Differentially Private Decomposable Submodular Maximization

Anamay Chaturvedi, Huy Lê Nguyễn, Lydia Zakynthinou
Khoury College of Computer Sciences, Northeastern University, Boston, Massachusetts 02115
chaturvedi.a@northeastern.edu, hu.nguyen@northeastern.edu, zakynthinou.l@northeastern.edu

Abstract

We study the problem of differentially private constrained maximization of decomposable submodular functions. A submodular function is decomposable if it takes the form of a sum of submodular functions. The special case of maximizing a monotone, decomposable submodular function under cardinality constraints is known as the Combinatorial Public Projects (CPP) problem (Papadimitriou, Schapira, and Singer 2008). Previous work by Gupta et al. (2010) gave a differentially private algorithm for the CPP problem.

We extend this work by designing differentially private algorithms for both monotone and non-monotone decomposable submodular maximization under general matroid constraints, with competitive utility guarantees. We complement our theoretical bounds with experiments demonstrating improved empirical performance.

Introduction

A set function $f : 2^{\mathcal{N}} \to \mathbb{R}$ is submodular if it satisfies the following property of diminishing marginal returns: for all sets $S \subseteq T \subseteq \mathcal{N}$ and every element $u \in \mathcal{N} \setminus T$, $f(S \cup \{u\}) - f(S) \geq f(T \cup \{u\}) - f(T)$. Optimization problems involving the maximization of a submodular function arise naturally in many different applications, which span a wide range of fields such as combinatorial optimization (e.g. Max Cut, Max r-Cover, Facility Location, and Generalized Assignment problems), computer vision, operations research, and electrical networks (see Narayanan 1997; Fujishige 2005; Schrijver 2003). Furthermore, submodular functions are extensively used in economics (e.g. in the problem of welfare maximization in combinatorial auctions (Dobzinski and Schapira 2006; Feige 2006; Feige and Vondrák 2006; Vondrák 2008)). Recently, submodular maximization has found numerous applications to problems in machine learning (Kawahara et al. 2009), such as influence maximization in social networks (Kempe, Kleinberg, and Tardos 2003; Borgs et al. 2014; Borodin et al. 2017), result diversification in recommender systems (Puthiya Parambath, Usunier, and Grandvalet 2016), feature selection for classification (Krause and Guestrin 2005), dictionary selection (Krause and Cevher 2010), document and corpus summarization (Lin and Bilmes 2011; Kirchhoff and Bilmes 2014; Sipos et al. 2012), crowd teaching (Singla et al. 2014), and exemplar-based clustering (Dueck and Frey 2007; Gomes and Krause 2010).

In most applications, machine learning tools are applied to users' sensitive data, causing privacy concerns to become increasingly important. Differential Privacy (DP) (Dwork et al. 2006) has been widely-accepted as a robust mathematical guarantee that a model produced by a machine learning algorithm does not reveal sensitive, personal information contained in the training data. Notably, Mitrovic et al. (2017) gave differentially private algorithms for monotone and non-monotone submodular maximization under cardinality, matroid, and p-extendible system constraints. Gupta et al. (2010) studied a variety of combinatorial optimization problems under differential privacy and, in particular, gave a differentially private algorithm for the Combinatorial Public Projects (CPP) problem by Papadimitriou, Schapira, and Singer (2008). This is a special case of monotone submodular maximization under cardinality constraints, as the objective function f is *decomposable* (also known as *Sum-of-Submodular*).

Decomposable submodular functions encompass many examples of submodular functions studied in the context of machine learning as well as welfare maximization. In the latter, each agent has a valuation function over sets and the goal is to maximize the sum of the valuations of the agents, i.e., the "social welfare". In machine learning, data summarization, where the goal is to select a representative subset of elements of small size, falls into this setting and has numerous applications, including exemplar-based clustering, image summarization, recommender systems, active set selection, and corpus summarization. The line of work of Mirzasoleiman, Badanidiyuru, and Karbasi (2016); Mirzasoleiman et al. (2016); Mirzasoleiman, Zadimoghaddam, and Karbasi (2016) studies decomposable submbodular maximization under p-systems constraints in various data summarization settings and takes different approaches to user privacy.

Our Contributions

We focus on the problem of maximization of a decomposable submodular function under matroid constraints.

Definition 1. *A function $f : 2^{\mathcal{N}} \to \mathbb{R}$ is* λ-decomposable *if $f(S) = \sum_{I \in D} f_I(S)\ \forall S \subseteq \mathcal{N}$, for submodular functions $f_I : 2^{\mathcal{N}} \to [0, \lambda]$.*

Concretely, suppose there exists a set of agents D of size

m and a ground set of elements $\mathcal{N}$ of size n. We assume that each agent I has a submodular function $f_I : 2^{\mathcal{N}} \to [0, \lambda]$, and the goal is to find the subset S maximizing $f(S) = \sum_{I \in D} f_I(S)$ subject to a matroid constraint $\mathcal{M} = (\mathcal{N}, \mathcal{I})$ of rank r, under differential privacy. Let $f(\texttt{OPT})$ denote the value of the optimal non-private solution.

We provide two algorithms for the maximization of monotone and non-monotone decomposable submodular functions under matroid constraints, with utility guarantees close to the non-private optimal. Our contributions, denoted by $(\star)$, are summarized in Table 1. Our solution exhibits a tradeoff between the multiplicative and the additive factor via an arbitrarily small constant η, which depends on the chosen number of rounds of the algorithm.

Our results extend the results of Gupta et al. (2010) from cardinality to matroid constraints, as well as to non-monotone functions. The multiplicative factor of our utility guarantee for the monotone case is arbitrarily close to the optimal for the non-private version of the problem and the additive factor is optimal for any ε-differentially private algorithm with approximation factor 1 (see lower bound by Gupta et al. 2010, Thm. 8.5). In particular, delving into the proof of the lower bound in (Gupta et al. 2010), we can see that a stronger statement holds: no ε-differentially private algorithm can achieve additive factor less than r/ε without incurring a polynomial approximation factor (in the order of $(n/r)^{1/5}$). Therefore, if we want a constant approximation factor, an additive error of r/ε is necessary. In general, for combinatorial optimization problems such as this, the cost of privacy manifests itself in the additive error (see lower bounds in (Gupta et al. 2010)). Thus, minimizing the additive error and reaching this fundamental limit, which in our case can be achieved due to the decomposability assumption, is our foremost consideration.

In comparison, the general case of submodular function maximization assumes functions of bounded *sensitivity*, that is, $\max_S \max_{A,B} |f_A(S) - f_B(S)| \leq \lambda$ for A, B sets of agents that differ in at most one agent. The decomposability assumption allows us to improve on the utility guarantees of the general case of the maximization of submodular λ-sensitive functions, studied by Mitrovic et al. (2017), in our multiplicative and additive factor. Mitrovic et al. (2017) also note that using their general greedy algorithm for monotone submodular maximization under matroid constraints with the analysis of Gupta et al. (2010) yields a result for decomposable functions with improved additive error.

In proving our results, we also fix a lemma that is essential in the privacy analysis of the CPP problem of Gupta et al. (2010) and, in turn, in the result for decomposable monotone submodular maximization under matroid constraints of (Mitrovic et al. 2017) mentioned above, which allows for the improved additive error.

