Weakly Submodular Function Maximization Using Local Submodularity Ratio

Richard Santiago
ETH Zürich, Switzerland
rtorres@ethz.ch

Yuichi Yoshida
National Institute of Informatics, Tokyo, Japan
yyoshida@nii.ac.jp

Abstract

Weak submodularity is a natural relaxation of the diminishing return property, which is equivalent to submodularity. Weak submodularity has been used to show that many (monotone) functions that arise in practice can be efficiently maximized with provable guarantees. In this work we introduce two natural generalizations of weak submodularity for non-monotone functions. We show that an efficient randomized greedy algorithm has provable approximation guarantees for maximizing these functions subject to a cardinality constraint. We then provide a more refined analysis that takes into account that the weak submodularity parameter may change (sometimes improving) throughout the execution of the algorithm. This leads to improved approximation guarantees in some settings. We provide applications of our results for monotone and non-monotone maximization problems.

2012 ACM Subject Classification Theory of computation → Approximation algorithms analysis

Keywords and phrases weakly submodular, non-monotone, local submodularity ratio

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2020.64

Related Version A full version of the paper can be found in [36], `https://arxiv.org/abs/2004.14650`.

Acknowledgements Part of this work was conducted during the first author's visit to the National Institute of Informatics in Tokyo. The second authors was supported by JSPS KAKENHI Grant Number 18H05291.

1 Introduction

Submodularity is a property of set functions equivalent to the notion of diminishing returns. More formally, we say that a set function $f : 2^E \to \mathbb{R}$ is *submodular* if for any two sets $A \subseteq B \subseteq E$ and an element $e \notin B$, the corresponding marginal gains satisfy $f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$. Submodularity has found a wide range of connections and applications to different computer science areas in recent years.

However, many applications in practice does not satisfy the diminishing returns property, but rather a weaker version of it. This has motivated several lines of work exploring different ways to relax the submodularity property [11, 18, 10, 28, 17, 21, 22]. One such relaxation that has received a lot of attention from the machine learning community is the notion of weak submodularity (we postpone the formal definition to Section 1.1), originally introduced by Das and Kempe [11]. They provided applications to the feature selection and the dictionary selection problems, and showed that the standard greedy algorithm achieves a $(1 - e^{-\gamma})$-approximation for the monotone maximization problem subject to a cardinality constraint. Here the parameter $\gamma \in [0, 1]$ is called the submodularity ratio, and it measures how "close" the function is to being submodular. Weak submodularity has found applications in areas

31st International Symposium on Algorithms and Computation (ISAAC 2020).
Editors: Yixin Cao, Siu-Wing Cheng, and Minming Li; Article No. 64; pp. 64:1–64:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

such as linear and nonlinear sparse regression [14, 29], high-dimensional subset selection [15], interpretability of black-box neural network classifiers [14], video summarization, splice site detection, and black-box interpretation of images [9].

In subsequent work, Das and Kempe [12] left as an open question whether some of these theoretical guarantees can be extended to non-monotone objectives. As their definition of weak submodularity is targeted at monotone functions, they raise the question of whether there is a more general definition that retains some of the positive results of their work, while also yielding an analogue to non-monotone objectives.

One main goal of this work is to answer that question. We believe this is interesting for both theoretical and practical purposes, given that non-monotone submodular objectives have found a wide range of applications in computer science. Some of these include document summarization [32, 33], MAP inference for determinantal point processes [24], personalized data summarization [34], nonparametric learning [39], image summarization [37], and removing redundant elements from DNA sequencing [31]. Hence, it seems natural to study how the approximation guarantees for non-monotone submodular maximization degrade in terms of the submodularity ratio.

In this work we introduce a natural generalization of weak submodularity to the non-monotone setting. We then show that a fast and simple randomized greedy algorithm retains some of the good theoretical guarantees available for (non-monotone) submodular objectives. In addition, for monotone weakly submodular functions, this algorithm retains the approximation guarantee of $1 - e^{-\gamma}$ given in [11].

A second main contribution of our work is to provide a more refined analysis that takes into account that the submodularity ratio parameter may change (some times improving) throughout the execution of the algorithm. We provide several applications where this more refined bound leads to improved approximation guarantees, for both monotone and non-monotone maximization problems.

The rest of this section is organized as follows. In Section 1.1 we extend weak submodularity to the non-monotone setting. In Section 1.2 we discuss the notion of local submodularity ratio. We discuss several examples and applications in Section 1.3. Our main contributions are presented in Section 1.5. Additional related work regarding weak submodularity and non-monotone submodular maximization is discussed in Section 1.4.

1.1 Weak submodularity and non-monotonicity

Throughout this paper we use $f_A(B)$ to denote the marginal gain of adding the set B to A, that is $f(A \cup B) - f(A)$. A non-negative monotone set function $f : 2^E \to \mathbb{R}_+$ is *γ-weakly submodular* for some parameter $0 \leq \gamma \leq 1$, if for any pair of disjoint sets $A, B \subseteq E$, it satisfies $\sum_{e \in B} f_A(e) \geq \gamma \cdot f_A(B)$. We note that this is the definition used in [2, 9, 14], which is slightly adapted from the original definition given in [11, 12]. The parameter γ is called the *submodularity ratio*.

When f is monotone, it is clear that for any value of $\gamma \in [0, 1]$ the above class contains monotone submodular functions. However, for non-monotone objectives the marginal gains can be negative, and in this case we have $\gamma f_A(B) \geq f_A(B)$ whenever $f_A(B) \leq 0$, leading to a stronger condition than diminishing returns. This motivates us to introduce the following two classes of non-monotone non-submodular functions.

▶ **Definition 1.1** (pseudo and weak submodularity). *Given a scalar $0 < \gamma \leq 1$, we say that a set function $f : 2^E \to \mathbb{R}_+$ is:*

1. *γ-pseudo submodular if $\sum_{e \in B} f_A(e) \geq \gamma f_A(B)$ for any pair of disjoint sets $A, B \subseteq E$.*
2. *γ-weakly submodular if $\sum_{e \in B} f_A(e) \geq \min\{\gamma f_A(B), \frac{1}{\gamma} f_A(B)\}$ for any $A, B \subseteq E$ disjoint.*

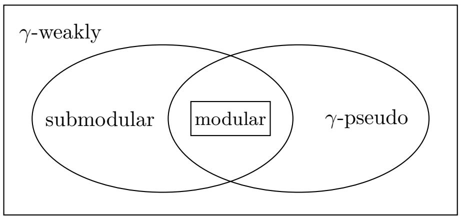

Figure 1 Hierarchy of the different function classes.

We first note that for monotone functions, the above two definitions are equivalent to the notion of γ-weakly submodularity from previous works [2, 9, 14]. This follows immediately from the fact that monotone functions satisfy $f_A(B) \geq 0$ for all $A, B \subseteq E$, and hence $\min\{\gamma f_A(B), \frac{1}{\gamma} f_A(B)\} = \gamma f_A(B)$.

For any value $\gamma \in (0, 1]$ the above definition of γ-weakly submodularity leads to a weaker notion of diminishing returns (i.e., it contains non-monotone submodular functions). Indeed, if $f_A(B) \geq 0$ we have $\sum_{e \in B} f_A(e) \geq \gamma f_A(B)$, while if $f_A(B) < 0$ we have $\sum_{e \in B} f_A(e) \geq \frac{1}{\gamma} f_A(B)$. On the other hand, while the class of γ-pseudo submodular functions does not properly contain non-monotone submodular functions, it does contain functions that are not necessarily submodular. We show this in Figure 1.

1.2 Local submodularity ratio

The submodularity ratio γ is in general a very pessimistic bound for most applications. This is due to the fact that γ is defined as a *global* bound, in the sense that it must hold for any pair of disjoint sets $A, B \subseteq E$. Or at least for any pair of sets that are relevant to the execution of the algorithm, e.g., sets of cardinality at most k. We next discuss a natural way to "refine" this bound.

Given a function $f : 2^E \to \mathbb{R}_+$ and any pair of disjoint sets $A, B \subseteq E$, in this work we denote by $\gamma^f_{A,B}$ any non-negative scalar satisfying $\sum_{e \in B} f_A(e) \geq \gamma^f_{A,B} \cdot f_A(B)$. When it is clear from the context we usually simplify the notation to $\gamma_{A,B}$ instead of $\gamma^f_{A,B}$. One of our contributions is showing how using these local bounds can be beneficial in some settings. In particular we discuss several natural classes of functions for which (i) one can compute explicit bounds for the value $\gamma_{A,B}$ (see Section 1.3), and (ii) using the local bounds $\gamma_{A,B}$ (instead of γ) leads to significantly better theoretical guarantees (we discuss this in more detail in Section 1.5). We believe this is interesting for both theoretical and practical applications.

