(t)) \odot \mathbf{x}(t)$ and $\mathbf{1}_{\varnothing}$ both belong to P as well. This means that for $T \leq 1$ the vector $\mathbf{y}(T) = (1 - T) \cdot \mathbf{1}_{\varnothing} + \int_0^T (\mathbf{1}_{\mathcal{N}} - y(t)) \odot \mathbf{x}(t) dt$ is a convex combination of vectors in P, and therefore belongs to P by the convexity of P.

It remains to lower bound the value of $F(\mathbf{y}(T))$. By the chain rule,

$$\frac{dF(\mathbf{y}(t))}{dt} = \sum_{u \in \mathcal{N}} \left(\frac{dy_u(t)}{dt} \cdot \frac{\partial F(\mathbf{y})}{\partial y_u}\Big|_{\mathbf{y}=\mathbf{y}(t)} \right) = \sum_{u \in \mathcal{N}} \left((1 - y_u(t)) \cdot x_u(t) \cdot \frac{\partial F(\mathbf{y})}{\partial y_u}\Big|_{\mathbf{y}=\mathbf{y}(t)} \right) \ .$$

Since F is multilinear, its partial derivative with respect to a single coordinate is equal to the difference between the value of the function for two different values of this coordinate over the difference between these values. Plugging this observation into the previous inequality yields

$$\frac{dF(\mathbf{y}(t))}{dt} = \sum_{u \in \mathcal{N}} \left((1 - y_u(t)) \cdot x_u(t) \cdot \frac{F(\mathbf{y}(t) \vee 1_{\{u\}}) - F(\mathbf{y}(t))}{1 - y_u(t)} \right) = \mathbf{x}(t) \cdot \mathbf{w}(t) \ .$$

One possible candidate to be $\mathbf{x}(t)$ is $\mathbf{1}_{OPT}$. Hence, by the definition of $\mathbf{x}(t)$, $\mathbf{x}(t) \cdot \mathbf{w}(t) \geq \mathbf{1}_{OPT} \cdot \mathbf{w}(t)$. Combining this inequality with the previous one, we get

$$\begin{aligned}
\frac{dF(\mathbf{y}(t))}{dt} &\geq \mathbf{1}_{OPT} \cdot \mathbf{w}(t) = \sum_{u \in OPT} \left[F(\mathbf{y}(t) \vee 1_{\{u\}}) - F(\mathbf{y}(t)) \right] \\
&\geq F(\mathbf{y}(t) \vee \mathbf{1}_{OPT}) - F(\mathbf{y}(t)) \geq [1 - (1 - m) \cdot \|\mathbf{y}(t)\|_\infty] \cdot f(OPT) - F(\mathbf{y}(t)) \\
&\geq [1 - (1 - m)(1 - e^{-t})] \cdot f(OPT) - F(\mathbf{y}(t)) \\
&= [m + (1 - m)e^{-t}] \cdot f(OPT) - F(\mathbf{y}(t)) \ ,
\end{aligned}$$

where the second inequality holds by the submodularity of f, the penultimate inequality holds by Lemma 2.1, and the last inequality follows from Lemma E.2.

Solving the differential inequality that we got for the initial condition $F(\mathbf{y}(0)) \geq 0$ (which holds by the non-negativity of f) yields

$$F(y(t)) \geq \left[m(1 - e^{-t}) + (1 - m)te^{-t} \right] \cdot f(OPT) \ ,$$

and the theorem now follows by plugging $t = T$. $\square$

E.3 Analysis of Random Greedy for Matroids

In this section we prove Theorem 5.3, which we repeat here for convenience.

Theorem 5.3. *For every $\varepsilon \in (0, 1)$, Random Greedy for Matroids (Algorithm 5) has an approximation ratio of at least $\frac{1+m+e^{-2/(1-m)}}{4} - \varepsilon - o_k(1)$ for the problem of maximizing a non-negative m-monotone submodular function subject to a matroid constraint (except in the case of $m = 1$ in which the approximation ratio is $1/2 - \varepsilon - o_k(1)$), where $o_k(1)$ represents a term that diminishes with k.*

To prove the theorem, we first need to state the algorithm it refers to. Towards this goal, let us assume that the ground set $\mathcal{N}$ contains a set D of $2k$ "dummy" elements that are known to the algorithm and have the following two properties.

- $f(S) = (S \setminus D)$ for every set $S \subseteq \mathcal{N}$.

- $S \in \mathcal{I}$ if and only if $S \setminus D \in \mathcal{I}$ and $|S| \leq k$.

This assumption is useful since it allows us to assume that the optimal solution OPT is a base of $\mathcal{M}$, and thus, simplifies the description of our algorithm (Random Greedy for Matroids). We can justify our assumption using the following procedure: (i) add $2k$ dummy elements to the ground set, (ii) extend f and $\mathcal{I}$ according to the above properties, (iii) execute Random Greedy for Matroids on the resulting instance, and (iv) remove from the output of the algorithm any dummy elements that end up in it. This procedure guarantees that any approximation guarantee obtained by Random Greedy for Matroids using our assumption can be obtained also without the assumption.

Our version of the Random Greedy for Matroids algorithm is given as Algorithm 5. Like the original version of the algorithm (due to [6]), our version starts with a base of $\mathcal{M}$ consisting only of dummy elements, and then modifies it in a series of iterations. In each iteration i, the algorithm starts with a solution S_{i-1}, and then identifies a base M_i of $\mathcal{M}$ whose elements have the largest total marginal contribution with respect to S_{i-1} (M_i is also required to be disjoint from S_{i-1}). The algorithm then picks a uniformly random element $u_i \in S_{i-1}$, and adds it to the solution S_{i-1} at the expense of an element $g_i(u_i)$ of S_{i-1} given by a function g_i that is chosen carefully (the existence of such a function follows, for example, from Corollary 39.12a of [47]).

As mentioned above, our version of Random Greedy for Matroids differs compared to the version of [6] in two respects. The first modification is in the number of iterations that the algorithm makes. To get the result of Buchbinder et al. [6], it suffices to use k iterations. However, the optimal number of iterations increases with m, and therefore, our version of the algorithm uses k/ε iterations for some parameter $\varepsilon \in (0, 1)$ (we assume without loss of generality that k/ε is integral; otherwise, we can replace ε with a value which is smaller than ε by at most a factor of 2 and has this property). Furthermore, since we do not want to assume knowledge of m in the algorithm, we use a number of iterations that is appropriate for $m = 1$, which requires us to make the second modification to the algorithm; namely, we check whether replacing $g(u_i)$ with u_i is beneficial, and make the swap only if this is indeed the case. This guarantees that doing more iterations can never decrease the value of the algorithm's solution.

Algorithm 5: Random Greedy for Matroids($f, \mathcal{M} = (\mathcal{N}, \mathcal{I}), \varepsilon$)

Initialize S_0 to be an arbitrary base containing only elements of D.
for $i = 1$ **to** k/ε **do**
  Let $M_i \subseteq N$ be a base of $\mathcal{M}$ that contains only elements of $\mathcal{N} \setminus S_{i-1}$ and maximizes $\sum_{u \in M_i} f(u \mid S_{i-1})$ among all such bases.
  Let g_i be a function mapping each element of M_i to an element of S_{i-1} obeying $S_{i-1} - g_i(u) + u \in \mathcal{I}$ for every $u \in S_{i-1}$.
  Let u_i be a uniformly random element from M_i. **if** $f(S_{i-1} - g_i(u_i) + u_i) > f(S_{i-1})$ **then** Let $S_i \leftarrow S_{i-1} - g_i(u_i) + u_i$.
  else Let $S_i \leftarrow S_{i-1}$.
return $S_{k/\varepsilon}$.