We complement our theoretical bounds with experiments on a dataset of Uber pickups in Section . We show that our algorithms perform better than the more general algorithms of (Mitrovic et al. 2017) for monotone submodular maximization, and are close to the non-private greedy algorithm.

Related Work

Submodular maximization There is a vast literature on submodular maximization (see (Buchbinder and Feldman 2018) for a survey), for which the greedy technique has been a dominant approach. Nemhauser, Wolsey, and Fisher (1978) introduced the basic greedy algorithm for the maximization of a monotone submodular function that iteratively builds a solution by choosing the item with the largest marginal gain with respect to the set of previously selected items. This algorithm achieves a $\left(1 - \frac{1}{e}\right)$-approximation for a cardinality constraint (which is optimal, see (Raz and Safra 1997)) and a $1/2$-approximation for a matroid constraint.

Calinescu et al. (2011) developed a framework based on continuous optimization and rounding that led to an optimal $\left(1 - \frac{1}{e}\right)$-approximation for the problem. The approach is to turn the discrete optimization problem of maximizing a submodular function f subject to a matroid constraint into a continuous problem of maximizing the multilinear extension F (a continuous extension of f) subject to the matroid polytope (a convex polytope whose vertices are the feasible integral solutions). The continuous problem can be solved within a $\left(1 - \frac{1}{e}\right)$ factor with a *Continuous Greedy* algorithm (Vondrák 2008). In each round $t \in [T]$, this algorithm estimates the marginal gains of each element u with respect to the current fractional solution $y^{(t)}$: $F(y^{(t)} \vee \mathbf{1}_u) - F(y^{(t)}) = \mathbb{E}\big[f(R(y^{(t)}) \cup \{u\}) - f(R(y^{(t)}))\big]$, where $R(y)$ is a random set which contains v with probability y_v. The algorithm finds an independent set of the matroid, $B^{(t)}$, maximizing the sum of the estimated marginal gains of the items, and updates the current fractional solution by taking a small step $\eta = 1/T$ in the direction of the selected set: $y^{(t+1)} = y^{(t)} + \eta \mathbf{1}_{B^{(t)}}$. The final solution $y^{(T)}$ is then rounded without loss (Chekuri, Vondrak, and Zenklusen 2010).

The *Measured Continuous Greedy* algorithm of Feldman, Naor, and Schwartz (2011) is a variant of the continuous greedy, which increases the coordinates of its fractional solution more slowly, and achieves a $\frac{1}{e}$-approximation for the general case of non-monotone submodular functions. This is not the optimal approximation factor (Buchbinder and Feldman 2019; Ene and Nguyen 2016), but the structure of the algorithm is favorable for its private adaptation.

Private submodular maximization A randomized algorithm $\mathcal{A} : D \to \mathcal{R}$ is (ε, δ)*-differentially private* (Dwork et al. 2006) if for all neighboring sets D, D' (i.e., that differ in at most one element) and any measurable output set R, $\Pr[\mathcal{A}(D) \in R] \leq e^{\varepsilon} \Pr[\mathcal{A}(D') \in R] + \delta$. The private algorithms of (Mitrovic et al. 2017) and (Gupta et al. 2010) are based on the discrete greedy algorithm, where the greedy step of selecting an item in each round is implemented via the Exponential Mechanism (McSherry and Talwar 2007), which guarantees that the selected item is almost as good as the marginal gain maximizer. By the advanced composition property of DP (Dwork, Rothblum, and Vadhan 2010), r consecutive runs of an $(\varepsilon, 0)$-DP algorithm lead to a cumulative privacy guarantee of the order of $\sqrt{r}\varepsilon$. Remarkably, for the case of decomposable monotone submodular functions, Gupta et al. (2010) show that the privacy guarantee

	r-Cardinality	Matroid (rank r)
λ-decomposable		
Monotone	$\left(1-\frac{1}{e}\right)f(\texttt{OPT})-\frac{r\lambda}{\varepsilon}\log n$ (GLMRT10)	$\left(1-\frac{1}{e}-\eta\right)f(\texttt{OPT})-\frac{r\lambda}{\eta\varepsilon}\log n\ (\star)$ $\frac{1}{2}f(\texttt{OPT})-\frac{r\lambda}{\varepsilon}\log n$ (MBKK17)
Non-monotone	$\left(\frac{1}{e}-\eta\right)f(\texttt{OPT})-\frac{r\lambda}{\eta\varepsilon}\log n\ (\star)$	$\left(\frac{1}{e}-\eta\right)f(\texttt{OPT})-\frac{r\lambda}{\eta\varepsilon}\log n\ (\star)$
λ-sensitive		
Monotone	$\left(1-\frac{1}{e}\right)f(\texttt{OPT})-\frac{r^{3/2}\lambda}{\varepsilon}\log n$ (MBKK17)	$\frac{1}{2}f(\texttt{OPT})-\frac{r^{3/2}\lambda}{\varepsilon}\log n$ (MBKK17) $\left(1-\frac{1}{e}\right)f(\texttt{OPT})-\frac{nr^{7}\lambda}{\varepsilon^{3}}\log n$ (RY20)
Non-monotone	$\frac{1}{e}\left(1-\frac{1}{e}\right)f(\texttt{OPT})-\frac{r^{3/2}\lambda}{\varepsilon}\log n$ (MBKK17)	–
General, non-private		
Monotone	$\left(1-\frac{1}{e}\right)f(\texttt{OPT})$ (NWF78)	$\left(1-\frac{1}{e}\right)f(\texttt{OPT})$ (CCPRV11; V08)
Non-monotone	$0.385f(\texttt{OPT})$ (BF19)	$0.385f(\texttt{OPT})$ (BF19)

Table 1: Expected utility guarantees of submodular maximization algorithms. The bottom section refers to submodular maximization without privacy constraints, whereas the top two refer to DP submodular maximization. All results omit any $\log\frac{1}{\delta}$ and constant factors.

of r rounds is, up to constant factors, the same as that of a single run of the exponential mechanism, which allows for the improved additive error in this case. It is the main idea of this proof by Gupta et al. (2010) that we extend to the case of matroid constraints and non-monotone funtions.

More recently, Rafiey and Yoshida (2020) proposed a differentially private submodular maximization algorithm for the general case of λ-sensitive functions achieving a multiplicative approximation factor of $(1-1/e)$, which is arbitrarily close to our approximation for decomposable submodular functions. However, the additive error, which is precisely the error we aim to minimize, is in the order of $\frac{nr^7\log n}{\varepsilon^3}$, which is large in comparison to our $\frac{r\lambda}{\varepsilon}$ for the case of decomposable submodular functions, especially in the high rank regime.

We finally note that, in principle, differentially private submodular optimization is related to submodular maximization in the presence of noise (Hassidim and Singer 2017). However, the structure of the noise is of a multiplicative nature, so it is not clear how these algorithms could be applicable.

Techniques

Our algorithms for the monotone and non-monotone problems are a private adaptation of the Continuous and Measured Continuous Greedy algorithms, respectively. They both use the Exponential Mechanism to greedily find an independent set $B^{(t,r)}$ in each round t and update with this set the current fractional solution. Our privacy analysis is based on the technique of (Gupta et al. 2010).