1.3 Examples and applications

In this section we present several classes of functions for which the parameter $\gamma_{A,B}$ can be bounded explicitly, and discuss applications arising from these results. Due to space limitations we postpone the proofs to the long version of the paper [36].

Our first example is the so-called metric diversity function (also known as remote clique). Here we are given a metric (i.e., a distance that satisfies the triangle inequality) $d : E \times E \to \mathbb{R}_+$ over a finite set E, where $d(u, v)$ measures the dissimilarity between two elements u and v. One then defines a set function $f(S) = \frac{1}{2} \sum_{u \neq v \in S} d(u, v)$ that measures the diversity inside the set S. The problem $\max d(S) : |S| \leq k$ of finding a diverse subset has been studied in the operations research community [27, 35, 3], and has found applications in other areas [1, 13].

▶ **Example 1.2.** Given a metric $d : E \times E \to \mathbb{R}_+$, consider the function given by $f(S) = \sum_{\{u,v\} \subseteq S} d(u,v)$, which is monotone and supermodular. Then, we have $\gamma_{A,B} \geq \frac{a}{a+b-1}$ for any two disjoint sets $A, B \subseteq E$, where $a = |A|$ and $b = |B|$.

The works of [6, 5] introduced the notion of proportionally submodular functions[1]. A set function $f : 2^E \to \mathbb{R}_+$ is *proportionally submodular* if $|S|f(T) + |T|f(S) \geq |S \cap T|f(S \cup T) + |S \cup T|f(S \cap T)$ for every $S, T \subseteq E$. In the monotone setting, this class properly contains monotone submodular functions. In addition, this class also contains some non-submodular objectives such as the (supermodular) metric diversity function discussed in Example 1.2. Since these functions are closed under addition, the sum of a monotone submodular function and a metric diversity function is proportionally submodular. Our next result bounds the parameter $\gamma_{A,B}$ for this class, in both the monotone and non-monotone settings.

▶ **Example 1.3.** A non-negative proportionally submodular function $f : 2^E \to \mathbb{R}_+$ has $\gamma_{A,B} \geq \frac{3a(1+a)}{3a^2+3ab+b^2-1}$ for any two disjoint sets $A, B \subseteq E$, where $a = |A|$ and $b = |B|$.

The above result leads to interesting applications. First it allows to improve over the current best approximation for maximizing a monotone proportionally submodular function subject to a cardinality constraint. In addition, combining this with other results from this work, we can also get improved approximations for the product $f \cdot g$ of a monotone submodular function f and a monotone proportionally submodular function g. We discuss this in more detail in Section 3.

▶ **Example 1.4.** Let $f, g : 2^E \to \mathbb{R}_+$ be two monotone set functions with parameters $\gamma^f_{A,B}$ and $\gamma^g_{A,B}$ respectively. Then the product function $h(S) := f(S) \cdot g(S)$ is also non-negative and monotone, with parameter

$$\gamma_{A,B} \geq \begin{cases} \frac{f(A)}{f(A \cup B)} \gamma^g_{A,B} & \text{if } \gamma^f_{A,B} \geq \gamma^g_{A,B} \\ \frac{g(A)}{g(A \cup B)} \gamma^f_{A,B} & \text{if } \gamma^g_{A,B} \geq \gamma^f_{A,B}, \end{cases}$$

for any two disjoint sets $A, B \subseteq E$. In particular, if f and g have global parameters γ^f and γ^g respectively, such that $\gamma^f \geq \gamma^g$, then the product function h has parameter $\gamma_{A,B} \geq \gamma^g \cdot \max\{\frac{f(A)}{f(A \cup B)}, \frac{g(A)}{g(A \cup B)}\}$.

Using that submodular functions satisfy $\gamma_{A,B} \geq 1$, we can combine the above result with Examples 1.2 and 1.3 to get the following.

▶ **Example 1.5.** Let $f, g : 2^E \to \mathbb{R}_+$ be two monotone functions, and let $h(S) := f(S) \cdot g(S)$ be the product function with parameter $\gamma_{A,B}$. Then we have the following.

(a) If f and g are submodular then $\gamma_{A,B} \geq \max\{\frac{f(A)}{f(A \cup B)}, \frac{g(A)}{g(A \cup B)}\}$.

(b) If f is submodular and g is the metric diversity function from Example 1.2, then $\gamma_{A,B} \geq \frac{f(A)}{f(A \cup B)} \cdot \frac{a}{a+b-1}$, where $a = |A|$ and $b = |B|$.

(c) If f is submodular and g is proportionally submodular then $\gamma_{A,B} \geq \frac{f(A)}{f(A \cup B)} \cdot \frac{3a(1+a)}{3a^2+3ab+b^2-1}$, where $a = |A|$ and $b = |B|$.

By taking a non-monotone submodular function f, and either multiplying it or dividing by the cardinality function, we obtain a new function that is no longer submodular. The next example bounds the parameter $\gamma_{A,B}$ for these functions.

[1] They called them weakly submodular at first, and changed the name in subsequent work.

▶ **Example 1.6.** Let $f : 2^E \to \mathbb{R}_+$ be a submodular function. Then for any two disjoint sets $A, B \subseteq E$ with $|A| = a$ and $|B| = b$ we have the following.

(a) The function $g(S) := |S| \cdot f(S)$ satisfies $\gamma_{A,B} \geq \frac{a+1}{a+b}$.

(b) The function $g(S) := \frac{f(S)}{|S|}$ has $\gamma_{A,B} \leq \frac{a+b}{a+1}$.

We next discuss the behavior of the parameter $\gamma_{A,B}$ under summation, and how this result allows us to generalize some of the bounds previously discussed in this section.

▶ **Proposition 1.7.** *Let $f, g : 2^E \to \mathbb{R}_+$ be two set functions with parameters $\gamma^f_{A,B}$ and $\gamma^g_{A,B}$ respectively. We have the following.*

(a) *If f and g are both monotone, then $f + g$ is also monotone with parameter $\gamma_{A,B} \geq \min\{\gamma^f_{A,B}, \gamma^g_{A,B}\}$. In particular, if $0 \leq \gamma^g_{A,B} \leq \gamma^f_{A,B}$ holds for all pairs of disjoint sets A and B, then $f + g$ has parameter $\gamma_{A,B} \geq \gamma^g_{A,B}$.*

(b) *If f is monotone and g is non-monotone, and $0 \leq \gamma^g_{A,B} \leq \gamma^f_{A,B}$ holds for all pairs of disjoint sets A and B, then $f + g$ has parameter $\gamma_{A,B} \geq \gamma^g_{A,B}$.*

By combining the above proposition with Examples 1.2, 1.6, and 1.3 we get the following.

▶ **Example 1.8.** Let f be a non-negative monotone submodular function. Then:

- The sum $f + g$ where g is a metric diversity function satisfies $\gamma_{A,B} \geq \frac{|A|}{|A|+|B|-1}$.
- The sum $f(S) + |S| \cdot g(S)$ where g is non-monotone submodular satisfies $\gamma_{A,B} \geq \frac{|A|+1}{|A|+|B|}$.
- The sum $f + g$ where g is non-monotone proportionally submodular satisfies $\gamma_{A,B} \geq \frac{3a(1+a)}{3a^2+3ab+b^2-1}$, where $a = |A|$ and $b = |B|$.

We can also combine the above result with Example 1.4 to get that the product function $(f + g) \cdot h$ satisfies $\gamma_{A,B} \geq \frac{f(A)}{f(A\cup B)} \cdot \frac{|A|}{|A|+|B|-1}$, whenever f and h are monotone submodular and g is a metric diversity function. This generalizes the bound from Example 1.5 (b).

We note that the sum of a monotone submodular function and a metric diversity function has been previously studied [4]. We discuss this in more detail in Section 3.