Since Theorem 5.3 is trivial for a constant k, we can assume in the analysis of Algorithm 5 that k is larger than any given constant. The first step in this analysis is proving the following lower bound on the expected value of $OPT \cup S_i$.

Observation E.3. *For every integer* $0 \leq i \leq k/\varepsilon$, $\mathbb{E}[f(OPT \cup S_i)] \geq \frac{1}{2}(1 + m + (1 - m)(1 - 2/k)^i) \cdot f(OPT)$.

Proof. For every integer $0 \leq i \leq k/\varepsilon$ and element $u \in \mathcal{N} \setminus D$, let $p_{u,i}$ denote the probability u belongs to S_i. We would like to argue that when $i > 0$, we have $p_{u,i} \leq p_{u,i-1}(1 - 2/k) + 1/k$. To see why this is the case, note that u belongs to S_i only if one of the following happens: (i) u belongs to S_{i-1} and is not removed from the solution (happens with probability $p_{u,i-1}(1 - 1/k)$ since $g_i(u_i)$ is a uniformly random element of S_{i-1}), or (ii) u belongs to M_{i-1} and is chosen as u_i (happens with probability at most $(1 - p_{u,i})/k$). Therefore,

$$p_{u,i} \leq p_{u,i-1} \cdot (1 - 1/k) + (1 - p_{u,i-1})/k = p_{u,i-1} \cdot (1 - 2/k) + 1/k \ . \tag{7}$$

Next, we aim to prove by induction that $p_{u,i} \leq \frac{1}{2}(1-(1-2/k)^i)$ for every integer $0 \leq i \leq k/\varepsilon$. For $i = 0$, this is true since $u \in \mathcal{N} \setminus D$ implies that $p_{u,0} = 0 = \frac{1}{2}(1-(1-2/k)^0)$. Assume now that the claim holds for $i-1$, and let us prove it for $i \geq 1$. By the induction hypothesis and Inequality (7),

$$p_{u,i} \leq p_{u,i-1}(1-2/k) + 1/k \leq \tfrac{1}{2}(1-(1-2/k)^{i-1})(1-2/k) + 1/k = \tfrac{1}{2}(1-(1-2/k)^i) \ .$$

The observation now follows since Lemma 2.1 guarantees that $\mathbb{E}[f(OPT \cup S_i)] = \mathbb{E}[f(OPT \cup (S_i \setminus D))] \geq (1-(1-m) \cdot \max_{u \in \mathcal{N} \setminus D} p_{i,u}) \cdot f(OPT)$. □

Below we prove a lower bound on the value of the solution of Algorithm 5 after any number of iterations. However, to prove this lower bound we first need to prove the following technical observation.

Observation E.4. *For every positive integer i,*

$$\left(1-\frac{2}{k}\right)^{i-1} \geq e^{-\frac{2i}{k}} - \frac{k}{i} \cdot o_k(1) \ .$$

Proof. Note that

$$\begin{aligned} e^{-\frac{2i}{k}} &= \left(e^{-\frac{2}{k}}\right)^i \leq \left(1-\frac{2}{k}+\frac{4}{k^2}\right)^i \leq \left(1-\frac{2}{k}\right)^i + \frac{4}{k^2} \cdot i \left(1-\frac{2}{k}+\frac{4}{k^2}\right)^{i-1} \\ &\leq \left(1-\frac{2}{k}\right)^i + \frac{4i}{k^2}\left(1-\frac{1}{k}\right)^{i-1} \leq \left(1-\frac{2}{k}\right)^i + \frac{4i}{k^2} \cdot e^{-\frac{i-1}{k}} \ , \end{aligned}$$

where the third inequality holds for $k \geq 4$, and the second inequality holds since the derivative of the function $(1-2/k+x)^i$ is $i(1-2/k+x)^{i-1}$, which implies

$$\begin{aligned} \left(1-\frac{2}{k}+\frac{4}{k^2}\right)^i &= \left(1-\frac{2}{k}\right)^i + \int_0^{4/k^2} i(1-2/k+x)^{i-1} dx \\ &\leq \left(1-\frac{2}{k}\right)^i + \frac{4i}{k^2}(1-2/k+4/k^2)^{i-1} dx \ . \end{aligned}$$

To complete the proof of the observation, it remains to note that, since the maximum of the function $x^2 e^{-x}$ for $x \geq 0$ is $4e^{-2}$,

$$\frac{4i}{k^2} \cdot e^{-\frac{i-1}{k}} \leq \frac{16e^{-2}}{i} \cdot e^{\frac{1}{k}} = \frac{k}{i} \cdot o_k(1) \ .$$ □

We are now ready to prove the promised lower bound on the value of the solution S_i of Algorithm 5 after any number of iterations.

Lemma E.5. *For every integer $0 \leq i \leq k/\varepsilon$,*

$$\mathbb{E}[f(S_i)] \geq \left[\frac{1+m}{4} \cdot \left(1-e^{-\frac{2i}{k}}\right) + \frac{(1-m)i}{2k} \cdot e^{-\frac{2i}{k}} - o_k(1)\right] \cdot f(OPT) \ .$$

Proof. For $i = 0$ the lemma follows from the non-negativity of f since the right hand side of the inequality that we need to prove is non-positive for $i = 0$. Together with Observation E.4, this implies that it suffices to prove the following inequality

$$\mathbb{E}[f(S_i)] \geq \left[\frac{1+m}{4} \cdot \left(1-\left(1-\frac{2}{k}\right)^i\right) + \frac{(1-m)i}{2k} \cdot \left(1-\frac{2}{k}\right)^{i-1}\right] \cdot f(OPT) \ , \qquad (8)$$

and the rest of the proof is devoted to this goal.

Fix an arbitrary integer $1 \leq i \leq k/\varepsilon$. We would like to derive a lower bound on the expected marginal contribution of the element u_i to the set S_{i-1}, and an upper bound on the expected marginal contribution of the element $g(u_i)$ to the set $S_{i-1} \setminus g(u_i)$. Let A_{i-1} be an event fixing all random choices of Algorithm 5 up to iteration $i-1$ (including), and let $\mathcal{A}_{i-1}$ be the set of all possible A_{i-1}

events. Conditioned on any event $A_{i-1} \in \mathcal{A}_{i-1}$, the sets S_{i-1} and M_i becomes deterministic, and we can define M'_i as a set containing the elements of $OPT \setminus S_{i-1}$ plus enough dummy elements of $D \setminus S_{i-1}$ to make the size of M'_i exactly k. Then,

$$\begin{aligned}\mathbb{E}[f(u \mid S_{i-1}) \mid A_{i-1}] &= \frac{\sum_{u \in M_i} f(u \mid S_{i-1})}{k} \geq \frac{\sum_{u \in M'_i} f(u \mid S_{i-1})}{k} \\ &= \frac{\sum_{u \in OPT \setminus S_{i-1}} f(u \mid S_{i-1})}{k} \geq \frac{f(OPT \cup S_{i-1}) - f(S_{i-1})}{k} \ ,\end{aligned}$$

where S_i, M_i and M'_i represent here their values conditioned on A_i, the first inequality follows from the definition of M_i and the second inequality holds by the submodularity of f. Similarly,

$$\begin{aligned}\mathbb{E}[f(g(u_i) \mid S_{i-1} - g(u_i)) \mid A_{i-1}] &= \frac{\sum_{u \in M_i} f(g(u_i) \mid S_{i-1} - g(u))}{k} \\ &\leq \frac{f(S_{i-1}) - f(\varnothing)}{k} \leq \frac{f(S_{i-1})}{k} \ ,\end{aligned}$$

where the first inequality follows from the submodularity of f. Taking expectation over the event A_{i-1}, we get

$$\begin{aligned}\mathbb{E}[f(u_i \mid S_{i-1})] &\geq \frac{\mathbb{E}[f(OPT \cup S_{i-1})] - \mathbb{E}[f(S_{i-1})]}{k} \\ &\geq \frac{\frac{1}{2}(1 + m + (1-m)(1-2/k)^{i-1}) \cdot f(OPT) - \mathbb{E}[f(S_{i-1})]}{k} \ ,\end{aligned}$$

where the last inequality is due to Observation E.3, and

$$\mathbb{E}[f(g(u_i) \mid S_{i-1} - g(u_i))] \leq \frac{\mathbb{E}[f(S_{i-1})]}{k} \ .$$