The main idea of this technique is the following. Let A, B be two sets of agents which differ in the individual I, as $A = B \cup \{I\}$. The privacy loss of the algorithm is bounded by the sum over the rounds of the expected marginal gains of each item with respect to the valuation function of agent I, where the expected value is calculated over a distribution that depends on the valuation functions of the rest of the agents B. More formally, the privacy loss is bounded by $\sum_{i=1}^{r}\mathbb{E}_u[f_I(S_{i-1}\cup\{u\})-f_I(S_{i-1})]$. By a key lemma, whose proof we fix and state in Section , this is bounded by a function of the sum of the *realized* marginal gains $\sum_{i=1}^{r}[f_I(S_{i-1}\cup\{u_i\})-f_I(S_{i-1})] = \sum_{i=1}^{r}[f_I(S_i)-f_I(S_{i-1})] = f_I(S)-f_I(\emptyset)$, which in turn is bounded by λ. Note that it is important in this argument that the sum telescopes to the total utility gain of f_I.

We now explain the main challenges in its application to our continuous algorithms. Recall that we use the continuous greedy algorithm to achieve the optimal multiplicative guarantee for the monotone case, which means that instead of calculating the marginal gain of u in each round with respect to f, we have to estimate $F(y^{(t,i-1)}\vee\mathbf{1}_u)-F(y^{(t,i-1)})$. Since the random sets used to estimate these marginal gains are drawn independently in each round, the final sum of the estimated marginal gains with respect to f_I is not a telescoping sum. If instead we use concentration to argue that the final sum is close to the true marginal gains with respect to F_I, the final telescoping sum would be in the order of $T\lambda$.

To overcome both these problems, we take two steps. First, we choose the smoother marginal gains $F(y^{(t,i-1)}+\eta\mathbf{1}_u)-F(y^{(t,i-1)})$, so that the realized marginal gain is $F_I(y^{(t,i)})-F_I(y^{(t,i-1)})$, which, by concentration, leads to the telescoping sum $F_I(y^{(T,r)})-F_I(y^{(1,0)})$ up to the sampling error term. However, this is not enough as the latter would be on the order of m, the number of agents. In regimes of interest, this is large enough that we would want to avoid

Algorithm 1 Private Continuous Greedy

Input: Utility parameters $\eta, \gamma \in (0,1]$, privacy parameters $\varepsilon, \delta \in (0,1]$, and set of agents D.
Let $T \leftarrow \lceil \frac{1}{\eta} \rceil$ and $\varepsilon_0 \leftarrow 2\log\left(1 + \frac{\varepsilon}{4+\log(1/\delta)}\right)$.
Draw $s = 6r^2T^4\log(n/\gamma)$ independent random vectors such that $r^j \leftarrow \mathcal{U}^n$ for all $j \in [s]$.
$y^{(1,0)} = \mathbf{1}_\emptyset$.
for $t = 1, \ldots, T$ **do**
  $B^{(t,0)} = \emptyset$.
  for $i = 1, \ldots, r$ **do**
    Let $\mathcal{N}^{(t,i)} = \{u \in \mathcal{N} \setminus B^{(t,i-1)} : B^{(t,i-1)} \cup \{u\} \in \mathcal{I}\}$.
    if $\mathcal{N}^{(t,i)} = \emptyset$ **then** let $y^{(t,r)} = y^{(t,i-1)}$ and break the loop.
    Define $\tilde{w}_D^{(t,i)}(u) = G(y^{(t,i-1)} + \eta\mathbf{1}_u) - G(y^{(t,i-1)})$ for all $u \in \mathcal{N}^{(t,i)}$.
    Let $u^{(t,i)} \leftarrow \mathcal{O}_{\varepsilon_0}(\tilde{w}_D^{(t,i)})$.
    Let $y^{(t,i)} = y^{(t,i-1)} + \eta\mathbf{1}_{u^{(t,i)}}$.
    Let $B^{(t,i)} \leftarrow B^{(t,i-1)} \cup \{u^{(t,i)}\}$.
  $y^{(t+1,0)} = y^{(t,r)}$.
return SWAP-ROUNDING$(y^{(T,r)}, \mathcal{I})$.

any dependence on m in the utility or sample complexity.

Second, we introduce a function $G : [0,1]^{\mathcal{N}} \to \mathbb{R}$. This function is not itself submodular but serves as a proxy for F. To construct G, we draw uniform vectors $r^j \in [0,1]^{\mathcal{N}}$ in the beginning of the algorithm, and define $G(x)$ to be the average over samples $f(\{u \in \mathcal{N} : r_u^j < x_u\})$. Therefore, $G_I(x)$ is always bounded by λ and the sum of estimated marginal gains of agent I telescopes to $G_I(y^{(T,r)}) - G_I(y^{(1,0)}) \le \lambda$. It follows that G's sampling error only affects utility.

Further applying this technique to the non-monotone case requires a bound on the sum of the *absolute* realizable marginal gains of the function on non-decreasing inputs. This bound does not hold for all non-monotone functions, but it is true for submodular functions, as we prove in Section .

Monotone

We denote by $w_D^{(t,i)}(u) = F(y^{(t,i-1)} + \eta\mathbf{1}_u) - F(y^{(t,i-1)})$ the true marginal gain of an element u in round (t,i). We let $G(x)$ be the estimate of $F(x)$ for any point $x \in [0,1]^n$. To compute G, we generate s uniformly random vectors $r^j \leftarrow \mathcal{U}^n$ for $j \in [s]$ in the beginning of the algorithm, where $\mathcal{U}$ denotes the uniform distribution over $[0,1]$, and set

$$G(x) = \frac{1}{s}\sum_{j=1}^{s} f(\{u \in \mathcal{N} : r_u^j < x_u\}).$$

We denote by $\mathcal{O}_{\varepsilon_0}(h)$ the Exponential Mechanism (McSherry and Talwar 2007) with privacy parameter ε_0 and score function h, formally defined in the supplementary. For ease of notation, we consider 1-decomposable functions. This is equivalent to adding a pre-processing step to scale the function by $1/\lambda$, which only affects the additive error.

Theorem 1. *Let $f : 2^{\mathcal{N}} \to \mathbb{R}_+$, where $|\mathcal{N}| = n$, be a monotone, 1-decomposable, submodular function and $\mathcal{M} = (\mathcal{N}, \mathcal{I})$ a matroid of rank r. Algorithm 1 with parameters η and γ is (ε, δ)-differentially private and returns a set $S \in \mathcal{I}$ such that, with probability $1-\gamma$,*

$$\mathbb{E}[f(S)] \ge (1 - \frac{1}{e} - O(\eta))f(\texttt{OPT}) - O\left(\frac{r}{\eta\varepsilon}\log\frac{nr}{\eta\gamma} \cdot \log\frac{1}{\delta}\right).$$

Algorithm 1 makes $O\left(\frac{nr^3}{\eta^5}\log\frac{n}{\gamma}\right)$ oracle calls.

We prove the theorem by combining the utility and privacy guarantees of our algorithm, as stated in Theorem 2 and Theorem 3, respectively. We remark that Theorem 2 lower bounds the utility of the fractional solution $F(y^{(T,r)})$. Since $y^{(T,r)} = \sum_{t=1}^{T}\eta\mathbf{1}_{B^{(t,r)}}$, where $B^{(t,r)} \in \mathcal{I}$ for all $t \in [T]$, it follows that $y^{(T,r)} \in \mathcal{P}(\mathcal{M})$ (the convex and down-closed polytope of $\mathcal{M}$). The results of (Chekuri, Vondrak, and Zenklusen 2010) can be applied to yield the final guarantees of the integral solution returned by the swap-rounding process.