1.4 Additional related work

The notion of weak submodularity was introduced by Das and Kempe [11], where they showed that the standard greedy algorithm achieves a $(1 - e^{-\gamma})$-approximation for the monotone maximization problem subject to a cardinality constraint. They provided applications to the feature selection problem for linear regression and the dictionary selection problems. Khanna et al. [29] showed that faster (such as distributed and stochastic) versions of the greedy algorithm also retain provable theoretical guarantees for monotone weakly submodular maximization under a cardinality constraint. They discussed applications for the sparse linear regression problem and the support selection problem. Elenberg et al. [14] considered the above problem in the random order streaming setting, and provided applications to nonlinear sparse regression and interpretability of black-box neural network classifiers. Connections between weak submodularity and restricted strong convexity were shown by Elenberg et al. [15], and used for high-dimensional subset selection problems. The work of Chen et al. [9] goes beyond the cardinality constraint, and considers the monotone maximization problem subject to a matroid constraint. They provided an approximation ratio of $(1 + 1/\gamma)^{-2}$ for this problem, and discuss applications to video summarization, splice site detection, and black-box interpretation of images. Gatmiry and Gomez [20] showed that the standard deterministic greedy algorithm also enjoys provable guarantees for the above problem, though worse than the one provided by [9]. They provide applications to tree-structured

Gaussian graphical model estimation. The recent work of Harshaw et al. [26] considers the problem $\max\{f(S)-m(S) : |S| \leq k\}$, where f is non-negative monotone γ-weakly submodular and m is a non-negative modular function. Using the special structure of this type of objective, they circumvented the potential roadblocks of $f - m$ being negative or non-monotone, and provided a bifactor guarantee satisfying $f(S) - m(S) \geq (1 - e^{-\gamma})f(OPT) - m(OPT)$. In addition, they showed that this approximation ratio is tight in the value oracle model.

Non-monotone submodular maximization subject to a cardinality constraint has been studied extensively. The first constant factor approximation for this problem was given by Lee et al. [30]. Since then a large series of works [8, 16, 19, 23, 25, 38] have improved the approximation factor to the current best 0.385 ratio due to Buchbinder and Feldman [7]. Some of the latter works, however, use an approach that involves using a continuous relaxation of the objective function and then applying rounding methods to the fractional solution. While this approach has been extremely successful for proving strong theoretical guarantees, due to the run time they usually become impractical in real-world scenarios with large amounts of data. In our work we use a randomized greedy algorithm proposed by Buchbinder et al. [8], where it is shown that this algorithm produces a $1/e$-approximation (on expectation). On the inapproximability side, Gharan and Vondrak [23] show that it is impossible to achieve a 0.491 approximation for this problem in the value oracle model.

1.5 Our contributions

One main contribution of this work is showing that an easy-to-implement and fast randomized greedy algorithm (i.e., Algorithm 1) has provable theoretical guarantees for the problem $\max\{f(S) : |S| \leq k\}$ when the function $f : 2^E \to \mathbb{R}_+$ is non-monotone weakly submodular (as defined in Section 1.1). This is encapsulated in the following result. To the best of our knowledge, this is the first time that weakly submodular functions are considered in the non-monotone setting.

▶ **Theorem 1.9.** *There exists an efficient randomized greedy algorithm which has an approximation ratio (on expectation) of at least $\gamma(1 - 1/\gamma k)^{k-1}$ for the problem of maximizing a non-negative non-monotone γ-weakly submodular function subject to a cardinality constraint of size k. This approximation ratio is asymptotically $\gamma \cdot e^{-1/\gamma}$ as $k \to \infty$. For non-negative non-monotone γ-pseudo submodular functions, the approximation ratio is of at least $\gamma \cdot e^{-\gamma}$.*

We remark that when γ approaches to 1, our bounds recover the $1/e$ approximation factor given in [8] for the analysis of the same algorithm over submodular functions (i.e., the case when $\gamma = 1$).

A key ingredient for analyzing non-monotone objectives is to bound the term $\mathbb{E}[f(S_i \cup \text{OPT})]$ with respect to $f(\text{OPT})$. For submodular functions the work of [8] (see their Lemma 2.2 and Observation 1) bounds the above term by using the diminishing returns property, i.e., $f_A(e) \geq f_B(e)$ whenever $A \subseteq B$ and $e \notin B$. However, it is not clear how one could imitate such argument in the case of non-submodular functions. In particular, it is not obvious whether from the definition of weak submodularity, one could find a parameter $\beta > 0$ satisfying some approximate version $f_A(e) \geq \beta f_B(e)$ of diminishing returns. We circumvent this issue by analyzing the quantity $\mathbb{E}[f(S_i \cup \text{OPT})]$ directly with respect to the execution of the algorithm (see Lemma 2.3).

Another important piece of our work is to provide a more refined analysis that allows for the submodularity ratio to change throughout the execution of the algorithm. This is particularly useful since many classes of functions will usually satisfy this (see for instance Section 1.3). Our most general result (Theorem 2.4) assumes some local bounds for the submodularity ratio throughout the algorithm, and provides approximation guarantees based on these bounds. Its statement is somewhat less clean to express since it depends on the notation used in Algorithm 1 (which we introduce later in Section 2.1), so we defer its full presentation and discussion to Section 2.3. We next present some of its consequences, which lead to some of our main applications.

▶ **Theorem 1.10.** *Assume we run the randomized greedy algorithm described in Algorithm 1 on a function* $f : 2^E \to \mathbb{R}_+$ *with parameters* $\gamma_{A,B} \in [0,1]$ *for any pair of disjoint sets* $A, B \subseteq E$*. Moreover, assume there are values* $0 \leq \gamma_i \leq 1$ *for* $i \in \{0, 1, 2, \ldots, k-1\}$ *so that* $\sum_{e \in \mathrm{OPT}} f_{S_i}(e) \geq \min\{\gamma_i \cdot f_{S_i}(\mathrm{OPT}), f_{S_i}(\mathrm{OPT})\}$ *holds for any possible solution* S_i *of the algorithm after iteration* i*. Then the algorithm produces (on expectation):*

- *An approximation factor of at least* $1 - \exp(-\frac{1}{k}\sum_{i=0}^{k-1}\gamma_i)$ *if* f *is monotone.*
- *An approximation factor of at least* $\frac{1}{ek}\sum_{i=0}^{k-1}\gamma_i$ *if* f *is non-monotone.*

We remark that for monotone γ-weakly submodular objectives the above result retains the $(1 - e^{-\gamma})$-approximation given in [11]. This follows by noticing that for monotone functions we always have that $\min\{\gamma_i \cdot f_{S_i}(\mathrm{OPT}), f_{S_i}(\mathrm{OPT})\} = \gamma_i \cdot f_{S_i}(\mathrm{OPT})$ since $f_{S_i}(\mathrm{OPT}) \geq 0$ and $\gamma_i \in [0,1]$. One can then use the γ-weak submodularity of the function to set $\gamma_i = \gamma$ for all i.

Combining the above theorem with the results from Section 1.3 leads to interesting applications. We now highlight some of them, and defer a more detailed discussion to Section 3. The above theorem allows us to obtain provable guarantees for some of the non-monotone objectives discussed in Section 1.3. These include, for instance, the non-monotone functions from Example 1.8, which satisfy the property $\gamma_{A,B} \in [0,1]$.

Theorem 1.10 also leads to interesting results for monotone objectives. Applying it to Example 1.3 we get a 0.197-approximation for maximizing monotone proportionally submodular functions subject to a cardinality constraint. This improves over the current best 0.168-approximation from [6, 5]. Another set of applications is obtained via Example 1.5, which allows us to get several constant factor approximations for the product of set functions. For instance, for the product $f \cdot (g + h)$ where f, g are monotone submodular and h is a metric diversity function, our results lead to a 0.058-approximation. For the product $f \cdot g$ where f is monotone submodular and g is monotone proportionally submodular, we get a 0.046-approximation. We are not aware of previous work for these problems.

2 Approximation guarantees

In this section we present the main theoretical contribution of this work, which is to analyze the performance of a randomized greedy algorithm on non-monotone functions (see Section 2.3). We present the analysis for monotone objectives in Section 2.2. We next describe the randomized greedy algorithm that we use in this work.

2.1 Randomized greedy algorithm

In this section, we explain the randomized greedy algorithm introduced in the work of [8], where they study the problem of maximizing a non-monotone submodular function subject to a cardinality constraint. We note that this algorithm has also been used in [9] for the problem of maximizing a monotone weakly submodular function subject to a matroid constraint.

Given a set function $f : 2^E \to \mathbb{R}$ over a ground set E, we first add a set D of $2k$ dummy elements to the ground set. That is, for any set $A \subseteq E$ and $U \subseteq D$ the function satisfies $f_A(U) = 0$. Then, for each $1 \leq i \leq k$, we take a set of k elements that maximizes the sum

Algorithm 1 RandomizedGreedy(f, k).

```
Add a set $D$ of $2k$ dummy elements to $f$.
Initialize: $S_0 \leftarrow \emptyset$.
for $i = 1$ to $k$ do
    Let $M_i \subseteq (E \cup D) \setminus S_{i-1}$ be a subset of size $k$ maximizing $\sum_{e \in M_i} f_{S_{i-1}}(e)$. In case of ties
      between dummy elements and elements from $E$, always choose the latter.
    Let $e_i$ be a uniformly random element from $M_i$.
    $S_i \leftarrow S_{i-1} + e_i$.
return $S_k$.
```

of the marginal gains, where in case of ties we always give preference to elements from the original ground set E. Finally, we choose uniformly at random one of the k elements, and add it to the current solution. We summarize this procedure in Algorithm 1.