Combing the last two inequalities now yields

$$\begin{aligned}\mathbb{E}[f(S_i)] &\geq \mathbb{E}[f(S_{i-1} - g(u_i) + u_i)] \\ &= \mathbb{E}[f(S_{i-1})] + \mathbb{E}[f(u_i \mid S_{i-1} - g(u_i))] - \mathbb{E}[f(g(u_i) \mid S_{i-1} - g(u_i)] \\ &\geq \mathbb{E}[f(S_{i-1})] + \mathbb{E}[f(u_i \mid S_{i-1})] - \mathbb{E}[f(g(u_i) \mid S_{i-1} - g(u_i)] \\ &\geq \left(1 - \frac{2}{k}\right) \cdot \mathbb{E}[f(S_{i-1})] + \frac{\frac{1}{2}(1 + m + (1-m)(1-2/k)^{i-1}) \cdot f(OPT)}{k} \ ,\end{aligned} \tag{9}$$

where the first inequality follows from the submodularity of f since $g(u_i) \neq u_i$ because $g(u_i) \in S_{i-1}$ and $u_i \in M_i$.

Since Inequality (9) holds for every integer $1 \leq i \leq k/\varepsilon$, we can use it repeatedly to get, for every integer $0 \leq i \leq k/\varepsilon$,

$$\begin{aligned}\mathbb{E}[f(S_i)] \geq \frac{1}{2k} &\left[(1+m)\sum_{j=1}^{i}\left(1 - \frac{2}{k}\right)^{i-j} + (1-m)\sum_{j=1}^{i}\left(1 - \frac{2}{k}\right)^{i-1}\right] \cdot f(OPT) \\ &+ \left(1 - \frac{2}{k}\right)^{i} \cdot f(S_0) \ .\end{aligned}$$

Since the non-negativity of f guarantees that $f(S_0) \geq 0$, the last inequality implies Inequality (8), and therefore, completes the proof of the lemma. □

One can show that the lower bound for $f(S_i)$ proved by Lemma E.5 is maximized when $i = k/(1-m)$. Unfortunately, we cannot simply plug this i value into the lower bound due to two issues: this value of i might not be integral, and this value of i might be larger than the number k/ε of iterations. The following two lemmata prove the approximation guarantee of Theorem 5.3 despite these issues, and together they complete the proof of the theorem.

Lemma E.6. *When $m \leq 1 - \varepsilon$, the approximation ratio of Algorithm 5 is at least*

$$\frac{1 + m + e^{-2/(1-m)}}{4} - o_k(1) \ .$$

Proof. Let $i' = \lfloor k/(1-m) \rfloor$. Due to the condition of the lemma, Algorithm 5 makes at least i' iterations. Furthermore, since Algorithm 5 makes a swap in its solution only when this swap is beneficial, the expected value of the output of the algorithm is at least

$$\begin{aligned}
\mathbb{E}[f(S_{i'})] &\geq \left[\frac{1+m}{4} \cdot \left(1 - e^{-\frac{2i'}{k}}\right) + \frac{(1-m)i'}{2k} \cdot e^{-\frac{2i'}{k}} - o_k(1)\right] \cdot f(OPT) \\
&\geq \left[\frac{1+m}{4} \cdot \left(1 - e^{\frac{2}{k} - \frac{2}{1-m}}\right) + \frac{k-1}{2k} \cdot e^{-\frac{2}{1-m}} - o_k(1)\right] \cdot f(OPT) \\
&\geq \left[\frac{1+m}{4} \cdot \left(1 - e^{-\frac{2}{1-m}}\right) - \frac{e^{\frac{2}{k}} - 1}{2} + \frac{1}{2} \cdot e^{-\frac{2}{1-m}} - \frac{1}{2k} - o_k(1)\right] \cdot f(OPT) \ ,
\end{aligned}$$

where the first inequality follows from Lemma E.5, and the second inequality holds since $k/(1-m) - 1 \leq i' \leq k/(1-m)$. Since the terms $\frac{e^{2/k}-1}{2}$ and $\frac{1}{2k}$ are both diminishing with k (and therefore, can be replaced with $o_k(1)$), the last inequality implies the lemma. □

Lemma E.7. *When $1 - \varepsilon \leq m < 1$, the approximation ratio of Algorithm 5 is at least*

$$\frac{1 + m + e^{-2/(1-m)}}{4} - \varepsilon - o_k(1) \ ,$$

and when $m = 1$ the approximation ratio of this algorithm is at least $1/2 - \varepsilon - o_k(1)$.

Proof. The output set of Algorithm 5 is $f(S_{k/\varepsilon})$. By Lemma E.5, the expected value of this set is at least

$$\begin{aligned}
\mathbb{E}[f(S_{k/\varepsilon})] &\geq \left[\frac{1+m}{4} \cdot \left(1 - e^{-\frac{2}{\varepsilon}}\right) + \frac{1-m}{2\varepsilon} \cdot e^{-\frac{2}{\varepsilon}} - o_1(k)\right] \cdot f(OPT) \\
&\geq \left[\frac{1+m}{4} \cdot \left(1 - e^{-\frac{2}{\varepsilon}}\right) - o_k(1)\right] \cdot f(OPT) \\
&\geq \left[\frac{1+m}{4} \cdot \left(1 - \frac{1}{1+2/\varepsilon}\right) - o_k(1)\right] \cdot f(OPT) \\
&= \left[\frac{1+m}{2(\varepsilon+2)} - o_k(1)\right] \cdot f(OPT) \geq \left[\frac{1+m}{4} - \frac{\varepsilon}{4} - o_k(1)\right] \cdot f(OPT) \ ,
\end{aligned}$$

where the third inequality holds since for every $x \geq 0$, $\ln(1/(1+x)) = \ln(1 - x/(1+x)) \geq -\frac{x/(1+x)}{1-x/(1+x)} = -x$.

The above inequality completes the proof for the case of $m = 1$. To complete the proof also for the case of $1 - \varepsilon \leq m < 1$, it suffice to observe that in this case

$$e^{-2/(1-m)} \leq e^{-2/\varepsilon} \leq \frac{1}{1+2/\varepsilon} \leq \frac{\varepsilon}{2} \ .$$

□

E.4 Inapproximability for a Matroid Constraints

In this section we state and prove the inapproximability result mentioned in Section 5.

Theorem E.8. *For any constant $\varepsilon > 0$, no polynomial time algorithm can obtain an approximation ratio of*

$$\min_{\alpha \in [0,1]} \max_{x \in [0,1/2]} \{\alpha(mx^2 + 2x - 2x^2) + 2(1-\alpha)(1 - e^{-1/2})(1 - (1-m)x)\} + \varepsilon$$

for the problem of maximizing a non-negative m-monotone submodular function subject to a matroid constraint.