For the utility analysis, we first show that G is a good proxy for F by bounding the sampling error.

Lemma 1. *With probability at least $1 - 2\gamma$, for any sequence of points picked by the algorithm $\left\{\{u^{(t,i)}\}_{i=1}^{r}\right\}_{t=1}^{T}$ and any $u \in \mathcal{N}$, it holds that*

$$(1-\eta)w_D^{(t,i)}(u) - \frac{\eta f(\texttt{OPT})}{rT} \le \tilde{w}_D^{(t,i)}(u) \text{ and}$$

$$\tilde{w}_D^{(t,i)}(u) \le (1+\eta)w_D^{(t,i)}(u) + \frac{\eta f(\texttt{OPT})}{rT}.$$

The utility proof follows the steps of that of the Continuous Greedy algorithm, yet accounting for the discretization, the error of the Exponential Mechanism, and the sampling.

Theorem 2. *With probability at least $1 - 3\gamma$,*

$$F(y^{(T+1,0)}) \ge (1 - 1/e - O(\eta))f(\texttt{OPT}) - \frac{8r}{\eta\varepsilon_0}\log\frac{nr}{\eta\gamma}.$$

Proof Sketch. With probability $1-2\gamma$, the bounds of Lemma 1 hold. We condition on this event.

$$F(y^{(t+1,0)}) - F(y^{(t,0)}) = \sum_{i=1}^{r} w_D^{(t,i)}(u^{(t,i)})$$

$$\ge \frac{1}{1+\eta}\sum_{i=1}^{r}\tilde{w}_D^{(t,i)}(u^{(t,i)}) - \frac{\eta f(\texttt{OPT})}{T}.$$

We assume wlog that $|B^{(t,r)}| = r$ and that there exists a mapping of $u^{(t,i)}$ to $o^{(t,i)}$, where $\texttt{OPT} = \{o^{(t,1)}, \ldots, o^{(t,r)}\}$. Since $o^{(t,i)}$ is a feasible option in the i-th round, by the guarantees of the Exponential Mechanism, with probability $1-\gamma$ we have that for all rounds $(t,i) \in [T] \times [r]$,

$$\tilde{w}_D^{(t,i)}(u^{(t,i)}) \ge \tilde{w}_D^{(t,i)}(o^{(t,i)}) - \frac{2}{\varepsilon_0}\log\frac{nrT}{\gamma}.$$

We condition on this event for the rest of the proof. Thus, by Lemma 1, with probability $1-3\gamma$,

$$F(y^{(t+1,0)}) - F(y^{(t,0)})$$
$$\ge \frac{1-\eta}{1+\eta}\sum_{i=1}^{r} w_D^{(t,i)}(o^{(t,i)}) - \frac{2\eta f(\texttt{OPT})}{T} - \frac{2r}{\varepsilon_0}\log\frac{nrT}{\gamma}. \tag{1}$$

Claim 1. *For mototone f, for all $t \in [T]$, $\sum_{i=1}^{r} w_D^{(t,i)}(o^{(t,i)}) \geq \eta[f(\texttt{OPT}) - F(y^{(t,r)})]$.*

Substituting this bound in inequality (1), we get that

$$F(y^{(t+1,0)}) - F(y^{(t,0)}) \geq \eta[(1-2\eta)f(\texttt{OPT}) - F(y^{(t+1,0)})] - \frac{2\eta f(\texttt{OPT})}{T} - \frac{2r}{\varepsilon_0}\log\frac{nrT}{\gamma}.$$

By rearranging and induction, we have that with probability at least $1-3\gamma$,

$$F(y^{(T+1,0)}) \geq \left(1 - \frac{1}{(1+\eta)^T}\right)(1-2\eta)f(\texttt{OPT}) - T\left(\frac{2\eta f(\texttt{OPT})}{T} + \frac{2r}{\varepsilon_0}\log\frac{nrT}{\gamma}\right)$$

$$\Rightarrow F(y^{(T+1,0)}) \geq (1-1/e-O(\eta))f(\texttt{OPT}) - \frac{8r}{\eta\varepsilon_0}\log\frac{nr}{\eta\gamma}$$

$$(\text{since } T = \lceil \tfrac{1}{\eta} \rceil \leq \tfrac{2}{\eta})$$

This concludes the sketch of the proof. □

For the privacy analysis, we need the next concentration bound (Claim 2). A stronger version of this bound with respect to constant factors also appears in (Gupta et al. 2010), but its proof is not entirely correct. We briefly explain the mistake in (Gupta et al. 2010) in the supplementary.

Claim 2. *Consider an n-round probabilistic process. In each round $i \in [n]$, an adversary chooses a distribution $\mathcal{D}_i$ over $[0,1]$ and a sample R_i is drawn from this distribution. Let $Z_1 = 1$ and $Z_{i+1} = Z_i - R_i Z_i$. We define the random variable $Y_j = \sum_{i=j}^{n} Z_i \mathbb{E}[R_i]$. Then for any $j \in [n]$,*

$$\mathbb{P}[Y_j \geq qZ_j] \leq \exp(3-q).$$

Proof. The proof is by reverse induction on j. For $j = n$, $Y_n = \mathbb{E}[R_n]Z_n \leq Z_n$ since $\mathbb{E}[R_n] \in [0,1]$. It follows that $\mathbb{P}[Y_n \geq qZ_n] = 0$ for $q > 1$, so the claim is trivially true for $j = n$ and for any q.

For the inductive step, suppose $\mathbb{P}[Y_{j+1} \geq qZ_{j+1}] \leq \exp(3-q)$. We will prove that $\mathbb{P}[Y_j \geq qZ_j] \leq \exp(3-q)$. For $q \leq 3$ the RHS is at least 1, so the claim is trivially true. Let us denote $\mu_j = \mathbb{E}[R_j]$.

$$\mathbb{P}[Y_j \geq qZ_j] = \mathbb{E}\left[\mathbb{P}\left[Y_{j+1} \geq \frac{q-\mu_j}{1-R_j}\cdot Z_{j+1}\right]\right] \leq \mathbb{E}\left[\exp\left(3 - \frac{q-\mu_j}{1-R_j}\right)\right],$$

by the inductive hypothesis. It suffices to prove that $\mathbb{E}\left[\exp\left(3 - \frac{q-\mu_j}{1-R_j}\right)\right] \leq \exp(3-q)$ for $q > 3$. This is equivalent to $\mathbb{E}\left[\exp\left(\frac{\mu_j - qR_j}{1-R_j}\right)\right] \leq 1$, for $q > 3$.