The algorithm is quite efficient as it makes $O(nk)$ queries to the value oracle. This is the same number of queries that the standard deterministic greedy algorithm makes. Moreover, adding $2k$ dummy elements to the original ground set guarantees the following.

▶ **Observation 2.1.** *At any iteration $1 \leq i \leq k$ of the RandomizedGreedy algorithm the following is satisfied:*

1. $|M_i| = k$.
2. $f_{S_{i-1}}(e_i) \geq 0$*, and hence* $f(S_i) \geq f(S_{i-1})$.
3. $\sum_{e \in M_i} f_{S_{i-1}}(e) \geq \sum_{e \in \mathrm{OPT}} f_{S_{i-1}}(e)$.

Proof. The first two statements are immediate from the fact that we add $2k$ dummy elements. To see the last statement, let $\bar{M}_i$ denote a set of size k containing $\mathrm{OPT} \setminus S_{i-1}$ and potentially some dummy elements (so that $|\bar{M}_i| = k$). Then, by definition of M_i we have

$$\sum_{e \in M_i} f_{S_{i-1}}(e) \geq \sum_{e \in \bar{M}_i} f_{S_{i-1}}(e) = \sum_{e \in \mathrm{OPT}} f_{S_{i-1}}(e). \qquad ◀$$

2.2 Analysis for monotone functions

In this section, we analyze the performance of the RandomizedGreedy algorithm on monotone functions. We note that we keep the term depending on the initial set S_0 in the approximation factor. The main reason for this is that while in many settings this will just be the empty set, in some applications one needs to start from a non-empty initial set S_0 to have provable guarantees for the parameter γ_i. (See for instance our applications for the product of set functions discussed in Section 3.) Then we would just run the algorithm for $k - |S_0|$ iterations.

▶ **Theorem 2.2.** *Let $f : 2^E \to \mathbb{R}_+$ be a monotone set function. Assume there are values $0 \leq \gamma_i \leq 1$ for $i \in \{0, 1, 2, \ldots, k-1\}$ so that*

$$\sum_{e \in \mathrm{OPT}} f_{S_i}(e) \geq \gamma_i \cdot f_{S_i}(\mathrm{OPT})$$

throughout the execution of the RandomizedGreedy algorithm, where S_i denotes the set of chosen elements after the ith iteration (i.e., $|S_i| = i$). Then at any iteration $1 \leq i \leq k$ the algorithm satisfies

$$\begin{aligned}\mathbb{E}[f(S_i)] &\geq \left(1-\prod_{j=0}^{i-1}\left(1-\frac{\gamma_j}{k}\right)\right)\cdot f(\mathrm{OPT})+\prod_{j=0}^{i-1}\left(1-\frac{\gamma_j}{k}\right)\cdot\mathbb{E}[f(S_0)]\\ &\geq \left(1-\exp\left(-\sum_{j=0}^{i-1}\frac{\gamma_j}{k}\right)\right)\cdot f(\mathrm{OPT})+\prod_{j=0}^{i-1}\left(1-\frac{\gamma_j}{k}\right)\cdot\mathbb{E}[f(S_0)].\end{aligned}$$

Proof. Fix $1 \leq i \leq k$ and a possible realization $S_1, S_2, \ldots, S_{i-1}$ of the algorithm of up to iteration $i-1$. Then (conditioned on this event) we have

$$\begin{aligned}\mathbb{E}[f_{S_{i-1}}(e_i)] &= \frac{1}{k}\sum_{e\in M_i} f_{S_{i-1}}(e) \geq \frac{1}{k}\sum_{e\in OPT} f_{S_{i-1}}(e) \geq \frac{\gamma_{i-1}}{k} f_{S_{i-1}}(OPT)\\ &= \frac{\gamma_{i-1}}{k}[f(S_{i-1}\cup OPT)-f(S_{i-1})] \geq \frac{\gamma_{i-1}}{k}[f(OPT)-f(S_{i-1})],\end{aligned}$$

where the first inequality follows from Observation 2.1, the second inequality from the theorem's assumption, and the last inequality from non-negativity of f. We then have

$$\mathbb{E}[f(S_i)] - f(S_{i-1}) \geq \frac{\gamma_{i-1}}{k}[f(OPT)-f(S_{i-1})],$$

and rearranging the terms we get

$$f(OPT) - \mathbb{E}[f(S_i)] \leq \left(1-\frac{\gamma_{i-1}}{k}\right)\Big[f(OPT)-f(S_{i-1})\Big].$$

By unfixing the realization $S_1, S_2, \ldots, S_{i-1}$ and taking expectations over all such possible realizations of the algorithm we get

$$\begin{aligned}f(OPT)-\mathbb{E}[f(S_i)] &\leq \left(1-\frac{\gamma_{i-1}}{k}\right)\Big[f(OPT)-\mathbb{E}[f(S_{i-1})]\Big]\\ &\leq \left(1-\frac{\gamma_{i-1}}{k}\right)\left(1-\frac{\gamma_{i-2}}{k}\right)\Big[f(OPT)-\mathbb{E}[f(S_{i-2})]\Big]\\ &\leq \cdots\\ &\leq \left(\prod_{j=0}^{i-1}\left(1-\frac{\gamma_j}{k}\right)\right)[f(OPT)-\mathbb{E}[f(S_0)]].\end{aligned}$$

Hence,

$$\begin{aligned}\mathbb{E}[f(S_i)] &\geq \left(1-\prod_{j=0}^{i-1}\left(1-\frac{\gamma_j}{k}\right)\right)\cdot f(\mathrm{OPT})+\prod_{j=0}^{i-1}\left(1-\frac{\gamma_j}{k}\right)\cdot\mathbb{E}[f(S_0)]\\ &\geq \left(1-\exp\left(-\sum_{j=0}^{i-1}\frac{\gamma_j}{k}\right)\right)\cdot f(\mathrm{OPT})+\prod_{j=0}^{i-1}\left(1-\frac{\gamma_j}{k}\right)\cdot\mathbb{E}[f(S_0)],\end{aligned}$$

where the last inequality uses that $1-x \leq e^{-x}$ for all $x \geq 0$. ◀

The above result now proves the first part of Theorem 1.10. This follows because by monotonicity of f we have $f_{S_i}(\mathrm{OPT}) \geq 0$, and hence $\min\{\gamma_i \cdot f_{S_i}(\mathrm{OPT}), f_{S_i}(\mathrm{OPT})\} = \gamma_i \cdot f_{S_i}(\mathrm{OPT})$.

2.3 Analysis for non-monotone functions

In this section we analyze the performance of the RandomizedGreedy algorithm on non-monotone functions. As mentioned in Section 1.5, a key ingredient for analyzing the non-monotone case is to bound the term $\mathbb{E}[f(S_i \cup \mathrm{OPT})]$ from below with respect to $f(\mathrm{OPT})$. For monotone objectives this is trivial, since by monotonicity we always have $f(S_i \cup \mathrm{OPT}) \geq f(\mathrm{OPT})$. The techniques used in [8] for analyzing RandomizedGreedy with respect to submodular functions make use of the diminishing returns property (see their Lemma 2.2 and Observation 1). However, it is not clear how to extend those techniques for non-monotone weakly submodular functions, since it is not obvious whether they satisfy some type of approximate diminishing returns property $f_A(e) \geq \beta f_B(e)$. Our next result circumvents this issue by analyzing the quantity $\mathbb{E}[f(S_i \cup \mathrm{OPT})]$ directly with respect to the execution of the algorithm.

▶ **Lemma 2.3.** *Let f be a non-negative set function. Assume there are numbers $0 \leq \bar{\alpha}_i \leq \bar{\beta}_i \leq k$ such that*

$$\sum_{u \in M_i} f_{S_{i-1} \cup \mathrm{OPT}}(u) \geq \min\{\bar{\alpha}_i \cdot f_{S_{i-1} \cup \mathrm{OPT}}(M_i), \bar{\beta}_i \cdot f_{S_{i-1} \cup \mathrm{OPT}}(M_i)\}$$

is satisfied for any choice of M_i and S_{i-1} throughout the execution of the RandomizedGreedy algorithm. Then at any iteration $1 \leq i \leq k$ the algorithm satisfies $\mathbb{E}[f(S_i \cup \mathrm{OPT})] \geq \prod_{j=1}^{i}(1 - \bar{\beta}_j/k) \cdot f(\mathrm{OPT})$.