The proof of Theorem E.8 is very similar to the proof of Theorem D.4. Recall that in Section D.3, we proved Theorem D.4 by constructing an instance $\mathcal{I}$ of submodular maximization subject to a cardinality constraint, and then applying Theorem 3.2 to this instance. The proof of Theorem E.8 is based on an instance $\mathcal{I}'$ of submoduar maximization subject to a matroid constrained that is identical

to $\mathcal{I}$ except for the following difference. In $\mathcal{I}$, the constraint is a cardinality constraint allowing the selection of up to 2 elements from the ground set $\mathcal{N} = \{a, b\} \cup \{a_i, b_i \mid i \in [r]\}$. In $\mathcal{I}'$, we have instead a (simplified) partition matroid constraint allowing the selection of up to 1 element from $\{a, b\}$ and up to 1 element from $\{a_i, b_i \mid i \in [r]\}$.

Since the instances $\mathcal{I}$ and $\mathcal{I}'$ have the same objective function, the properties of this function stated in Lemma D.5 apply to both of them. Furthermore, one can verify that $\mathcal{I}'$ is strongly symmetric with respect to the group $\mathcal{G}$ of permutation defined in Section D.3. Therefore, we concentrate on analyzing the symmetry gap of $\mathcal{I}'$.

Lemma E.9. *The symmetry gap of $\mathcal{I}'$ is at most*

$$\begin{aligned}&\max_{x\in[0,1/2]}\{\alpha(mx^2+2x-2x^2)+2(1-\alpha)[1-(1-1/(2r))^r](1-(1-m)x)\}\\ \leq&\max_{x\in[0,1/2]}\{\alpha(mx^2+2x-2x^2)+2(1-\alpha)(1-e^{-1/2})(1-(1-m)x)\}+1/(2r)\ .\end{aligned}$$

Proof. One possible feasible solution for $\mathcal{I}'$ is the set $\{a, b_1\}$ whose value according to f is 1. Therefore, the value of the optimal solution for $\mathcal{I}'$ is at least 1. Since the symmetry gap is the ratio between the value of the best symmetric solution and the value of the best solution, to prove the lemma it remains to argue that the best symmetric solution for $\mathcal{I}$ has a value of at most $\max_{x\in[0,1/2]}\{\alpha(mx^2+2x-2x^2)+2(1-\alpha)[1-(1-1/(2r))^r](1-(1-m)x)\}$.

We remind the reader that a symmetric solution for $\mathcal{I}'$ is $\bar{\mathbf{y}} = \mathbb{E}_{\sigma\in\mathcal{G}}[\mathbf{y}]$ for some vector $\mathbf{y} \in [0,1]^{\mathcal{N}}$ obeying $y_a + y_b \leq 1$ and $\sum_{i=1}^{r} y_{a_i} + y_{b_i} \leq 1$. Since a and b can be exchanged with each other by the permutations of $\mathcal{G}$, the values of the coordinates of a and b in $\bar{\mathbf{y}}$ must be equal to each other. Similarly, every two elements of $\{a_i, b_i \in i \in [r]\}$ can be exchanged by the permutations of $\mathcal{G}$, and therefore, the values of the coordinates of all these elements in $\bar{\mathbf{y}}$ must be all identical. Thus, any symmetric solution $\bar{\mathbf{y}}$ can be represented as

$$\bar{y}_u = \begin{cases} x & \text{if } u = a \text{ or } u = b \\ z & \text{if } u \in \{a_i, b_i \mid i \in [r]\} \end{cases}\ ,$$

for some values $x \in [0, 1/2]$ and $z \in [0, 1/(2r)]$. The value of this solution (according to the multilinear extension F of f) is

$$\begin{aligned}&\alpha[m(1-(1-x)^2)+2(1-m)x(1-x)]+2(1-\alpha)(1-(1-z)^r)(1-(1-m)x)\\ =&\ \alpha(mx^2+2x-2x^2)+2(1-\alpha)(1-(1-z)^r)(1-(1-m)x)\\ \leq&\ \alpha(mx^2+2x-2x^2)+2(1-\alpha)(1-(1-1/(2r))^r)(1-(1-m)x)\ .\end{aligned}$$

Therefore, one can obtain an upper bound on the value of the best symmetric solution for $\mathcal{I}'$ by taking the maximum of the last expression over all the values that x can take, which completes the proof of the lemma. □

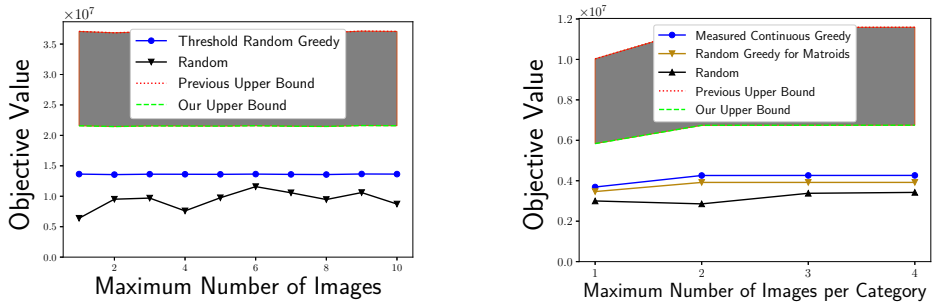
Since any refinement of a (simplified) partition matroid constraint is a (generalized) partition matroid constraint on its own right, plugging Lemmata D.5 and Lemma E.9 into Theorem 3.2 yields the following corollary.

Corollary E.10. *For every constant $\varepsilon' > 0$, no polynomial time algorithm for maximizing a non-negative m-monotone submodular function subject to a matroid contraint obtains an approximation ratio of*

$$\max_{x\in[0,1/2]}\{\alpha(mx^2+2x-2x^2)+2(1-\alpha)(1-e^{-1/2})(1-(1-m)x)\}+1/(2r)+\varepsilon'\ .$$

Theorem E.8 now follows from the last corollary by choosing $\varepsilon' = \varepsilon/2$, $r = \lceil \varepsilon^{-1} \rceil$ and

$$\alpha = \arg\min_{\alpha'\in[0,1]}\max_{x\in[0,1/2]}\{\alpha'(mx^2+2x-2x^2)+2(1-\alpha')(1-e^{-1/2})(1-(1-m)x)\}\ .$$



(a) Results for Personalized Image Summarization with a cardinality constraint for varying number of images in the summary produced.

(b) Results for Personalized Image Summarization with a matroid constraint. The x-axis is the number k of images allowed from each category.

Figure 3: Personalized Image Summerization Results

F Personalized Image Summarization

Consider a setting in which we get as input a collection $\mathcal{N}$ of images from ℓ disjoint categories (e.g., birds, dogs, cats) and the user specifies $r \in [\ell]$ categories, and then demands a subset of the images in these categories that summarize all the images of the categories. Following [40] again, to evaluate a given subset of images we use the function $f(S) = \sum_{u \in \mathcal{N}} \max_{v \in S} s_{u,v} - \frac{1}{|\mathcal{N}|} \sum_{u \in S} \sum_{v \in S} s_{u,v}$, where $s_{u,v}$ is a non-negative similarity between u and v.

One can verify that the above function f is non-negative and submodular. Unfortunately, this function can have a very low monotonicity ratio. To compensate for this, we observe that most the analyses we described in the previous sections use the monotonicity ratio only to show that $f(S \cup T) \geq m \cdot f(S)$ for sets S and T that are feasible. This motivates the following weak version of the monotonicity ratio. We note that many continuous properties of set functions have such weak versions. For example, the original paper presenting the submodularity-ratio [16] presented in fact the weak version of this property, and the non-weak version was only formulated at a later point.