Let us denote $f(R_j) = \exp\left(\frac{\mu_j - qR_j}{1-R_j}\right)$. Calculations show that $f''(R_j) > 0$ so f is convex for $q > 3$ and $R_j, \mu_j \in [0,1]$. Therefore, $\mathbb{E}[f(R_j)] \leq \mathbb{E}[(1-R_j)f(0) + R_j f(1)] = (1-\mu_j)f(0) + \mu_j f(1) = (1-\mu_j)\exp(\mu_j) + 0 \leq 1$, concluding the proof of the inductive step and the claim. □

Theorem 3. *Algorithm 1 is $\big((e^{\varepsilon_0/2}-1)(4+\log\frac{1}{\delta}), \delta\big)$-differentially private.*

The privacy analysis follows (Gupta et al. 2010) using Claim 2. For A, B sets of agents such that $A \triangle B = \{I\}$, we bound the ratio of the probabilities that the sequence of chosen elements over the rounds be U, under inputs A and B, which suffices by the post-processing property of DP (Dwork et al. 2006). Crucially, our setting of G allows $\sum_{t=1}^{T}\sum_{i=1}^{r}[G_I(y^{(t,i-1)} + \eta\mathbf{1}_{u^{(t,i)}}) - G_I(y^{(t,i-1)})] = G_I(y^{(T,r)}) - G_I(y^{(1,0)}) \leq 1$.

Proof Sketch. Let A and B be two sets of agents such that $A \triangle B = \{I\}$. Suppose that, instead of the output set, we reveal the sequence in which we pick the elements of our algorithm, denoted by $U = (u^{(1,1)}, u^{(1,2)}, \dots, u^{(T,r)})$. We then want to bound the ratio of the probabilities that the output sequence be U under input A and B. By the post-processing property of DP (Dwork et al. 2006) this suffices to achieve the same privacy parameters over the output of the algorithm, SWAP-ROUNDING$(y^{(T,r)}, \mathcal{I})$.

$$\frac{\mathbb{P}[\mathbf{M}(A) = U]}{\mathbb{P}[\mathbf{M}(B) = U]} = \left(\prod_{t=1}^{T}\prod_{i=1}^{r}\frac{\exp(\frac{\varepsilon_0}{2}\tilde{w}_A^{(t,i)}(u^{(t,i)}))}{\exp(\frac{\varepsilon_0}{2}\tilde{w}_B^{(t,i)}(u^{(t,i)}))}\right) \cdot \left(\prod_{t=1}^{T}\prod_{i=1}^{r}\frac{\sum_{u\in\mathcal{N}^{(t,i)}}\exp(\frac{\varepsilon_0}{2}\tilde{w}_B^{(t,i)}(u))}{\sum_{u\in\mathcal{N}^{(t,i)}}\exp(\frac{\varepsilon_0}{2}\tilde{w}_A^{(t,i)}(u))}\right) \tag{2}$$

If $A \setminus B = \{I\}$, the second factor of (2) is bounded above by 1, and the first factor by $\exp(\frac{\varepsilon_0}{2})$. If $B \setminus A = \{I\}$, the first factor of (2) is bounded from above by 1, and the second factor by:

$$\prod_{t=1}^{T}\prod_{i=1}^{r}\mathop{\mathbb{E}}_{u}\left[e^{\frac{\varepsilon_0}{2}(G_I(y^{(t,i-1)}+\eta\mathbf{1}_u) - G_I(y^{(t,i-1)}))}\right] \leq e^{(e^{\varepsilon_0/2}-1)\sum_{(1,1)}^{(T,r)}\mathop{\mathbb{E}}_{u}\left[G_I(y^{(t,i-1)}+\eta\mathbf{1}_u) - G_I(y^{(t,i-1)})\right]}.$$

$$(\text{Since } e^x \leq 1 + \tfrac{e^{\varepsilon_0/2}-1}{\varepsilon_0/2}x\ \forall x \in [0, \tfrac{\varepsilon_0}{2}] \text{ and } 1+t \leq e^t\ \forall t)$$

Here, the expectations are over $u \leftarrow P^{(t,i)}$, where $P^{(t,i)}$ are the distributions defined by the weights of the Exponential Mechanism with respect to $\tilde{w}_A^{(t,i)}$. Consider the underlying Tr-round process, where $Z_{(t,i)}$ is the total remaining realized marginal gain with respect to G_I and $R_{(t,i)}$ its expected increase with respect to $P^{(t,i)}$. By Claim 2,

$$\mathbb{P}\left[\sum_{(1,1)}^{(T,r)}\mathop{\mathbb{E}}_{u}\left[G_I(y^{(t,i)} + \eta\mathbf{1}_u) - G(y^{(t,i)}))\right] \geq 3 + \log\frac{1}{\delta}\right] \leq \delta$$

Thus, with probability $1-\delta$, the ratio of equation (2) is at most $\exp((e^{\varepsilon_0/2}-1)(3+\log\frac{1}{\delta}))$. In general, for two neighboring sets of agents A, B, Algorithm 1 is $\big((e^{\varepsilon_0/2}-1)(4+\log\frac{1}{\delta}), \delta\big)$-DP. □

Algorithm 2 Private Measured Continuous Greedy

1: **Input**: Utility parameters $\eta, \gamma \in (0, 1]$, privacy parameters $\varepsilon, \delta \in (0, 1]$, and set of agents D.
2: Let $T \leftarrow \lceil \frac{1}{\eta} \rceil$ and $\varepsilon_0 \leftarrow \varepsilon/(14 + 4\log(1/\delta))$.
3: Draw $s = 48r^3T^7 \log(n/\gamma)$ independent random vectors such that $r^j \leftarrow \mathcal{U}^n$ for all $j \in [s]$.
4: $y^{(1,0)} = \mathbf{1}_{\emptyset}$.
5: **for** $t = 1, \dots, T$ **do**
6:     $B^{(t,0)} = \emptyset$.
7:     **for** $i = 1, \dots, r$ **do**
8:         Let $\mathcal{N}^{(t,i)} = \{u \in \mathcal{N} \setminus B^{(t,i-1)} : B^{(t,i-1)} \cup \{u\} \in \mathcal{I}\}$.
9:         **if** $\mathcal{N}^{(t,i)} = \emptyset$ **then** let $y^{(t,r)} = y^{(t,i-1)}$ and break the loop.
10:         Define $\tilde{w}_D^{(t,i)}(u) = G(y^{(t,i-1)} + \eta(1 - y_u^{(t,i-1)})\mathbf{1}_u) - G(y^{(t,i-1)})$ for all $u \in \mathcal{N}^{(t,i)}$.
11:         Let $u^{(t,i)} \leftarrow \mathcal{O}_{\varepsilon_0}(\tilde{w}_D^{(t,i)})$.
12:         Let $y^{(t,i)} = y^{(t,i-1)} + \eta(1 - y_{u^{(t,i)}}^{(t,i-1)})\mathbf{1}_{u^{(t,i)}}$.
13:         Let $B^{(t,i)} \leftarrow B^{(t,i-1)} \cup \{u^{(t,i)}\}$.
14:     $y^{(t+1,0)} = y^{(t,r)}$.
15: **return** SWAP-ROUNDING$(y^{(T,r)}, \mathcal{I})$.

Non-monotone

Algorithm 2 is an adaptation of the Measured Continuous Greedy algorithm introduced by Feldman, Naor, and Schwartz (2011). The main difference from Algorithm 1 is the update step in line 12, which also leads to a change in the definition of the marginal gains $\tilde{w}_D^{(t,i)}(u)$ in line 10.