Proof. Fix $1 \leq i \leq k$ and an event $S_1, S_2, \ldots, S_{i-1}$ of a possible path of the algorithm up to iteration $i-1$. Then (conditioned on this event) we have

$$\begin{aligned}
&\mathbb{E}[f(S_i \cup \mathrm{OPT})] \\
&= f(S_{i-1} \cup \mathrm{OPT}) + \mathbb{E}[f_{S_{i-1} \cup \mathrm{OPT}}(u_i)] = f(S_{i-1} \cup \mathrm{OPT}) + \frac{1}{k} \sum_{u \in M_i} f_{S_{i-1} \cup \mathrm{OPT}}(u) \\
&\geq f(S_{i-1} \cup \mathrm{OPT}) + \frac{1}{k} \min\{\bar{\alpha}_i \cdot f_{S_{i-1} \cup \mathrm{OPT}}(M_i), \bar{\beta}_i \cdot f_{S_{i-1} \cup \mathrm{OPT}}(M_i)\}.
\end{aligned}$$

We now consider separately the cases where the marginal gain $f_{S_{i-1} \cup OPT}(M_i)$ is either negative or non-negative. If it is non-negative, using that $0 \leq \bar{\alpha}_i \leq \bar{\beta}_i$ we get

$$\mathbb{E}[f(S_i \cup \mathrm{OPT})] \geq f(S_{i-1} \cup \mathrm{OPT}) + \frac{\bar{\alpha}_i}{k} f_{S_{i-1} \cup OPT}(M_i) \geq f(S_{i-1} \cup \mathrm{OPT}).$$

If on the other hand, the marginal gain is negative, then

$$\begin{aligned}
\mathbb{E}[f(S_i \cup \mathrm{OPT})] &\geq f(S_{i-1} \cup \mathrm{OPT}) + \frac{\bar{\beta}_i}{k} f_{S_{i-1} \cup \mathrm{OPT}}(M_i) \\
&= f(S_{i-1} \cup \mathrm{OPT}) + \frac{\bar{\beta}_i}{k}[f(S_{i-1} \cup \mathrm{OPT} \cup M_i) - f(S_{i-1} \cup \mathrm{OPT})] \\
&\geq f(S_{i-1} \cup \mathrm{OPT}) - \frac{\bar{\beta}_i}{k} f(S_{i-1} \cup \mathrm{OPT}) = \Big[1 - \frac{\bar{\beta}_i}{k}\Big] f(S_{i-1} \cup \mathrm{OPT}),
\end{aligned}$$

where the last inequality follows from non-negativity. Thus, for each possible fixed realization $S_1, S_2, \ldots, S_{i-1}$ of the algorithm up to iteration $i-1$ we have

$$\mathbb{E}[f(S_i \cup \mathrm{OPT})] \geq \Big[1 - \frac{\bar{\beta}_i}{k}\Big] f(S_{i-1} \cup \mathrm{OPT}).$$

By unconditioning on the event $S_1, S_2, \ldots, S_{i-1}$, and taking the expectation over all such possible events we get:

$$\begin{aligned}\mathbb{E}[f(S_i \cup \mathrm{OPT})] &\geq \left[1 - \frac{\bar{\beta}_i}{k}\right]\mathbb{E}[f(S_{i-1} \cup \mathrm{OPT})] \geq \left[1 - \frac{\bar{\beta}_i}{k}\right]\left[1 - \frac{\bar{\beta}_{i-1}}{k}\right]\mathbb{E}[f(S_{i-2} \cup \mathrm{OPT})] \\ &\geq \cdots \geq \prod_{j=1}^{i}\left[1 - \frac{\bar{\beta}_j}{k}\right]\mathbb{E}[f(S_0 \cup \mathrm{OPT})] = \prod_{j=1}^{i}\left[1 - \frac{\bar{\beta}_j}{k}\right] f(\mathrm{OPT}). \qquad \blacktriangleleft\end{aligned}$$

For submodular functions the above result becomes $\mathbb{E}[f(S_i \cup \mathrm{OPT})] \geq (1 - 1/k)^i \cdot f(\mathrm{OPT})$, since we can take $\bar{\alpha}_i = \bar{\beta}_i = 1$ for all i. We remark that this matches the bound provided in [8] for submodular functions (see their Observation 1). We now prove our main result.

▶ **Theorem 2.4.** *Let $f : 2^E \to \mathbb{R}_+$ be a set function. Assume there are values $0 \leq \bar{\alpha}_i \leq \bar{\beta}_i \leq k$ and $0 \leq \alpha_i \leq \beta_i \leq k$ such that*

$$\sum_{u \in M_i} f_{S_{i-1} \cup \mathrm{OPT}}(u) \geq \min\{\bar{\alpha}_i \cdot f_{S_{i-1} \cup \mathrm{OPT}}(M_i), \bar{\beta}_i \cdot f_{S_{i-1} \cup \mathrm{OPT}}(M_i)\}$$

and

$$\sum_{e \in OPT} f_{S_{i-1}}(e) \geq \min\{\alpha_{i-1} \cdot f_{S_{i-1}}(\mathrm{OPT}), \beta_{i-1} \cdot f_{S_{i-1}}(\mathrm{OPT})\}$$

is satisfied for any choice of M_i and S_{i-1} throughout the execution of the RandomizedGreedy *algorithm. Then at any iteration $1 \leq i \leq k$ the algorithm satisfies*

$$\mathbb{E}[f(S_i)] \geq \left(\prod_{j=1}^{i-1} \min\left\{1 - \frac{\bar{\beta}_j}{k}, 1 - \frac{\alpha_j}{k}\right\}\right) \cdot \left(\sum_{j=0}^{i-1} \frac{\alpha_j}{k}\right) \cdot f(\mathrm{OPT}).$$

Proof. Fix $1 \leq i \leq k$ and an event $S_1, S_2, \ldots, S_{i-1}$ of a possible realization of the algorithm up to iteration $i - 1$. Then (conditioned on this event) we have

$$\begin{aligned}\mathbb{E}[f_{S_{i-1}}(e_i)] &= \frac{1}{k}\sum_{e \in M_i} f_{S_{i-1}}(e) \geq \frac{1}{k}\sum_{e \in OPT} f_{S_{i-1}}(e) \\ &\geq \frac{1}{k}\min\{\alpha_{i-1} \cdot f_{S_{i-1}}(\mathrm{OPT}), \beta_{i-1} \cdot f_{S_{i-1}}(\mathrm{OPT})\},\end{aligned}$$

where the first inequality follows from Observation 2.1, and the second inequality from the theorem's assumption. We now consider separately the cases where the marginal gain $f_{S_{i-1}}(\mathrm{OPT})$ is either negative or non-negative. If it is non-negative, using that $0 \leq \alpha_i \leq \beta_i$ we get

$$\mathbb{E}[f_{S_{i-1}}(e_i)] \geq \frac{\alpha_{i-1}}{k} \cdot f_{S_{i-1}}(\mathrm{OPT}).$$

If it is negative we get

$$\mathbb{E}[f_{S_{i-1}}(e_i)] \geq 0 \geq \frac{\alpha_{i-1}}{k} \cdot f_{S_{i-1}}(\mathrm{OPT}),$$

where the first inequality follows from Observation 2.1. It then follows that for any fixed possible realization $S_1, S_2, \ldots, S_{i-1}$ of the algorithm of up to iteration $i - 1$ we have

$$\mathbb{E}[f_{S_{i-1}}(e_i)] \geq \frac{\alpha_{i-1}}{k} \cdot f_{S_{i-1}}(\mathrm{OPT}) = \frac{\alpha_{i-1}}{k}[f(S_{i-1} \cup \mathrm{OPT}) - f(S_{i-1})]. \tag{1}$$

We now unfix the realization $S_1, S_2, \ldots, S_{i-1}$ and take expectations over all such possible realizations of the algorithm.

$$\begin{aligned}\mathbb{E}[f(S_i)] &= \mathbb{E}[f(S_{i-1}) + f_{S_{i-1}}(e_i)] = \mathbb{E}[f(S_{i-1})] + \mathbb{E}[f_{S_{i-1}}(e_i)] \\ &\geq \mathbb{E}[f(S_{i-1})] + \frac{\alpha_{i-1}}{k}\mathbb{E}[f(S_{i-1} \cup \mathrm{OPT}) - f(S_{i-1})] \\ &= \left[1 - \frac{\alpha_{i-1}}{k}\right]\mathbb{E}[f(S_{i-1})] + \frac{\alpha_{i-1}}{k}\mathbb{E}[f(S_{i-1} \cup \mathrm{OPT})] \\ &\geq \left[1 - \frac{\alpha_{i-1}}{k}\right]\mathbb{E}[f(S_{i-1})] + \frac{\alpha_{i-1}}{k}\prod_{j=1}^{i-1}\left[1 - \frac{\bar{\beta}_j}{k}\right]f(\mathrm{OPT}), \end{aligned} \tag{2}$$

where the first inequality follows from Equation (1) and the last inequality follows from Lemma 2.3 (which we can use due to the theorem's assumptions).