Definition F.1. Consider maximization of a non-negative function f subject to some constraint. In the context of this constraint, we say that f is *m-weakly monotone* if $f(S \cup T) \geq m \cdot f(S)$ holds for every two feasible sets S and T.

Theorem F.2. *The objective function f of personalized image summarization is $1 - \frac{2k}{|\mathcal{N}|}$-weakly monotone when the size of feasible solutions is at most k for some $1 \leq i \leq |\mathcal{N}|$.*

Proof. When $k \geq |\mathcal{N}|/2$, the theorem is trivial. Thus, we can assume below $k < |\mathcal{N}|/2$. Consider two feasible sets $S, T \in \mathcal{N}$, and let us lower bound $f(S \cup T)$.

$$\begin{aligned}
f(S \cup T) &= \sum_{u \in \mathcal{N}} \max_{v \in S \cup T} s_{u,v} - \frac{1}{|\mathcal{N}|} \sum_{u \in S \cup T} \sum_{v \in S \cup T} s_{u,v} \\
&\geq \sum_{u \in \mathcal{N}} \max_{v \in S \cup T} s_{u,v} - \frac{|S \cup T|}{|\mathcal{N}|} \sum_{u \in S \cup T} \max_{v \in S \cup T} s_{u,v} \\
&\geq \sum_{u \in \mathcal{N}} \max_{v \in S \cup T} s_{u,v} - \frac{2k}{|\mathcal{N}|} \sum_{u \in S \cup T} \max_{v \in S \cup T} s_{u,v} \\
&= \left(1 - \frac{2k}{|\mathcal{N}|}\right) \sum_{u \in \mathcal{N}} \max_{v \in S \cup T} s_{u,v} \geq \left(1 - \frac{2k}{|\mathcal{N}|}\right) \sum_{u \in \mathcal{N}} \max_{v \in S} s_{u,v} \quad .
\end{aligned}$$

Using this lower bound, we now get

$$f(S) = \sum_{u \in E} \max_{v \in S} s_{u,v} - \frac{1}{|\mathcal{N}|} \sum_{u \in S} \sum_{v \in S} s_{u,v} \leq \sum_{u \in E} \max_{v \in S} s_{u,v} \leq \frac{f(S \cup T)}{1 - 2k/|\mathcal{N}|} \quad ,$$

which completes the proof of the theorem since S and T have been chosen as arbitrary feasible sets. $\square$

Our experiments for this setting are based on a subset of the CIFAR-10 dataset [33] including $10{,}000$ Tiny Images. These images belong to 10 classes, with 1000 images per class. Each image consists of 32×32 RGB pixels represented by a 3072 dimensional vector. To compute the similarity $s_{u,v}$ between images, we used the dot product.

In our first experiment, we simply looked for a summary consisting of a limited number of images. Since this is a cardinality constraint, we again used the scarecrow algorithm Random and the accelerated versions mentioned in Section 6.1 of the algorithms from Section 4. In Figure 3a we depict the outputs of Threshold Random Greedy and Random for various limits on the number of images in the summary (like in Section 6.1 we omit the other non-scarecrow algorithms since their performance is essentially identical to the one of Threshold Random Greedy, and we refer the reader to Appendix H for more detail). Figure 3a also includes the upper bounds on the optimal solution obtained via the previous approximation ratio for Random Greedy and our improved approximation ratio (the area between the two upper bounds is shaded). We can see that the upper bound obtained via our improved approximation ratio is much tighter, and this upper bound also demonstrates that the gap between the non-scarecrow and the scarecrow algorithms is significant compared to the optimal solution.

In our second experiment, we looked for a summary containing up to k images from each category selected by the user for some parameter k (we assumed in the experiment that the user chose the categories: "airplane", "automobile" and "bird"). Since this is a (generalized partition) matroid constraint, in this experiment we used versions of the algorithms from Section 5. Specifically, we used Random Greedy for Matroids and an accelerated version of Measured Continuous Greedy based on the acceleration technique underlying the Accelerated Continuous Greedy of [1]. Additionally, we used in this experiment a scarecrow algorithm called Random that outputs a set containing a random selection of k images from each one of the chosen categories. The values of the outputs of all these algorithms are depicted in Figure 3b (values shown are averaged over 10 executions).

Figure 3b also includes upper bounds on the value of the optimal solution. The previous upper bound is computed based on the previously known approximation ratios of the algorithms, and our upper bound is computed based on the approximation ratios proved in Theorems 5.2 and 5.3 and the weak monotonicity ratio proved in Theorem F.2.[7] As is evident from the similarity between Figures 3a and 3b, our observations from the first experiment extend also the more general constraint considered in the current experiment.

G Proofs of Section 6

In this section we prove the theorems from Section 6.

G.1 Proof of Theorem 6.1

Theorem 6.1. *The objective function f is monotone for $0 \leq \lambda \leq 1/2$ and $2(1-\lambda)$-monotone for $1/2 \leq \lambda \leq 1$.*

Proof. We first prove the first part of the theorem. Thus, we assume $\lambda \leq 1/2$, and we need to show that for arbitrary set $S \subseteq \mathcal{N}$ and element $u \in \mathcal{N} \setminus S$ the marginal contribution $f(u \mid S)$ is non-negative. This holds because

$$\begin{aligned} f(u \mid S) &= \sum_{v \in \mathcal{N}} s_{v,u} - \lambda \left[\sum_{v \in S} s_{u,v} + \sum_{v \in S} s_{v,u} + s_{u,u} \right] \\ &= \sum_{v \in \mathcal{N}} s_{v,u} - \lambda \left[2 \sum_{v \in S} s_{v,u} + s_{u,u} \right] \geq \sum_{v \in \mathcal{N}} s_{v,u} - \sum_{v \in S} s_{v,u} - s_{u,u} \geq 0 \ , \end{aligned}$$

where the second equality holds because $s_{u,v} = s_{v,u}$, and the first inequality holds since $\lambda \leq 1/2$ in the case we consider and the $s_{u,v}$ values are non-negative.

[7] From a purely formal point of view this upper bound is not fully justified since Measured Continuous Greedy is a rare example of an algorithm whose analysis cannot use in a black box fashion the weak monotonicity ratio instead of the monotonicity ratio. However, due to probabilistic concentration, we expect the upper bound to still hold up to at most a small error.