Theorem 4. *Let $f : 2^{\mathcal{N}} \to \mathbb{R}_+$, where $|\mathcal{N}| = n$, be a non-monotone, 1-decomposable, submodular function and $\mathcal{M} = (\mathcal{N}, \mathcal{I})$ a matroid of rank r. Algorithm 2 with parameters η and γ is (ε, δ)-differentially private and returns a set $S \in \mathcal{I}$ such that, with probability $1 - \gamma$,*

$$\mathbb{E}[f(S)] \geq (1/e - O(\eta))f(\mathrm{OPT}) - O\left(\frac{r}{\eta\varepsilon}\log\frac{nr}{\eta\gamma}\cdot\log\frac{1}{\delta}\right).$$

Algorithm 2 makes $O\left(\frac{nr^4}{\eta^8}\log\frac{n}{\gamma}\right)$ oracle calls.

The utility and privacy analyses follow the main steps of their counterparts for the monotone case, with a few key modifications. The utility analysis now accounts for possibly negative marginal gains. The privacy analysis now relies on bounding a sum of expected *absolute* marginal gains, which, using Claim 2, can be bounded by the sum of realized *absolute* marginal gains. Bounding the latter is not as trivial as in the monotone case; the "movement" of a non-monotone function could be unbounded, even though the function has a bounded range, so we have to leverage the fact that f_I is submodular:

Lemma 2. *Let $f_I : 2^{[n]} \to [0, 1]$ be a submodular function. Then for any sequence of non-decreasing sets $\emptyset = T_0 \subseteq \dots \subseteq T_r \subseteq [n]$, $\sum_{i=1}^r |f_I(T_i) - f_I(T_{i-1})| \leq 2 - f_I(\emptyset)$.*

Proof. Let $S_i = \{1, \dots, i\}$. Suppose $T = \{i_t : t = 1, \dots, k\}$, for some $k \in [n]$, is the set of indices for which $f_I(S_{i_t}) - f_I(S_{i_t-1}) \geq 0$. Then, by submodularity, for $t = 1, \dots, k$ we have that

$$\begin{aligned} f_I(S_{i_t}) - f_I(S_{i_t-1}) &\leq f_I(T \cap S_{i_t}) - f_I(T \cap S_{i_t-1}) \\ &= f_I(i_1, \dots, i_t) - f_I(i_1, \dots, i_{t-1}). \end{aligned}$$

Summing over the range of $t \in [k]$, it follows that

$$\begin{aligned} &\sum_{t=1}^k |f_I(S_{i_t}) - f_I(S_{i_t-1})| = \sum_{t=1}^k f_I(S_{i_t}) - f_I(S_{i_t-1}) \\ &\leq f_I(i_1, \dots i_k) - f_I(\emptyset) \leq 1 - f_I(\emptyset) \end{aligned} \tag{3}$$

Similarly, we let $j_1, \dots j_\ell$ be the indices for which $f_I(S_{j_t}) - f_I(S_{j_t-1}) < 0$. Then, by (3),

$$\begin{aligned} &\sum_{t=1}^{\ell} |f_I(S_{j_t}) - f_I(S_{j_t-1})| \\ &= \sum_{t=1}^k f_I(S_{i_t}) - f_I(S_{i_t-1}) - f_I([n]) + f_I(\emptyset) \\ &\leq 1 - f_I([n]) \leq 1 \end{aligned} \tag{4}$$

Adding inequalities (3) and (4), we get that $\sum_{i=1}^n |f_I(S_i) - f_I(S_{i-1})| \leq 2 - f_I(\emptyset)$. Since the order of the elements of $[n]$ is arbitrary, by the triangle inequality, we get the statement of the lemma. ☐

Experiments

We describe two experiments evaluating the Private Continuous Greedy (PCG) Algorithm 1[1]. We replicate the Uber location selection experiment in (Mitrovic et al. 2017), comparing PCG and its rank-invariant noise addition with the composition law based differentially private greedy (DPG) algorithm; (Mitrovic et al. 2017, Theorem 8). We also study a hard instance of a partition matroid constraint where PCG significantly outperforms the discrete DPG with the rank-invariant privacy parameter (Mitrovic et al. 2017, Theorem 9). We use the same dataset of coordinates of Uber pick-ups[2]. The goal is to choose a set of utility-maximizing waiting locations S under the given constraints, while keeping the pick-up data differentially private. If $M(l, p)$ is the ℓ_1 distance between $l, p \in \mathbb{R}^2$ normalised to lie in $[0, 1]$ for our dataset D, we define the utility of a set S using the monotone decomposable function

$$f_D(S) = \sum_{p \in D}\left(1 - \min_{l \in S} M(l, p)\right) = |D| - \sum_{p \in D}\min_{l \in S} M(l, p). \tag{5}$$

F_D, the multilinear extension of f_D, can be computed exactly in $O(nm\log n)$ time via the closed form expression $F_{\{p\}}(x) = \sum_{i=1}^n (1 - d(p, c_i))x_i \prod_{j<i}(1 - x_j)$ (where the c_i are locations sorted in increasing order of distance from p), and summing up over all p.

[1]The code and dataset used for our experiments are available at https://github.com/Anamay-Chaturvedi/Differentially-private-decomposable-submodular-optimization

[2]https://www.kaggle.com/fivethirtyeight/uber-pickups-in-new-york-city.

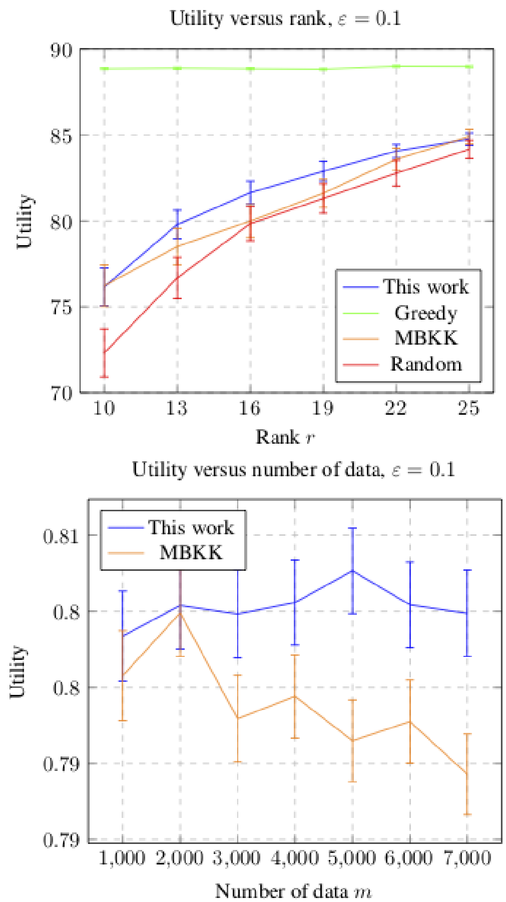

Figure 1: Empirical performance of our algorithm for the monotone case under cardinality (top) and matroid constraints (bottom).

Cardinality constraint For the r-cardinality constraint problem, our PCG algorithm and the general monotone submodular maximization algorithm DPG have additive error $\frac{r \log n}{\eta\varepsilon}$ and $\frac{r^{3/2} \log n}{\varepsilon}$, respectively.[3] We study the high rank regime, which in theory is more favourable for PCG.

Analogous to (Mitrovic et al. 2017), we use a 5×4 grid over Manhattan as potential waiting spots, but make their experiment harder by adding 80 copies of the northern corner of the grid. Due to the structure and density of the points, the original problem reduces to an easy instance of Geometric Maximum Coverage, which admits a PTAS (Li et al. 2015) and a randomly chosen set performs close to optimal for $r > 10$. The modified instance is harder for random selection, but essentially the same for the rest of the algorithms.