We are now ready to prove the statement of the theorem using induction on the value of $1 \leq i \leq k$. The base case $i = 1$ claims that $\mathbb{E}[f(S_1)] \geq (\alpha_0/k) \cdot f(\mathrm{OPT})$. This follows from Equation (2) by setting $i = 1$ and using that $f(S_0) = f(\emptyset) \geq 0$.

Now let $1 < i \leq k$ be arbitrary, and assume that the claim is true for all values $1 \leq i' < i$; we show it is also true for i. Using Equation (2) and the induction hypothesis we get

$$\begin{aligned}\mathbb{E}[f(S_i)] &\geq \left[1 - \frac{\alpha_{i-1}}{k}\right]\mathbb{E}[f(S_{i-1})] + \frac{\alpha_{i-1}}{k}\prod_{j=1}^{i-1}\left[1 - \frac{\bar{\beta}_j}{k}\right]f(\mathrm{OPT}) \\ &\geq \left[\left[1 - \frac{\alpha_{i-1}}{k}\right]\left(\prod_{j=1}^{i-2}\min\left\{1 - \frac{\bar{\beta}_j}{k}, 1 - \frac{\alpha_j}{k}\right\}\right) \cdot \left(\sum_{j=0}^{i-2}\frac{\alpha_j}{k}\right) + \frac{\alpha_{i-1}}{k}\prod_{j=1}^{i-1}\left[1 - \frac{\bar{\beta}_j}{k}\right]\right]f(\mathrm{OPT}) \\ &\geq \left(\prod_{j=1}^{i-1}\min\left\{1 - \frac{\bar{\beta}_j}{k}, 1 - \frac{\alpha_j}{k}\right\}\right) \cdot \left(\left(\sum_{j=0}^{i-2}\frac{\alpha_j}{k}\right) + \frac{\alpha_{i-1}}{k}\right) \cdot f(\mathrm{OPT}) \\ &= \left(\prod_{j=1}^{i-1}\min\left\{1 - \frac{\bar{\beta}_j}{k}, 1 - \frac{\alpha_j}{k}\right\}\right) \cdot \left(\sum_{j=0}^{i-1}\frac{\alpha_j}{k}\right) \cdot f(\mathrm{OPT}). \end{aligned}$$ ◀

2.3.1 Proof of Theorems 1.9 and 1.10

The above result leads to several interesting consequences by choosing the values of the parameters $\alpha_i, \bar{\alpha}_i, \beta_i, \bar{\beta}_i$ appropriately. For instance, for non-monotone γ-weakly submodular functions we have $\alpha_i = \bar{\alpha}_i = \gamma$ and $\beta_i = \bar{\beta}_i = 1/\gamma$ for all i. Hence we immediately get an approximation of $\gamma(1 - 1/\gamma k)^{k-1}$, which is asymptotically $\gamma e^{-1/\gamma}$ as $k \to \infty$. In a similar fashion, for γ-pseudo submodular functions we can take $\alpha_i = \bar{\alpha}_i = \beta_i = \bar{\beta}_i = \gamma$ for all i, leading to an approximation factor of $\gamma(1 - \gamma/k)^{k-1} \geq \gamma e^{-\gamma}$. This now proves Theorem 1.9.

One can now also prove the second part of Theorem 1.10 as follows. First, if the function has a parameter that satisfies $0 \leq \gamma_{A,B} \leq 1$ (such as in Example 1.8), we immediately get that $\alpha_i, \bar{\alpha}_i, \beta_i, \bar{\beta}_i \leq \max_{A \cap B = \emptyset} \gamma_{A,B} \leq 1$. Hence $\prod_{j=1}^{k-1}\min\{1 - \alpha_j/k, 1 - \bar{\beta}_j/k\} \geq [1 - 1/k]^{k-1} \geq 1/e$. In addition, using the assumptions from Theorem 1.10 one can take $\alpha_i = \gamma_i$ and $\beta_i = 1$ in Theorem 2.4; leading to an approximation factor of $(1/ek) \cdot \sum_{j=0}^{k-1}\gamma_j$ as desired.

Theorem 2.4 becomes particularly useful to prove tighter guarantees for some of the examples discussed in Section 1.3, which have a parameter $\gamma_{A,B}$ that changes throughout the algorithm. We discuss this and applications for monotone objectives in the next section.

3 Applications

We now present some applications for our results. We discuss the monotone case first.

For monotone functions it is clear that the RandomizedGreedy algorithm always selects k elements from the original ground set E (i.e., it never chooses dummy elements). In particular, the current solution S_i at iteration i always has i elements from E, while $OPT \setminus S_i$ is a set containing at most k elements. We can use this, together with the results from Section 1.3, to compute a lower bound for a parameter $\gamma_i \geq 0$ that satisfies $\sum_{e \in \mathrm{OPT}} f_{S_i}(e) \geq \gamma_i \cdot f_{S_i}(\mathrm{OPT})$. For instance, one can take

$$\gamma_i = \min_{|A|=i,\ 1 \leq |B| \leq k,\ A \cap B = \emptyset} \gamma_{A,B}.$$

We then immediately get a provable approximation ratio of at least $1 - \exp(-\frac{1}{k}\sum_{i=0}^{k-1} \gamma_i)$ via Theorem 1.10 (or Theorem 2.2).

For monotone proportionally submodular functions, Example 1.3 gives a bound of $\gamma_{A,B} \geq \frac{3a(1+a)}{3a^2+3ab+b^2-1}$ where $a = |A|$ and $b = |B|$. Hence $\gamma_i \geq \frac{3i(1+i)}{3i^2+3ik+k^2-1}$ for $i \in \{0, 1, \ldots, k-1\}$. By plugging this into Theorem 1.10 we get an expression that does not seem to have a closed form, but that numerically converges from above to 0.197. This improves over the approximation factor of 0.168 given in [5] for the same problem (they give it as a 5.95-approximation since they express approximation factors as numbers greater than 1).

▶ **Theorem 3.1.** *There is an efficient* 0.197*-approximation for the problem of maximizing a non-negative monotone proportionally submodular function subject to a cardinality constraint.*

Our next application is for the product of monotone set functions. First, let us consider the case $f \cdot g$ where f is submodular and g is either submodular, metric diversity, or proportionally submodular. Example 1.5 provides explicit bounds for the parameter $\gamma_{A,B}$ of these product functions. We have $\gamma_{A,B} \geq (f(A)/f(A \cup B)) \cdot \gamma^g_{A,B}$ where the latter term denotes the parameter of the function g. Hence, we need to lower bound the term $f(A)/f(A \cup B)$. We can do this as follows. One can show that for submodular functions, if there is a set S_f satisfying $f(S_f) \geq \alpha \cdot \max_{|S| \leq k} f(S)$ then $f(A)/f(A \cup B) \geq \alpha/(1+\alpha)$ for any set $A \supseteq S_f$ and any set B of size at most k (see the full version of the paper for details). We can then take $S_0 = S_f$ as the initial set and run the RandomizedGreedy algorithm during $k - |S_0|$ iterations (to get a set of size k), with a guarantee that the parameter of the product function satisfies $\gamma_{A,B} \geq \alpha \cdot \gamma^g_{A,B}$. This leads to approximation guarantees of $1 - \exp(-\frac{1}{k}\sum_{i=k/2}^{k-1} \alpha \cdot \gamma^g_i)$, where γ^g_i denotes the parameter γ_i of the function g.

For submodular functions, we can run the standard greedy algorithm on f during $k/2$ iterations to find a set $S_f \subseteq E$ of size $k/2$ satisfying $f(S_f) \geq (1 - e^{-1/2}) \cdot \max_{|S| \leq k} f(S)$. Combining this with the fact that submodular functions have $\gamma_i \geq 1$, the sum of submodular and metric diversity has $\gamma_i \geq \frac{i}{i+k-1}$, and proportionally submodular functions have $\gamma_i \geq \frac{3i(1+i)}{3i^2+3ik+k^2-1}$ for $i \in \{0, 1, \ldots, k-1\}$, one can obtain the following approximation guarantees.