It remains to prove the second part of the theorem. Thus, we assume from now on $\lambda \in [1/2, 1]$, and we consider two sets $S \subseteq T \subseteq \mathcal{N}$. To prove the theorem we need to show that $f(T) \geq 2(1-\lambda) \cdot f(S)$. The first step towards showing this is to prove the following lower bound on $f(S)$.

$$\begin{aligned} f(S) &= 2(1-\lambda) \cdot f(S) + (2\lambda - 1) \cdot \left[\sum_{u \in \mathcal{N}} \sum_{v \in S} s_{u,v} - \lambda \sum_{u \in S} \sum_{v \in S} s_{u,v} \right] && (10) \\ &\geq 2(1-\lambda) \cdot f(S) + (2\lambda - 1) \cdot \sum_{u \in T \setminus S} \sum_{v \in S} s_{u,v} \\ &= 2(1-\lambda) \cdot f(S) + (2\lambda - 1) \cdot \sum_{u \in S} \sum_{v \in T \setminus S} s_{u,v} \ , && (11) \end{aligned}$$

where the inequality holds since $\lambda \leq 1$, and the second equality holds since $s_{u,v} = s_{v,u}$. Using this lower bound, we now get

$$\begin{aligned} f(T) &= f(S) + \sum_{u \in \mathcal{N}} \sum_{v \in T \setminus S} s_{u,v} - \lambda \left[\sum_{u \in S} \sum_{v \in T \setminus S} s_{u,v} + \sum_{u \in T \setminus S} \sum_{v \in S} s_{u,v} + \sum_{u \in T \setminus S} \sum_{v \in T \setminus S} s_{u,v} \right] \\ &= f(S) + \sum_{u \in \mathcal{N}} \sum_{v \in T \setminus S} s_{u,v} - \lambda \left[2 \sum_{u \in S} \sum_{v \in T \setminus S} s_{u,v} + \sum_{u \in T \setminus S} \sum_{v \in T \setminus S} s_{u,v} \right] \\ &\geq f(S) + (1 - 2\lambda) \cdot \sum_{u \in S} \sum_{v \in T \setminus S} s_{u,v} \geq 2(1-\lambda) \cdot f(S) \ , \end{aligned}$$

where the first inequality holds since $\lambda \leq 1$, and the second inequality holds by Inequality (10). □

G.2 Proof of Theorem 6.2

Theorem 6.2. *For $\beta \in (0, 1/2)$, the objective function F given by Equation (1) is $\frac{(1-2\beta)\cdot\alpha}{1+\alpha}$-monotone. Furthermore, when $\min_{\bar{0} \leq x \leq \mathbf{u}} (\frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{h}\mathbf{x}) \geq 0$, F is even $(1-2\beta)$-monotone.*

Proof. Fix two vectors $\bar{0} \leq \mathbf{x} \leq \mathbf{y} \leq \mathbf{u}$. We begin this proof by providing a lower bound on $F(\mathbf{y})$ and an upper bound on $F(\mathbf{x})$. The lower bound on $F(\mathbf{y})$ is as following.

$$F(\mathbf{y}) = \frac{1}{2}\mathbf{y}^T \mathbf{H} \mathbf{y} + \mathbf{h}^T \mathbf{y} + c \geq \min_{\bar{0} \leq \mathbf{x} \leq \mathbf{u}} \left(\frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{h}\mathbf{x} \right) + c \ .$$

To get the upper bound on $F(\mathbf{x})$, we first need to prove an upper bound on c.

$$c \geq -\min_{0 \leq \mathbf{x} \leq \mathbf{u}} \left(\frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{h}^T \mathbf{x} \right) = -\min_{0 \leq \mathbf{x} \leq \mathbf{u}} \left(\frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} - \beta \mathbf{u}^T \mathbf{H} \mathbf{x} \right) \geq -\left(\frac{1}{2} - \beta \right) \mathbf{u}^T \mathbf{H} \mathbf{u} \ .$$

The promised upper bound on $F(\mathbf{x})$ now follows.

$$F(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{h}^T \mathbf{x} + c \leq \mathbf{h}^T \mathbf{x} + c \leq \mathbf{h}^T \mathbf{u} + c = -\beta \mathbf{u}^T \mathbf{H} \mathbf{u} + c \leq \frac{\beta c}{1/2 - \beta} + c = \frac{c}{1 - 2\beta} \ ,$$

where the first inequality holds since $\mathbf{H}$ is non-positive, and the second inequality holds since $\mathbf{h}$ is non-negative.

Recall now that $c = -M + \alpha|M|$, which implies

$$\min_{0 \leq \mathbf{x} \leq \mathbf{u}} \left(\frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{h}^T \mathbf{x} \right) = M \geq -\frac{c}{1+\alpha} \ ,$$

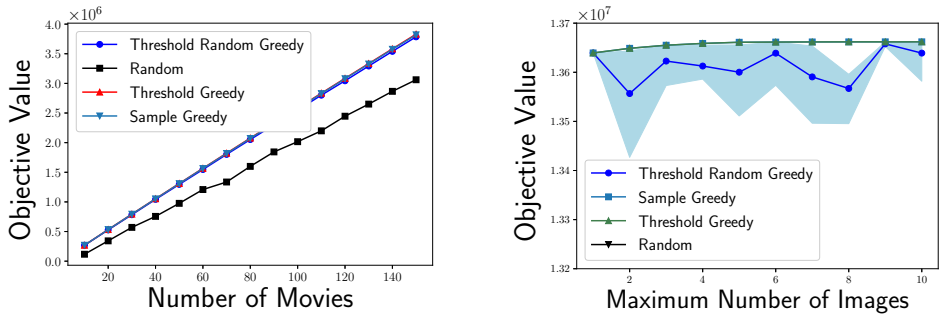
and therefore,

$$F(\mathbf{y}) \geq -\frac{c}{1+\alpha} + c = \frac{c\alpha}{1+\alpha} \geq \frac{(1-2\beta)\alpha}{1+\alpha} \cdot F(\mathbf{x}) \ .$$

It remains to consider the case in which $\min_{\bar{0} \leq \mathbf{x} \leq \mathbf{u}} \left(\frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{h}\mathbf{x} \right) \geq 0$. In this case

$$F(\mathbf{y}) \geq c \geq (1 - 2\beta) \cdot F(\mathbf{x}) \ .$$

□



(a) Performance of the various algorithms in the movie recommendation setting (for $\lambda = 0.75$).

(b) Performance of the non-scarecrow algorithms in the image summarization setting with a cardinality constraint. The shaded area represents the standard deviation of Threshold Random Greedy.

Figure 4: Comparing the performance of algorithms for a cardinality constraint in our experiments.

H Additional Plots for Section 6

As discussed in Section 6, the various algorithms we use in the context of a cardinality constraint have very similar empirical performance. Figure 4a presents the performance of all these algorithms in the movie recommendation setting with the number of movies in the summery varying. One can observe that the lines of the three non-scarecrow algorithms almost overlap. Figure 4b presents the performance of the non-scarecrow algorithms in the image summarization setting. In this figure we had to ignore the scarecrow algorithm Random because otherwise the lines of the three non-scarecrows algorithms are indistinguishable. Furthermore, we had to zoom in on a very small range of y-axis values. Despite these steps, the lines of Sample Greedy and Threshold Greedy still completely overlap, but the large zoom allows us to see that Threshold Random Greedy is marginally worse.

I Maximizating DR-submodular Functions subject to a Polytope Constraint

There are (at least) two natural ways in which the notion of submodularity can be extended from set functions to continuous functions. The more restrictive of these is known as DR-submodularity (first defined by [3]). Given a domain $\mathcal{X} = \prod_{i=1}^{n} \mathcal{X}_i$, where $\mathcal{X}_i$ is a closed range in $\mathbb{R}$ for every $i \in [n]$, a function $F\colon \mathcal{X} \to \mathbb{R}$ is *DR-submodular* if for every two vectors $\mathbf{a}, \mathbf{b} \in \mathcal{X}$, positive value k and coordinate $i \in [n]$ the inequality

$$F(\mathbf{a} + k\mathbf{e}_i) - F(\mathbf{a}) \geq F(\mathbf{b} + k\mathbf{e}_i) - F(\mathbf{b})$$

holds whenever $\mathbf{a} \leq \mathbf{b}$ and $\mathbf{b} + k\mathbf{e}_i \in \mathcal{X}$ (here and throughout the section $\mathbf{e}_i$ denotes the standard i-th basis vector, and comparison between two vectors should be understood to hold coordinate-wise). If F is continuously differentiable, then the above definition of DR-submodulrity is equivalent to ∇F being an antitone mapping from $\mathcal{X}$ to $\mathbb{R}^n$ (i.e., $\nabla F(\mathbf{a}) \geq \nabla F(\mathbf{b})$ for every two vectors $\mathbf{a}, \mathbf{b} \in \mathcal{X}$ that obey $\mathbf{a} \leq \mathbf{b}$). Moreover, when F is twice differentiable, it is DR-submodular if and only if its Hessian is non-positive at every vector $\mathbf{x} \in \mathcal{X}$.

