Differentially Private Submodular Maximization: Data Summarization in Disguise (Full version)

Marko Mitrovic, Mark Bun, Andreas Krause, Amin Karbasi

June 12, 2017

Abstract

How can we extract representative features from a dataset containing sensitive personal information, while providing individual-level privacy guarantees? Many data summarization applications are captured by the general framework of submodular maximization. As a consequence, a wide range of efficient approximation algorithms for submodular maximization have been developed. However, when such applications involve sensitive data about individuals, their privacy concerns are not automatically addressed by these algorithms.

To remedy this problem, we propose a general and systematic study of differentially private submodular maximization. We present privacy-preserving algorithms for both monotone and non-monotone submodular maximization under cardinality, matroid, and p-extendible system constraints, with guarantees that are competitive with optimal solutions. Along the way, we analyze a new algorithm for non-monotone submodular maximization under a cardinality constraint, which is the first (even non-privately) to achieve a constant approximation ratio with a linear number of function evaluations. We additionally provide two concrete experiments to validate the efficacy of these algorithms. In the first experiment, we privately solve the facility location problem using a dataset of Uber pickup locations in Manhattan. In the second experiment, we perform private submodular maximization of a mutual information measure to select features relevant to classifying patients by diabetes status.

1 Introduction

A set function $f : 2^V \to \mathbb{R}$ is said to be *submodular* if for all sets $S \subseteq T \subseteq V$ and every element $v \in V$ we have $f(S \cup \{v\}) - f(S) \geq f(T \cup \{v\}) - f(T)$. That is, the marginal contribution of any element v to the value of the function $f(S)$ diminishes as the input set S increases. The theory of *submodular maximization* unifies and generalizes diverse problems in combinatorial optimization, including the Max-Cover, Max-Cut, and Facility Location problems. In turn, this theory has recently found numerous applications to problems in machine learning, data science, and artificial intelligence. A few such applications include exemplar-based clustering (Krause & Gomes, 2010), feature selection for classification (Krause & Guestrin, 2005), document and corpus summarization (Lin & Bilmes, 2011; Kirchhoff & Bilmes, 2014; Sipos et al., 2012), crowd teaching (Singla et al., 2014), and influence maximization in social networks (Kempe et al., 2003).

Some of the most compelling use cases for these applications concern sensitive data about individuals (Mirzasoleiman et al., 2016a,b). As a running example, let us consider the specific problem of determining which of a collection of features (e.g. age, height, weight, etc.) are most relevant to a binary classification task (e.g. predicting whether an individual is likely to have diabetes). In this problem, a sensitive training

set takes the form $D = \{(x_i, y_i)\}_{i=1}^n$ where each individual i's data consists of features $x_{i,1}, \ldots, x_{i,m}$ together with a class label y_i. The goal is to identify a small subset $S \subseteq [m]$ of features which can then be used to build a good classifier for y. Many techniques exist for feature selection, including one based on maximizing a submodular function which captures the mutual information between a subset of features and the class label of interest (Krause & Guestrin, 2005). However, for both legal (e.g. compliance with HIPAA regulations) and ethical reasons, it is important that the selection of relevant features does not compromise the privacy of any individual who has contributed to the training data set. Unfortunately, the theory of submodular maximization does not in itself accommodate such privacy concerns.

To this end, we propose a systematic study of *differentially private submodular maximization* to enable these applications based on submodular maximization, while provably guaranteeing individual-level privacy. The notion of differential privacy (Dwork et al., 2006) offers a strong protection of individual-level privacy. Nevertheless, differential privacy has been shown to permit useful data analysis and machine learning tasks. In a nutshell, the definition formalizes a guarantee that no individual's data should have too significant an effect on the outcome of a computation. We provide the formal definition in Section 2. Such a privacy guarantee is obtained through the introduction of random noise, so private submodular maximization is conceptually related to the problem of submodular maximization in the presence of noise (Cheraghchi, 2012; Hassidim & Singer, 2016).