▶ **Theorem 3.2.** *Let f, g and h be non-negative and monotone. If f is submodular, then:*

- *There is an approximation (on expectation) of* 0.131 *for $f \cdot g$ when g is submodular.*
- *There is an approximation (on expectation) of* 0.058 *for $f \cdot (g + h)$ when g is a metric diversity function and h is submodular.*
- *There is an approximation (on expectation) of* 0.046 *for $f \cdot g$ when g is proportionally submodular.*

We are not aware of previous work for the product of set functions that we can compare our results to. However, when the functions are monotone, a natural baseline can be obtained by taking the set $S := S_f \cup S_g$ where S_f is obtained by running the greedy algorithm for $\max_{|S|\le k/2} f(S)$, and similarly S_g is obtained by running the greedy algorithm for $\max_{|S|\le k/2} g(S)$. Then if $f(S_f) \ge \alpha_f \cdot \max_{|S|\le k} f(S)$ and $g(S_g) \ge \alpha_g \cdot \max_{|S|\le k} g(S)$, we get that $(f \cdot g)(S_f \cup S_g) \ge \alpha_f \cdot \alpha_g \cdot fg(OPT)$. In the case of the above functions we get the following guarantees for α after running the greedy algorithm for $k/2$ iterations: for a submodular function we get $\alpha \ge 1 - e^{-1/2}$ via the standard greedy algorithm analysis, for the sum of submodular and metric diversity we get $\alpha \ge 1/8$ via the analysis from [4], and for proportionally submodular we get $\alpha \ge 0.05$ via the analysis using Example 1.3 and Theorem 1.10 (which improves over the previous analysis given in [5]). This leads to the following baselines (though there is room for optimizing the sizes of S_f and S_g): a 0.155-approximation for the product of two submodular functions, a 0.049-approximation for the product of a submodular function and the sum of submodular and metric diversity, and a 0.019-approximation for the product of a submodular function and a proportionally submodular function.

We note that our choice of cardinality $k/2$ for the initial set S_0 of the algorithm, and for the sets S_f and S_g used in the baselines, may not be optimal. For the sake of consistency and to keep the argument as clean as possible, we used the same cardinality for all of them.

By using a similar argument to the one from Theorem 3.2 one can also get constant factor approximations in the case where f is a metric diversity function. This follows since if $S_f \subseteq E$ satisfies $f(S_f) \ge \alpha \cdot \max_{|S|\le k} f(S)$, then $f(A)/f(A \cup B) \ge \alpha/(5+\alpha)$ for any set $A \supseteq S_f$ and any set B of size at most k (see the full version of the paper for details). The fact that this bound is worse than for submodular functions is expected, since f is supermodular and hence $f_A(e) \le f_B(e)$ whenever $A \subseteq B \subseteq E$ and $e \notin B$.

We now discuss the non-monotone case. While for monotone functions the algorithm always chooses k elements from the original ground set E (i.e., it never picks dummy elements), this may not be the case for non-monotone objectives. That is, for non-monotone objectives we have $S_k \subseteq E \cup D$. Hence, we cannot just directly plug the bounds for $\gamma_{A,B}$ from Section 1.3, since these depend on the number of elements from E that the current solution S_i has. Our next result gives a guarantee with respect to the number of elements from E that the algorithm picks.

▶ **Proposition 3.3.** *Let $f : 2^E \to \mathbb{R}_+$ be a set function with parameters $\gamma_{A,B} \in [0,1]$ satisfying that $\gamma_{A,B} \ge \gamma_{A',B}$ whenever $|A| \ge |A'|$. Then, if the RandomizedGreedy algorithm picks m elements from the original ground set E (i.e., not dummy elements), its output $S_k \subseteq E \cup D$ satisfies*

$$\mathbb{E}[f(S_k)] \ge \frac{1}{ke}\Big[(k-m)\bar{\gamma}_1 + \sum_{i=0}^{m-1} \bar{\gamma}_i\Big] \cdot f(\mathrm{OPT}),$$

where $\bar{\gamma}_i = \min\{\gamma_{A,B} : |A| = i,\ 1 \le |B| \le k,\ A \cap B = \emptyset\}$.

Proof. First note that since $\gamma_{A,B} \in [0,1]$ we also have that $\bar{\gamma}_i \in [0,1]$. We show that after iteration i, any realization of the algorithm (conditioned on the event that m elements from E are selected) satisfies $\sum_{e\in\mathrm{OPT}} f_{S_i}(e) \ge \min\{\gamma_i \cdot f_{S_i}(\mathrm{OPT}), f_{S_i}(\mathrm{OPT})\}$ where

$$\gamma_i = \begin{cases} \bar{\gamma}_0 & \text{if } i = 0, \\ \bar{\gamma}_1 & \text{if } 1 \le i \le k-m, \\ \bar{\gamma}_{i-k+m} & \text{if } k-m+1 \le i \le k-1. \end{cases}$$

Then the desired result follows from Theorem 1.10, since $\sum_{i=0}^{k-1} \gamma_i = \bar{\gamma}_0 + (k-m)\bar{\gamma}_1 + \sum_{i=1}^{m-1} \bar{\gamma}_i$.

First note that when $f_{S_i}(\mathrm{OPT}) < 0$ we have

$$\sum_{e\in\mathrm{OPT}} f_{S_i}(e) \geq \gamma_{S_i\cap E,\mathrm{OPT}\setminus S_i} \cdot f_{S_i}(\mathrm{OPT}) \geq f_{S_i}(\mathrm{OPT}) = \min\{\gamma_i \cdot f_{S_i}(\mathrm{OPT}), f_{S_i}(\mathrm{OPT})\},$$

where the first inequality follows from the definition of the parameter $\gamma_{A,B}$, the second inequality follows since $\gamma_{S_i\cap E,\mathrm{OPT}\setminus S_i} \in [0,1]$ and $f_{S_i}(\mathrm{OPT}) < 0$, and similarly the last equality follows since $\gamma_i \in [0,1]$ and $f_{S_i}(\mathrm{OPT}) < 0$. Hence the inequality $\sum_{e\in\mathrm{OPT}} f_{S_i}(e) \geq \min\{\gamma_i \cdot f_{S_i}(\mathrm{OPT}), f_{S_i}(\mathrm{OPT})\}$ is always satisfied when $f_{S_i}(\mathrm{OPT}) < 0$. In the case where $f_{S_i}(\mathrm{OPT}) \geq 0$ we get

$$\sum_{e\in\mathrm{OPT}} f_{S_i}(e) \geq \gamma_{S_i\cap E,\mathrm{OPT}\setminus S_i} \cdot f_{S_i}(\mathrm{OPT}) \geq \bar{\gamma}_{|S_i\cap E|} \cdot f_{S_i}(\mathrm{OPT}), \tag{3}$$

where the last inequality follows from the definition of $\bar{\gamma}_i$ and the fact that $|OPT \setminus S_i| \leq k$. We lower bound the term $|S_i \cap E|$ as follows.

We can always assume that in the first iteration the algorithm picks an element from the original ground set. This is because $f(\emptyset) = 0$ and $f(e) \geq 0$ for all $e \in E$ by non-negativity of f. Hence there is always a choice of k elements from the original ground set for the candidate set M_1. Now, since $\gamma_{A,B} \geq \gamma_{A',B}$ whenever $|A| \geq |A'|$, we have that the values $\bar{\gamma}_i$ are non-decreasing. It then follows that the worst scenario occurs when the algorithm picks the remaining $m-1$ non-dummy elements in the last $m-1$ iterations. In this case, we get that $|S_0 \cap E| = 0$, $|S_i \cap E| = 1$ for $1 \leq i \leq k-m$, and $|S_i \cap E| = i-k+m$ for $k-m+1 \leq i \leq k$. Combining this with Equation (3) leads to the desired result. ◀

The above result can be used to obtain bounds for some of the examples discussed in Section 1.3 that satisfy $0 \leq \gamma_{A,B} \leq 1$ and have non-decreasing values $\gamma_{A,B}$ as a function of $|A|$, such as those from Example 1.8. We discuss this next, where we state the approximation guarantees for the case the algorithm selects at least $k/2$ elements from E.