In this section we consider the problem of maximizing a non-negative DR-submodular function $F\colon 2^{\mathcal{N}} \to \mathbb{R}_{\geq 0}$ subject to a solvable down-closed[8] convex body $P \subseteq \mathcal{X}$ (usually polytope) constraint. As is standard when dealing with problems of this kind, we assume that F is L-smooth, i.e., for every two vectors $\mathbf{x}, \mathbf{y} \in \mathcal{X}$ it obeys

$$\|\nabla F(\mathbf{x}) - \nabla F(\mathbf{y})\|_2 \leq L\|\mathbf{x} - \mathbf{y}\|_2$$

[8] In Section E.2, down-closeness of was defined for the special case of $P \subseteq [0,1]^{\mathcal{N}}$. More generally, a body $P \subseteq \mathcal{X}$ is down-closed if $\mathbf{b} \in P$ implies $\mathbf{a} \in P$ for every vector $\mathbf{a} \in \mathcal{X}$ obeying $\mathbf{a} \leq \mathbf{b}$.

for some non-negative parameter L. Additionally, for simplicity, we assume that $\mathcal{X} = [0,1]^n$. This assumption is without loss of generality because the natural mapping from $\mathcal{X}$ to $[0,1]^n$ preserves all our results.

We analyze a variant of the Frank-Wolfe algorithm for the above problem due to [2] called Non-monotone Frank-Wolfe. This variant was motivated by the Measured Continuous Greedy algorithm studied in Section E.2, and its assumes access to the first order derivatives of F. The details of the algorithm we consider appear as Algorithm 6. This algorithm gets a quality control parameter $\varepsilon \in (0,1)$, and it is assumed that ε^{-1} is an integer (if this is not the case, one can fix that by reducing ε by at most a factor 2). Algorithm 6 and its analysis also employ the notation defined in Section E.2, namely, given two vectors $\mathbf{x}, \mathbf{y}$, their coordinate-wise multiplication is denoted by $\mathbf{x} \odot \mathbf{y}$. Additionally, we denote by $\bar{0}$ and $\bar{1}$ the all zeros and all ones vectors, respectively.

Algorithm 6: Non-monotone Frank-Wolfe(ε)

Let $\mathbf{y}^{(0)} \leftarrow \bar{0}$ and $t = 0$.
while $t \leq 1$ **do**
  $\mathbf{s}^{(t)} \leftarrow \arg\max_{\mathbf{x} \in P} \mathbf{x} \cdot ((\bar{1} - \mathbf{y}^{(t)}) \odot \nabla F(\mathbf{y}^{(t)}))$.
  $\mathbf{y}^{(t+\varepsilon)} \leftarrow \mathbf{y}^{(t)} + \varepsilon \cdot (\bar{1} - \mathbf{y}^{(t)}) \odot \mathbf{s}^{(t)}$.
  $t \leftarrow t + \varepsilon$.
return $\mathbf{y}^{(1)}$.

To analyze Algorithm 6 we need to define two additional parameters. The first parameter is the diameter $D = \max_{x \in P} \|\mathbf{x}\|_2$ of P, which is a standard parameter. The other parameter is the monotonicity ratio of F, which can be extended to the continuous setting we study in the following natural way.[9]

$$m = \inf_{\substack{\mathbf{x},\mathbf{y} \in \mathcal{X} \\ \mathbf{x} \leq \mathbf{y}}} \frac{F(\mathbf{y})}{F(\mathbf{x})} \ ,$$

where the ratio $F(\mathbf{y})/F(\mathbf{x})$ should be understood to have a value of 1 whenever $F(\mathbf{x}) = 0$. Additionally, let us denote by $\mathbf{o}$ an arbitrary optimal solution for the problem described above. Using these definitions, we are now ready to state the result that we prove for Algorithm 6.

Theorem I.1. *When given a non-negative m-monotone DR-submodular function $F \colon \mathcal{X} \to \mathbb{R}_{\geq 0}$ and a down-closed solvable convex body $P \subseteq \mathcal{X}$, the Measured Greedy Frank-Wolfe algorithm (Algorithm 6) outputs a solution $\mathbf{y} \in P$ such that $F(\mathbf{y}) \geq [m(1-1/e)+(1-m)\cdot(1/e)]\cdot F(\mathbf{o}) - \varepsilon L D^2$.*

Our first objective towards proving Theorem I.1 is to lower bound the expression $F(\mathbf{o} + \mathbf{y}^{(t)} \cdot (\bar{1} - \mathbf{o}))$, which we do in the next two lemmata.

Lemma I.2. *For every integer $i \in [0, \varepsilon^{-1}]$, $\mathbf{y}^{(\varepsilon i)} \geq \bar{0}$ and $\|\mathbf{y}^{(\varepsilon i)}\|_\infty \leq 1 - (1-\varepsilon)^{-i}$.*

Proof. We prove the lemma by induction on i. For $i = 0$, the lemma follows directly from the initialization $\mathbf{y}^{(0)} = \bar{0}$ because $1 - (1-\varepsilon)^{-0} = 0$. Assume now that the lemma holds for $i-1$, and let us prove it for an integer $0 < i \leq 1$. Observe that, for every $j \in [n]$,

$$y_j^{\varepsilon i} = y_j^{\varepsilon(i-1)} + \varepsilon \cdot \left(1 - y_j^{\varepsilon(i-1)}\right) \cdot s_j^{\varepsilon(i-1)} \geq y_j^{\varepsilon(i-1)} \geq 0 \ ,$$

where the first inequality holds since $y_j^{\varepsilon(i-1)} \leq 1$ by the induction hypothesis and the value of $s_j^{(\varepsilon(i-1))}$ is non-negative by definition. Moreover,

$$\begin{aligned} y_j^{\varepsilon i} &= y_j^{\varepsilon(i-1)} + \varepsilon \cdot \left(1 - y_j^{\varepsilon(i-1)}\right) \cdot s_j^{\varepsilon(i-1)} \leq y_j^{\varepsilon(i-1)} + \varepsilon \cdot \left(1 - y_j^{\varepsilon(i-1)}\right) \\ &= \varepsilon + (1-\varepsilon) \cdot y_j^{\varepsilon(i-1)} \leq \varepsilon + (1-\varepsilon) \cdot \left[1 - (1-\varepsilon)^{(i-1)}\right] = 1 - (1-\varepsilon)^i \ , \end{aligned}$$

where again the first inequality holds since $s^{(\varepsilon(i-1))} \in \mathcal{X}$, which implies $s_j^i \leq 1$; and the second inequality holds by the induction hypothesis. □

[9] In Appendix 6.2 we showed how the monotonicity ratio can be extended to the particular continuous setting studied in that section. The definition of Appendix 6.2 is obtained from the more general definition we give here by setting $\mathcal{X} = \prod_{i=1}^n [0, u_i]$.

Lemma I.3. *For every integer* $i \in [0, \varepsilon^{-1}]$, $F(\mathbf{o}+\mathbf{y}^{(\varepsilon i)} \cdot (\bar{1}-\mathbf{o})) \geq \left[(1-(1-m)\left(1-(1-\varepsilon)^i\right)\right] \cdot F(\mathbf{o}) = \left[m + (1-m)(1-\varepsilon)^i\right] \cdot F(\mathbf{o})$.