In this work, we study the following problem under various conditions on the submodular objective function f (monotone vs. non-monotone), and various choices of the constraint $\mathcal{C}$ (cardinality, matroid, or p-extendible system).

Problem 1.1. *Given a sensitive dataset D associated to a submodular function $f_D : 2^V \to \mathbb{R}$: Find a subset $S \in \mathcal{C} \subset 2^V$ that approximately maximizes $f_D(S)$ in a manner that guarantees differential privacy with respect to the input dataset D.*

An important special case of this problem was studied in prior work of Gupta et al. (2010). They considered the "combinatorial public projects" problem (Papadimitriou et al., 2008), where given a dataset $D = (x_1, \ldots, x_n)$, the function f_D takes the particular form $f_D(S) = \frac{1}{n}\sum_{i=1}^n f_{x_i}(S)$ for monotone submodular functions $f_{x_i} : 2^V \to [0, 1]$, and is to be maximized subject to a cardinality constraint $|S| \leq k$. We call functions of this form *decomposable*. They presented a simple greedy algorithm, which will be central to our work, together with a tailored analysis which achieves strong accuracy guarantees in this special case.

However, there are many cases of Problem 1.1 which do not fall into the combinatorial public projects framework. For some problems, including feature selection via mutual information, the submodular function f_D of interest depends on the dataset D in ways much more complicated than averaging functions associated to each individual. The focus of our work is on understanding Problem 1.1 in circumstances which capture a broader class of useful applications and constraints in machine learning. We summarize our specific contributions in Section 1.2.

1.1 The greedy paradigm

Even without concern for privacy, the problem of submodular maximization poses computational challenges. In particular, exact submodular maximization subject to a cardinality constraint is **NP**-hard. One of the principal approaches to designing efficient approximation algorithms is to use a greedy strategy (Nemhauser et al., 1978). Consider the problem of maximizing a set function $f(S)$ subject to the cardinality constraint $|S| \leq k$. In each of rounds $i = 1, \ldots, k$, the basic greedy algorithm constructs S_i from S_{i-1} by adding the element $v_i \in (V \setminus S_{i-1})$ which maximizes the marginal gain $f(S_{i-1} \cup \{v_i\}) - f(S_{i-1})$. Nemhauser et al.

	Cardinality	Matroid	p-Extendible
Comb. Pub. Proj.	$\left(1-\frac{1}{e}\right)\mathrm{OPT}-O\left(\frac{k\log\lvert V\rvert}{n}\right)$ (Gupta et al., 2010)	$\frac{1}{2}\mathrm{OPT}-O\left(\frac{k\log\lvert V\rvert}{n}\right)$	$\frac{1}{p+1}\mathrm{OPT}-O\left(\frac{k\log\lvert V\rvert}{n}\right)$
Monotone	$\left(1-\frac{1}{e}\right)\mathrm{OPT}-O\left(\frac{k^{3/2}\log\lvert V\rvert}{n}\right)$	$\frac{1}{2}\mathrm{OPT}-O\left(\frac{k^{3/2}\log\lvert V\rvert}{n}\right)$	$\frac{1}{p+1}\mathrm{OPT}-O\left(\frac{k^{3/2}\log\lvert V\rvert}{n}\right)$
Non-monotone	$\frac{1}{e}\left(1-\frac{1}{e}\right)\mathrm{OPT}-O\left(\frac{k^{3/2}\log\lvert V\rvert}{n}\right)$	–	–

Table 1: Guarantees of expected solution quality for privately maximizing a sensitivity-$(1/n)$ submodular function f_D. The parameter k represents either a cardinality constraint, or the size of the set returned (for matroid or p-extendible system constraints). Full expressions with explicit dependencies on differential privacy parameters ε, δ appear in the body of the paper.

(1978) famously showed that this algorithm yields a $(1-1/e)$-approximation to the optimal value of $f(S)$ whenever f is a monotone submodular function.