We draw $m = 100$ pickups uniformly at random 40 times and average the empirical utilities of DPG and our PCG over 10 runs for each draw. In PCG, we set $\eta = 0.33$ and use the closed-form expression for the multilinear relaxation of f_D. We also measure the performance of the non-private greedy which has optimal utility as a yardstick, and that of a randomly chosen basis set that serves as a trivial private baseline. We set $\varepsilon = 0.1$, $\delta = 1/m^{1.5}$ where $m = |D| = 100$, with which the privacy parameter used in the differentially private choices of increment is $\varepsilon_0 \approx 0.01006$.

In Figure 1 (top), we see that that the PCG algorithm starts to outperform the DPG algorithm around rank $r = 13$, but that both private algorithms become equivalent to picking a uniformly random set around $r = 25$. It is slightly beyond $r = 10$ that our setting for ε_0 starts to be larger than the rank-sensitive privacy parameter ε/r used in each round of the DPG algorithm, which justifies this trend. We also found that this algorithm scales well to large datasets, executing 10 runs for ranks $r = 10, 15, 20, 25$, each for datasets with $10,000$ points, in 25 minutes in total on a personal computer.

Partition matroid constraint As noted in (Mitrovic et al. 2017), in the decomposable case with matroid constraints, DPG combined with the privacy analysis of (Gupta et al. 2010) gives the optimal additive error (see Table 1). However, the $\frac{1}{2}$ approximation factor in the DPG guarantee is not a pessimistic bound but a tight one.

Suppose $S = \{A, B, C\}$ is the ground set, and $\{\{A\}, \{B, C\}\}$ the partition. We define a matroid so that member sets have at most one element per partition. For a monotone increasing submodular f the optimum must be either $\{A, B\}$ or $\{A, C\}$. If $f(\{B\}) > f(\{A\}), f(\{C\})$ but $f(\{A, B\}) < f(\{A, C\})$, then the greedy algorithm will consistently choose B, and then A for the sub-optimal output $\{A, B\}$. In particular, if $f(\{B\}) = 1$ and $f(\{A\}) = f(\{C\}) = 1-\epsilon$, but $f(\{A, B\}) = 1$ and $f(\{A, C\}) = 2-2\epsilon$ (readily extendable to a submodular function), the utility gained by greedy is at most $\frac{1}{2-2\epsilon}$ times the optimal.

We test DPG and PCG empirically in this type of worst-case instance. Although the noise induced by privacy would help DPG overcome this pitfall with some probability, our experiments show that the phenomenon described above is realized in the experiment. We pick three points in Manhattan which mimic this partition structure (with B closest to downtown, and A and C slightly further away) and compare the DPG and the PCG algorithms on a range of dataset sizes.

In Figure 1 (bottom), we compare the average utilities (normalised by the dataset size) obtained by PCG (with $\eta = 1/7$ and $\delta = 1/m^{1.5}$) and DPG with the improved privacy analysis. The error bars at each point mark the empirical standard deviation of the means. Although their performances are comparable for small datasets, the improvement of PCG increases as the dataset grows in size. There is high variance due to the random choices of the dataset for each set size, but the separation between the empirical confidence intervals still widens with larger datasets. We find that for these types of worst-case instances, compared to DPG (even with the improved privacy analysis) a significant performance enhancement can be obtained by switching to PCG for the decomposable setting.

[3] Mitrovic et al. (2017) report that DPG with the privacy parameter calculated using basic DP composition might outperform the one that uses advanced. Per instance, we check and use the best of the two for our comparison.

Acknowledgements

All the authors were supported by the National Science Foundation under NSF grants AF 1909314 and CAREER 1750716. LZ was additionally supported by NSF grants CCF- 1718088, CCF-1750640, and CNS-1816028.

References

Borgs, C.; Brautbar, M.; Chayes, J.; and Lucier, B. 2014. Maximizing Social Influence in Nearly Optimal Time. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA 14, 946–957. USA: Society for Industrial and Applied Mathematics. ISBN 9781611973389.

Borodin, A.; Jain, A.; Lee, H. C.; and Ye, Y. 2017. Max-Sum Diversification, Monotone Submodular Functions, and Dynamic Updates. *ACM Trans. Algorithms* 13(3). ISSN 1549-6325. doi:10.1145/3086464.

Buchbinder, N.; and Feldman, M. 2018. *Submodular Functions Maximization Problems*, volume 1, chapter 42. Chapman and Hall/CRC, 2nd edition.

Buchbinder, N.; and Feldman, M. 2019. Constrained Submodular Maximization via a Nonsymmetric Technique. *Mathematics of Operations Research* 44(3): 988–1005. doi: 10.1287/moor.2018.0955. URL https://doi.org/10.1287/moor.2018.0955.

Calinescu, G.; Chekuri, C.; Pál, M.; and Vondrák, J. 2011. Maximizing a Monotone Submodular Function Subject to a Matroid Constraint. *SIAM J. Comput.* 40(6): 1740–1766. ISSN 0097-5397. doi:10.1137/080733991. URL https://doi.org/10.1137/080733991.

Chekuri, C.; Vondrak, J.; and Zenklusen, R. 2010. Dependent Randomized Rounding via Exchange Properties of Combinatorial Structures. In *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, FOCS 10, 575–584. USA: IEEE Computer Society. ISBN 9780769542447. doi:10.1109/FOCS.2010.60. URL https://doi.org/10.1109/FOCS.2010.60.

Dobzinski, S.; and Schapira, M. 2006. An Improved Approximation Algorithm for Combinatorial Auctions with Submodular Bidders. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm*, SODA 06, 1064–1073. USA: Society for Industrial and Applied Mathematics. ISBN 0898716055.

Dueck, D.; and Frey, B. 2007. Non-metric affinity propagation for unsupervised image categorization. In *IEEE 11th International Conference on Computer Vision*, 1–8. ISBN 978-1-4244-1631-8. doi:10.1109/ICCV.2007.4408853.

Dwork, C.; McSherry, F.; Nissim, K.; and Smith, A. 2006. Calibrating Noise to Sensitivity in Private Data Analysis. In Halevi, S.; and Rabin, T., eds., *Theory of Cryptography*, 265–284. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-540-32732-5.

Dwork, C.; Rothblum, G. N.; and Vadhan, S. 2010. Boosting and Differential Privacy. In *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, FOCS '10, 51–60. USA: IEEE Computer Society. ISBN 9780769542447. doi:10.1109/FOCS.2010.12. URL https://doi.org/10.1109/FOCS.2010.12.

Ene, A.; and Nguyen, H. L. 2016. Constrained Submodular Maximization: Beyond 1/e. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, 248–257.

Feige, U. 2006. On Maximizing Welfare When Utility Functions Are Subadditive. In *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*, STOC 06, 41–50. New York, NY, USA: Association for Computing Machinery. ISBN 1595931341. doi:10.1145/1132516.1132523. URL https://doi.org/10.1145/1132516.1132523.

Feige, U.; and Vondrák, J. 2006. Approximation algorithms for allocation problems: Improving the factor of 1 - 1/e. In *Foundations of Computer Science, 1975., 16th Annual Symposium on*, 667 – 676. doi:10.1109/FOCS.2006.14.