Proof. Observe that

$$\begin{aligned}
F(\mathbf{o} + \mathbf{y}^{(\varepsilon i)} \cdot (\bar{1} - \mathbf{o})) &\geq \left(1 - \|\mathbf{y}^{(\varepsilon i)}\|_\infty\right) \cdot F(\mathbf{o}) + \|\mathbf{y}^{(\varepsilon i)}\|_\infty \cdot F\left(\mathbf{o} + \frac{\mathbf{y}^{(\varepsilon i)} \cdot (\bar{1} - \mathbf{o})}{\|\mathbf{y}^{(\varepsilon i)}\|_\infty}\right) \\
&\geq \left(1 - \|\mathbf{y}^{(\varepsilon i)}\|_\infty\right) \cdot F(\mathbf{o}) + m \cdot \|\mathbf{y}^{(\varepsilon t)}\|_\infty \cdot F(\mathbf{o}) \\
&= \left(1 - (1-m) \cdot \|\mathbf{y}^{(\varepsilon i)}\|_\infty\right) \cdot F(\mathbf{o}) \ ,
\end{aligned}$$

where the first inequality holds since the DR-submodularity of F implies that F is concave along positive directions (such as the direction $\mathbf{y}^{(\varepsilon i)} \cdot (\bar{1} - \mathbf{o})/\|\mathbf{y}^{(\varepsilon i)}\|_\infty$), and the second inequality holds since the monotonicity ratio of F is at least m. Plugging Lemma I.2 into the previous inequality completes the proof of the lemma. □

Using the previous lemma, we can now provide a lower bound on the increase in the value of $\mathbf{y}^{(t)}$ as a function of t.

Lemma I.4. *For every integer* $0 \leq i < \varepsilon^{-1}$, $F(\mathbf{y}^{(\varepsilon(i+1))}) - F(\mathbf{y}^{(\varepsilon i)}) \geq \varepsilon \cdot [(m + (1-m) \cdot (1-\varepsilon)^i) \cdot F(\mathbf{o}) - F(\mathbf{y}^{(\varepsilon i)})] - \varepsilon^2 L D^2$.

Proof. By the chain rule,

$$\begin{aligned}
F(\mathbf{y}^{\varepsilon(i+1)}) - F(\mathbf{y}^{(\varepsilon i)}) &= F(\mathbf{y}^{(\varepsilon i)} + \varepsilon \cdot \mathbf{s}^{(\varepsilon i)} \odot (\bar{1} - \mathbf{y}^{(\varepsilon i)})) - F(\mathbf{y}^{(\varepsilon i)}) \\
&= \int_0^\varepsilon \nabla F(\mathbf{y}^{(\varepsilon i)} + r \cdot \mathbf{s}^{(\varepsilon i)} \odot (\bar{1} - \mathbf{y}^{(\varepsilon i)})) \cdot (\mathbf{s}^{(\varepsilon i)} \odot (\bar{1} - \mathbf{y}^{(\varepsilon i)}))\, dr \\
&\geq \int_0^\varepsilon \nabla F(\mathbf{y}^{(\varepsilon i)}) \cdot (\mathbf{s}^{(\varepsilon i)} \odot (\bar{1} - \mathbf{y}^{(\varepsilon i)}))\, dr - \varepsilon^2 L D^2 \\
&= \varepsilon \cdot \nabla F(\mathbf{y}^{(\varepsilon i)}) \cdot (\mathbf{s}^{(\varepsilon i)} \odot (1 - \mathbf{y}^{(\varepsilon i)})) - \varepsilon^2 L D^2 \ ,
\end{aligned}$$

where the first inequality holds by the L-smoothness of F. Furthermore,

$$\begin{aligned}
\nabla F(\mathbf{y}^{(\varepsilon i)}) \cdot (\mathbf{s}^{(\varepsilon i)} \odot (\bar{1} - \mathbf{y}^{(\varepsilon i)})) &= ((\bar{1} - \mathbf{y}^{(\varepsilon i)}) \odot \nabla F(\mathbf{y}^{(\varepsilon i)})) \cdot \mathbf{s}^{(\varepsilon i)} \\
&\geq ((\bar{1} - \mathbf{y}^{(\varepsilon i)}) \odot \nabla F(\mathbf{y}^{(\varepsilon i)})) \cdot \mathbf{o} \\
&= \nabla F(\mathbf{y}^{(\varepsilon i)})) \cdot ((\bar{1} - \mathbf{y}^{(\varepsilon i)}) \odot \mathbf{o}) \\
&\geq F(\mathbf{o} + \mathbf{y}^{(\varepsilon i)}(\bar{1} - \mathbf{o})) - F(\mathbf{y}^{(\varepsilon i)}) \\
&\geq \left[m + (1-m) \cdot (1-\varepsilon)^i\right] \cdot F(\mathbf{o}) - F(\mathbf{y}^{(\varepsilon i)}) \ ,
\end{aligned}$$

where the first inequality holds by the definition of $\mathbf{s}^{(\varepsilon i)}$ since $\mathbf{o}$ is a candidate to be this vector, the second inequality follows from the concavity of F along positive directions, and the last inequality holds by Lemma I.3. The lemma now follows by combining the two above inequalities. □

We are now ready to prove Theorem I.1.

Proof of Theorem I.1. Rearranging the guarantee of Lemma I.4, we get

$$F(\mathbf{y}^{\varepsilon(i+1)}) \geq (1-\varepsilon) \cdot F(\mathbf{y}^{(\varepsilon i)}) + \varepsilon[m + (1-m) \cdot (1-\varepsilon)^i] \cdot F(\mathbf{o}) - \varepsilon^2 L D^2 \ .$$

Since this inequality applies for every integer $0 \leq i < \varepsilon^{-1}$, we can use it repeatedly to obtain

$$\begin{aligned}
F(\mathbf{y}^{(1)}) &\geq \varepsilon \cdot \sum_{i=1}^{1/\varepsilon} (1-\varepsilon)^{1/\varepsilon - i} \cdot \left[(m + (1-m) \cdot (1-\varepsilon)^{i-1}) \cdot F(\mathbf{o}) - \varepsilon L D^2\right] + (1-\varepsilon)^{1/\varepsilon} \cdot F(\bar{0}) \\
&\geq m\varepsilon \cdot \sum_{i=1}^{1/\varepsilon} (1-\varepsilon)^{\frac{1}{\varepsilon} - i} \cdot F(\mathbf{o}) + \varepsilon(1-m) \cdot \sum_{i=1}^{1/\varepsilon} [(1-\varepsilon)^{1/\varepsilon - 1} \cdot F(\mathbf{o}) - \varepsilon L D^2]
\end{aligned}$$

$$
\begin{aligned}
&= m\varepsilon \cdot \frac{1-(1-\varepsilon)^{1/\varepsilon}}{\varepsilon} \cdot F(\mathbf{o}) + \varepsilon(1-m) \cdot \frac{(1-\varepsilon)^{1/\varepsilon-1} \cdot F(\mathbf{o}) - \varepsilon L D^2}{\varepsilon} \\
&\geq \left[m(1-e^{-1}) + (1-m) \cdot e^{-1}\right] \cdot F(\mathbf{o}) - \varepsilon L D^2 \ ,
\end{aligned}
$$

where the second inequality follows from the non-negativity of F, and the last inequality holds since $(1-\varepsilon)^{1/\varepsilon} \leq e^{-1} \leq (1-\varepsilon)^{1/\varepsilon-1}$. $\square$