In the combinatorial public projects setting, Gupta et al. (2010) showed how to make the greedy algorithm compatible with differential privacy by randomizing the procedure for selecting each v_i. This selection procedure is specified by the differentially private *exponential mechanism* of McSherry & Talwar (2007), which (probabilistically) guarantees that the v_i selected in each round is almost as good as the true marginal gain maximizer. Remarkably, Gupta et al. (2010) show that the cumulative privacy guarantee of the resulting randomized greedy algorithm is not much worse than that of a single run of the exponential mechanism. This analysis is highly tailored to the structure of the combinatorial public projects problem. However, replacing this tailored analysis with the more generic "advanced composition theorem" for differential privacy (Dwork et al., 2010), one still obtains useful results for the more general class of "low-sensitivity" submodular functions.

1.2 Our contributions

Table 1 summarizes the approximation guarantees we obtain for Problem 1.1 under increasingly more general classes of submodular functions f_D (read top to bottom), and increasingly more general types of constraints (read left to right). In each entry, OPT denotes the value of the optimal non-private solution. Below we draw attention to a few particular contributions, including some that are not expressed in Table 1.

Non-monotone objective functions. Submodular maximization for non-monotone functions is significantly more challenging than it is for monotone objectives. In particular, the basic greedy algorithm of Nemahauser et al. fails, and cannot guarantee any constant-factor approximation. Several works (Feldman et al., 2017; Mirzasoleiman et al., 2016a; Buchbinder et al., 2014; Feldman et al., 2011) have identified variations of the greedy algorithm that do yield constant-factor approximations for non-monotone objectives. However, it is not clear how to modify any of these algorithms to accommodate differential privacy.

Our starting point is instead the "stochastic greedy" algorithm of Mirzasoleiman et al. (2015), which was originally designed to perform *monotone* submodular maximization in linear time. Drawing ideas from Buchbinder et al. (2014), we give a new analysis of the stochastic greedy algorithm to show that it also gives a $\frac{1}{e}(1-1/e)$-approximation for non-monotone submodular functions. To our knowledge, this is the first algorithm making exactly $|V|$ function evaluations which achieves a constant-factor approximation for either monotone or non-monotone objectives. Moreover, it is immediately clear how to use the exponential mechanism to make this algorithm differentially private.

This phenomenon is analogous to how stochastic variants of gradient descent are more naturally suited to

providing differential privacy than their deterministic counterparts (Song et al., 2013; Bassily et al., 2014). That is, our results illustrate how techniques for making algorithms *fast* are also helpful in making them *privacy-preserving*.

General constraints. While a cardinality constraint is perhaps the most natural to place on a submodular maximization problem, some machine learning problems, e.g. personalized data summarization (Mirzasoleiman et al., 2016a), require the use of more general types of constraints. For instance, one may wish to maximize a submodular function $f(S)$ subject to $S \in \mathcal{I}$ for an arbitrary matroid $\mathcal{I}$, or subject to S being contained in an intersection of p matroids (more generally, a p-extendible system). For these types of constraints, the greedy algorithm still yields a constant factor approximation for monotone objective functions (Fisher et al., 1978; Jenkyns, 1976; Călinescu et al., 2011). We show in this work that the analysis provided by Călinescu et al. (2011) for matroids and p-extendible families can be adapted to handle additional error introduced for differential privacy.

General selection procedures. For worst-case datasets, the exponential mechanism is optimal within each round of private maximization. However, it may be sub-optimal for datasets enjoying additional structural properties. Fortunately, the greedy framework we use is flexible with regard to the choice of the selection procedure. For instance, one can replace the exponential mechanism in a black-box manner with the "large margin mechanism" of Chaudhuri et al. (2014) to obtain error bounds that replace the explicit dependence on $\log |V|$ in Table 1 with a term that may be significantly smaller for real datasets. We give a slightly simplified analysis of the large margin mechanism, and present it in a manner suitable for greedy algorithms which access the same data set multiple times. (These guarantees are more complicated, but spelled out in Section 5.) For submodular functions exhibiting additional structure, one may also be able to perform each maximization step with the "choosing mechanism" of Beimel et al. (2016) and Bun et al. (2015).