Feldman, M.; Naor, J.; and Schwartz, R. 2011. A Unified Continuous Greedy Algorithm for Submodular Maximization. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, 570–579.

Fujishige, S. 2005. *Submodular Functions and Optimization*, volume 58. Elsevier, 2nd edition. ISBN 9780444520869.

Gomes, R.; and Krause, A. 2010. Budgeted Nonparametric Learning from Data Streams. In Fürnkranz, J.; and Joachims, T., eds., *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 391–398. Haifa, Israel: Omnipress. URL http://www.icml2010.org/papers/433.pdf.

Gupta, A.; Ligett, K.; McSherry, F.; Roth, A.; and Talwar, K. 2010. Differentially Private Combinatorial Optimization. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, 1106–1125. Society for Industrial and Applied Mathematics.

Hassidim, A.; and Singer, Y. 2017. Submodular Optimization under Noise. In Kale, S.; and Shamir, O., eds., *Proceedings of the 2017 Conference on Learning Theory*, volume 65 of *Proceedings of Machine Learning Research*, 1069–1122. Amsterdam, Netherlands: PMLR. URL http://proceedings.mlr.press/v65/hassidim17a.html.

Kawahara, Y.; Kiyohito, N.; Tsuda, K.; and Bilmes, J. A. 2009. Submodularity Cuts and Applications. In Bengio, Y.; Schuurmans, D.; Lafferty, J. D.; Williams, C. K. I.; and Culotta, A., eds., *Advances in Neural Information Processing Systems 22*, 916–924. Curran Associates, Inc. URL http://papers.nips.cc/paper/3762-submodularity-cuts-and-applications.pdf.

Kempe, D.; Kleinberg, J.; and Tardos, E. 2003. Maximizing the Spread of Influence through a Social Network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD 03, 137–146. New York, NY, USA: Association for Computing Machinery. ISBN 1581137370. doi:10.1145/956750.956769. URL https://doi.org/10.1145/956750.956769.

Kirchhoff, K.; and Bilmes, J. A. 2014. Submodularity for Data Selection in Statistical Machine Translation. In *Pro-*

ceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 131–141.

Krause, A.; and Cevher, V. 2010. Submodular Dictionary Selection for Sparse Representation. In Fürnkranz, J.; and Joachims, T., eds., *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 567–574. Haifa, Israel: Omnipress. URL http://www.icml2010.org/papers/366.pdf.

Krause, A.; and Guestrin, C. 2005. Near-Optimal Nonmyopic Value of Information in Graphical Models. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, UAI 05, 324–331. Arlington, Virginia, USA: AUAI Press. ISBN 0974903914.

Li, J.; Wang, H.; Zhang, B.; and Zhang, N. 2015. Linear Time Approximation Schemes for Geometric Maximum Coverage. *Computing and Combinatorics* 9198. doi:10.1016/j.tcs.2017.11.026.

Lin, H.; and Bilmes, J. 2011. A Class of Submodular Functions for Document Summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 510–520. Portland, Oregon, USA: Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P11-1052.

McSherry, F.; and Talwar, K. 2007. Mechanism Design via Differential Privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '07, 94–103. Washington, DC, USA: IEEE Computer Society. ISBN 0-7695-3010-9. doi:10.1109/FOCS.2007.41. URL http://dx.doi.org/10.1109/FOCS.2007.41.

Mirzasoleiman, B.; Badanidiyuru, A.; and Karbasi, A. 2016. Fast Constrained Submodular Maximization: Personalized Data Summarization. In Balcan, M. F.; and Weinberger, K. Q., eds., *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, 1358–1367. New York, New York, USA: PMLR. URL http://proceedings.mlr.press/v48/mirzasoleiman16.html.

Mirzasoleiman, B.; Karbasi, A.; Sarkar, R.; and Krause, A. 2016. Distributed Submodular Maximization. *Journal of Machine Learning Research* 17(235): 1–44. URL http://jmlr.org/papers/v17/mirzasoleiman16a.html.

Mirzasoleiman, B.; Zadimoghaddam, M.; and Karbasi, A. 2016. Fast Distributed Submodular Cover: Public-Private Data Summarization. In Lee, D. D.; Sugiyama, M.; Luxburg, U. V.; Guyon, I.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 29*, 3594–3602. Curran Associates, Inc. URL http://papers.nips.cc/paper/6540-fast-distributed-submodular-cover-public-private-data-summarization.pdf.

Mitrovic, M.; Bun, M.; Krause, A.; and Karbasi, A. 2017. Differentially Private Submodular Maximization: Data Summarization in Disguise. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 2478–2487. JMLR. org.

Narayanan, H. 1997. Submodular Functions and Electrical Networks. In *Annals of Discrete Mathematics*, volume 54. Elsevier, 1st edition. ISBN 9780080867946.

Nemhauser, G. L.; Wolsey, L. A.; and Fisher, M. L. 1978. An Analysis of Approximations for Maximizing Submodular Set Functions–I. *Math. Program.* 14(1): 265–294. ISSN 0025-5610. doi:10.1007/BF01588971. URL https://doi.org/10.1007/BF01588971.

Papadimitriou, C.; Schapira, M.; and Singer, Y. 2008. On the Hardness of Being Truthful. In *Proceedings of the 2008 49th Annual IEEE Symposium on Foundations of Computer Science*, FOCS 08, 250–259. USA: IEEE Computer Society. ISBN 9780769534367. doi:10.1109/FOCS.2008.54.

Puthiya Parambath, S. A.; Usunier, N.; and Grandvalet, Y. 2016. A Coverage-Based Approach to Recommendation Diversity On Similarity Graph. In *Proceedings of the 10th ACM Conference on Recommender Systems*, RecSys 16, 15–22. New York, NY, USA: Association for Computing Machinery. ISBN 9781450340359. doi:10.1145/2959100.2959149.

Rafiey, A.; and Yoshida, Y. 2020. Fast and Private Submodular and k-Submodular Functions Maximization with Matroid Constraints. In *Proceedings of the International Conference on Machine Learning*, 6443–6453.

Raz, R.; and Safra, S. 1997. A Sub-Constant Error-Probability Low-Degree Test, and a Sub-Constant Error-Probability PCP Characterization of NP. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, STOC 97, 475–484. New York, NY, USA: Association for Computing Machinery. ISBN 0897918886. doi:10.1145/258533.258641.

Schrijver, A. 2003. Combinatorial Optimization: Polyhedra and Efficiency. In *Journal of Computer and System Sciences - JCSS*, volume B. Springer.

Singla, A.; Bogunovic, I.; Bartók, G.; Karbasi, A.; and Krause, A. 2014. Near-Optimally Teaching the Crowd to Classify. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML14, II154II162. JMLR.org.

Sipos, R.; Swaminathan, A.; Shivaswamy, P.; and Joachims, T. 2012. Temporal Corpus Summarization Using Submodular Word Coverage. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM 12, 754–763. New York, NY, USA: Association for Computing Machinery. ISBN 9781450311564. doi:10.1145/2396761.2396857. URL https://doi.org/10.1145/2396761.2396857.

Vondrák, J. 2008. Optimal Approximation for the Submodular Welfare Problem in the Value Oracle Model. In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*, STOC 08, 67–74. New York, NY, USA: Association for Computing Machinery. ISBN 9781605580470. doi:10.1145/1374376.1374389. URL https://doi.org/10.1145/1374376.1374389.