▶ **Corollary 3.4.** *Let $f : 2^E \to \mathbb{R}_+$ be a (non-monotone) set function, and assume the RandomizedGreedy algorithm picks at least $k/2$ elements from E. Then its output $S_k \subseteq E \cup D$ satisfies the following guarantees:*

(a) *If $f = g + h$ where g is monotone submodular and h is non-monotone proportionally submodular, then $\mathbb{E}[f(S_k)] \geq \frac{0.05}{e} \cdot f(\mathrm{OPT})$.*

(b) *If $f(S) := g(S) + |S| \cdot h(S)$ where g is monotone submodular and h is non-monotone submodular, then $\mathbb{E}[f(S_k)] \geq \frac{0.09}{e} \cdot f(\mathrm{OPT})$.*

Proof. From Example 1.8 we know that the function f from part (b) satisfies $\bar{\gamma}_i \geq (i+1)/(i+k)$, while the function from part (a) satisfies $\bar{\gamma}_i \geq (3i(i+1))/(3i^2+3ik+k^2-1)$. Plugging these values into Proposition 3.3 leads to the desired bounds. ◀

4 Conclusion

In this paper we introduced a natural generalization of weak submodularity for non-monotone functions. We showed that a randomized greedy algorithm has provable approximation guarantees for maximizing these functions subject to a cardinality constraint. We also provided a fine-grained analysis that allows the submodularity ratio to change throughout the algorithm. We discussed applications of our results for monotone and non-monotone functions.

It is open whether the $(\gamma \cdot e^{-1/\gamma})$-approximation is asymptotically tight for the maximization problem subject to a cardinality constraint. Another natural direction for future work is to consider the non-monotone maximization problem under more general constraints, such as matroids or knapsacks.

References

1 Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. Diversifying search results. In *Proceedings of the second ACM international conference on web search and data mining*, pages 5–14, 2009.

2 Andrew An Bian, Joachim M Buhmann, Andreas Krause, and Sebastian Tschiatschek. Guarantees for greedy maximization of non-submodular functions with applications. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 498–507, 2017.

3 Benjamin Birnbaum and Kenneth J Goldman. An improved analysis for a greedy remote-clique algorithm using factor-revealing lps. *Algorithmica*, 55(1):42–59, 2009.

4 Allan Borodin, Aadhar Jain, Hyun Chul Lee, and Yuli Ye. Max-sum diversification, monotone submodular functions, and dynamic updates. *ACM Transactions on Algorithms (TALG)*, 13(3):1–25, 2017.

5 Allan Borodin, Dai Le, and Yuli Ye. Proportionally submodular functions, 2015. Manuscript: `http://www.cs.toronto.edu/~bor/Papers/proportional-talg-submit.pdf`.

6 Allan Borodin, Dai Tri Man Le, and Yuli Ye. Weakly submodular functions. *arXiv preprint*, 2014. `arXiv:1401.6697`.

7 Niv Buchbinder and Moran Feldman. Constrained submodular maximization via a nonsymmetric technique. *Mathematics of Operations Research*, 44(3):988–1005, 2019.

8 Niv Buchbinder, Moran Feldman, Joseph Seffi Naor, and Roy Schwartz. Submodular maximization with cardinality constraints. In *Proceedings of the 25th annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1433–1452, 2014.

9 Lin Chen, Moran Feldman, and Amin Karbasi. Weakly submodular maximization beyond cardinality constraints: Does randomization help greedy? In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 803–812, 2018.

10 Wei Chen, Shang-Hua Teng, and Hanrui Zhang. Capturing complementarity in set functions by going beyond submodularity/subadditivity. In *10th Innovations in Theoretical Computer Science Conference (ITCS 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

11 Abhimanyu Das and David Kempe. Submodular meets spectral: greedy algorithms for subset selection, sparse approximation and dictionary selection. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pages 1057–1064, 2011.

12 Abhimanyu Das and David Kempe. Approximate submodularity and its applications: subset selection, sparse approximation and dictionary selection. *The Journal of Machine Learning Research*, 19(1):74–107, 2018.

13 Marina Drosou and Evaggelia Pitoura. Search result diversification. *ACM SIGMOD Record*, 39(1):41–47, 2010.

14 Ethan Elenberg, Alexandros G Dimakis, Moran Feldman, and Amin Karbasi. Streaming weak submodularity: Interpreting neural networks on the fly. In *Advances in Neural Information Processing Systems (NIPS)*, pages 4044–4054, 2017.

15 Ethan R Elenberg, Rajiv Khanna, Alexandros G Dimakis, Sahand Negahban, et al. Restricted strong convexity implies weak submodularity. *The Annals of Statistics*, 46(6B):3539–3568, 2018.

16 Alina Ene and Huy L Nguyen. Constrained submodular maximization: Beyond $1/e$. In *Proceedings of the IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 248–257, 2016.

17 Uriel Feige, Michal Feldman, Nicole Immorlica, Rani Izsak, Brendan Lucier, and Vasilis Syrgkanis. A unifying hierarchy of valuations with complements and substitutes. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 872–878, 2015.

18 Uriel Feige and Rani Izsak. Welfare maximization and the supermodular degree. In *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, pages 247–256, 2013.

19 Moran Feldman, Joseph Naor, and Roy Schwartz. A unified continuous greedy algorithm for submodular maximization. In *Proceedings of the IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 570–579, 2011.

20 Khashayar Gatmiry and Manuel Gomez-Rodriguez. Non-submodular function maximization subject to a matroid constraint, with applications. *arXiv preprint*, 2018. `arXiv:1811.07863`.

21 Mehrdad Ghadiri, Richard Santiago, and Bruce Shepherd. Beyond submodular maximization via one-sided smoothness. *arXiv preprint*, 2019. `arXiv:1904.09216`.

22 Mehrdad Ghadiri, Richard Santiago, and Bruce Shepherd. A parameterized family of meta-submodular functions. *arXiv preprint*, 2020. `arXiv:2006.13754`.

23 Shayan Oveis Gharan and Jan Vondrák. Submodular maximization by simulated annealing. In *Proceedings of the 22nd annual ACM-SIAM symposium on Discrete Algorithms (SODA)*, pages 1098–1116, 2011.

24 Jennifer Gillenwater, Alex Kulesza, and Ben Taskar. Near-optimal map inference for determinantal point processes. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2735–2743, 2012.

25 Anupam Gupta, Aaron Roth, Grant Schoenebeck, and Kunal Talwar. Constrained non-monotone submodular maximization: offline and secretary algorithms. In *Proceedings of the 6th international conference on Internet and network economics (WINE)*, pages 246–257, 2010.

26 Chris Harshaw, Moran Feldman, Justin Ward, and Amin Karbasi. Submodular maximization beyond non-negativity: Guarantees, fast algorithms, and applications. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 2634–2643, 2019.

27 Refael Hassin, Shlomi Rubinstein, and Arie Tamir. Approximation algorithms for maximum dispersion. *Operations research letters*, 21(3):133–137, 1997.

28 Thibaut Horel and Yaron Singer. Maximization of approximately submodular functions. In *Advances in Neural Information Processing Systems*, pages 3045–3053, 2016.

29 Rajiv Khanna, Ethan Elenberg, Alex Dimakis, Sahand Negahban, and Joydeep Ghosh. Scalable greedy feature selection via weak submodularity. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1560–1568, 2017.

30 Jon Lee, Vahab S Mirrokni, Viswanath Nagarajan, and Maxim Sviridenko. Maximizing nonmonotone submodular functions under matroid or knapsack constraints. *SIAM Journal on Discrete Mathematics*, 23(4):2053–2078, 2010.

31 Maxwell W Libbrecht, Jeffrey A Bilmes, and William Stafford Noble. Choosing non-redundant representative subsets of protein sequence data sets using submodular optimization. In *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics (BCB)*, pages 566–566, 2018.

32 Hui Lin and Jeff Bilmes. Multi-document summarization via budgeted maximization of submodular functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 912–920, 2010.

33 Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, pages 510–520, 2011.

34 Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, and Amin Karbasi. Fast constrained submodular maximization: Personalized data summarization. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pages 1358–1367, 2016.

35 Sekharipuram S Ravi, Daniel J Rosenkrantz, and Giri Kumar Tayi. Heuristic and special case algorithms for dispersion problems. *Operations Research*, 42(2):299–310, 1994.

36 Richard Santiago and Yuichi Yoshida. Weakly submodular function maximization using local submodularity ratio. *arXiv preprint*, 2020. `arXiv:2004.14650`.

37 Sebastian Tschiatschek, Rishabh K Iyer, Haochen Wei, and Jeff A Bilmes. Learning mixtures of submodular functions for image collection summarization. In *Advances in neural information processing systems (NIPS)*, pages 1413–1421, 2014.

38 Jan Vondrák. Symmetry and approximability of submodular maximization problems. *SIAM Journal on Computing*, 42(1):265–304, 2013.

39 Ghahramani Zoubin. Scaling the indian buffet process via submodular maximization. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, pages 1013–1021, 2013.