2 Preliminaries

Let V be finite set which we will refer to as the *ground set* and let X be a finite set which we will refer to as the *data universe*. A dataset is an n-tuple $D = (x_1, \ldots, x_n) \in X^n$. Suppose each dataset D is associated to a set function $f_D : 2^V \to \mathbb{R}$. The manner in which f_D depends on D will be application-specific, but it is assumed that the association between D and f_D is public information.

Definition 2.1. A set function $f_D : 2^V \to \mathbb{R}$ is *submodular* if for all sets $S \subseteq T \subseteq V$ and every element $v \in V$, we have $f_D(S \cup \{v\}) - f_D(S) \geq f_D(T \cup \{v\}) - f_D(T)$.

Moreover, If $f_D(S) \leq f_D(T)$ whenever $S \subseteq T$, we say f_D is *monotone*. If for every dataset $D = (x_1, \ldots, x_n)$, the function $f_D = \frac{1}{n} \sum_{i=1}^{n} f_{x_i}$ for monotone submodular functions $f_{x_i} : 2^V \to [0, \lambda]$, we say f_D is λ-*decomposable*. The problem of maximizing a decomposable submodular function was considered as the "combinatorial public projects problem" by Papadimitriou et al. (2008).

We are interested in the problem of approximately maximizing a submodular function subject to differential privacy. The definition of differential privacy relies on the notion of *neighboring* datasets, which are simply tuples $D, D' \in X^n$ that differ in at most one entry. If D, D' are neighboring, we write $D \sim D'$.

Definition 2.2. A randomized algorithm $M : X^n \to \mathcal{R}$ satisfies (ε, δ)-*differential privacy* if for all measurable sets $T \subseteq \mathcal{R}$ and all neighboring datasets $D \sim D'$,

$$\Pr[M(D) \in T] \leq e^{\varepsilon} \Pr[M(D') \in T] + \delta.$$

Differentially private algorithms must be calibrated to the sensitivity of the function of interest with respect to small changes in the input dataset, defined formally as follows.

Definition 2.3. The *sensitivity* of a set function $f_D : 2^V \to \mathbb{R}$ (depending on a dataset D) with respect to a constraint $\mathcal{C} \subseteq 2^V$ is defined as

$$\max_{D \sim D'} \max_{S \in \mathcal{C}} |f_D(S) - f_{D'}(S)|.$$

Composition of Differential Privacy. The analyses of our algorithms rely crucially on *composition theorems* for differential privacy. For a sequence of privacy parameters $\{(\varepsilon_i, \delta_i)\}_{i=1}^k$, we informally refer to the *k-fold adaptive composition* of $(\varepsilon_i, \delta_i)$-differentially private algorithms as the output of a mechanism M^* that behaves as follows on an input D: In each of rounds $i = 1, \ldots, k$, the algorithm M^* selects an $(\varepsilon_i, \delta_i)$-differentially private algorithm M_i possibly depending on the previous outcomes $M_1(D), \ldots, M_i(D)$ (but *not* directly on the sensitive dataset D itself), and releases $M_i(D)$. For a formal treatment of adaptive composition, see (Dwork et al., 2010; Dwork & Roth, 2014).

Theorem 2.4. *(Dwork & Lei, 2009; Dwork et al., 2010; Bun & Steinke, 2016) The k-fold adaptive composition of $(\varepsilon_0, \delta_0)$-differentially private algorithms satisfies (ε, δ)-differential privacy where*

1. *$\varepsilon = k\varepsilon_0$ and $\delta = k\delta_0$. (Basic Composition).*
2. *$\varepsilon = \frac{1}{2}k\varepsilon_0^2 + \sqrt{2\log(1/\delta')}\varepsilon_0$ and $\delta = \delta' + k\delta$, for any $\delta' > 0$. (Advanced Composition)*

Exponential Mechanism. The exponential mechanism (McSherry & Talwar, 2007) is a general primitive for solving discrete optimization problems. Let $q : V \times X^n \to \mathbb{R}$ be a "quality" function measuring how good a solution $v \in V$ is with respect to a dataset $D \in X^n$. We say a quality function q has *sensitivity* λ if for all $v \in V$ and all neighboring datasets $D \sim D'$, we have $|q(v, D) - q(v, D')| \leq \lambda$.

Proposition 2.5. *Let $\varepsilon > 0$ and let $q : V \times X^n$ be a quality function with sensitivity λ. Define the* exponential mechanism *as the algorithm which selects every $v \in V$ with probability proportional to $\exp(\varepsilon q(v, D)/2\lambda)$.*

- *The exponential mechanism provides $(\varepsilon, 0)$-differential privacy.*
- *For every $D \in X^n$,*

$$\mathbb{E}[q(\hat{v}, D)] \geq \max_{v \in V} q(v, D) - \frac{2\lambda \cdot \ln |V|}{\varepsilon},$$

where $\hat{v}$ is the output of the exponential mechanism on dataset D.

The privacy guarantee and a "with high probability" utility guarantee of the exponential mechanism are due to McSherry & Talwar (2007). A simple proof of the utility guarantee in expectation appears in (Bassily et al., 2016).

3 Monotone Submodular Maximization

In this section, we present a variant of the basic greedy algorithm which will enable maximization of monotone submodular functions. This algorithm simply replaces each greedy selection step with a privacy-preserving selection algorithm denoted $\mathcal{O}$. The selection function $\mathcal{O}$ takes as input a quality function $q : U \times X^n \to \mathbb{R}$ and a dataset D, as well as privacy parameters ε_0, δ_0, and outputs an element $u \in U$.

We begin in the simplest case of monotone submodular maximization with a cardinality constraint (Algorithm 1). The algorithm for more general constraints appears in Section 3.1.

Algorithm 1 was already studied by Gupta et al. (2010) in the special case where f_D is decomposable, and $\mathcal{O}$ is the exponential mechanism. We generalize their result to the much broader class of low-sensitivity monotone submodular functions.

Algorithm 1 Diff. Private Greedy (Cardinality) $\mathcal{G}^{\mathcal{O}}$

Input: Submodular function $f_D : 2^V \to \mathbb{R}$, dataset D, cardinality constraint k, privacy parameters ε_0, δ_0
Output: Size k subset of V

1. Initialize $S_0 = \emptyset$
2. For $i = 1, \ldots, k$:
   - Define $q_i : (V \setminus S_{i-1}) \times X^n \to \mathbb{R}$ via $q_i(v, \tilde{D}) = f_{\tilde{D}}(S_{i-1} \cup \{v\}) - f_{\tilde{D}}(S_{i-1})$
   - Compute $v_i \leftarrow_{\mathrm{R}} \mathcal{O}(q_i, D; \varepsilon_0, \delta_0)$
   - Update $S_i \leftarrow (S_{i-1} \cup \{v_i\})$
3. Return S_k

Theorem 3.1. *(Gupta et al., 2010) Suppose $f_D : 2^V \to \mathbb{R}$ is λ-decomposable (cf. Definition 2.1). Let $\delta > 0$ and let $\varepsilon_0 \geq 0$ be such that $\varepsilon = 2 \cdot \varepsilon_0 \cdot (e-1)\ln(3e/\delta) \leq 1$. Then instantiating Algorithm 1 with $\mathcal{O} = \mathrm{EM}$ and parameter $\varepsilon_0 > 0$ provides (ε, δ)-differential privacy.*

Moreover, for every $D \in X^n$,

$$\mathbb{E}\left[f_D(S_k)\right] \geq \left(1 - \frac{1}{e}\right) \mathrm{OPT} - \frac{2\lambda k \ln |V|}{\varepsilon_0}$$

where $S_k \leftarrow_R \mathcal{G}^{\mathrm{EM}}(D)$.

Unfortunately, the privacy analysis of Theorem 3.1 makes essential use of the decomposability of f_D, and does not directly generalize to arbitrary submodular functions of low-sensitivity. Replacing the privacy analysis of Gupta et al. (2010) with the Composition Theorem 2.4 instead gives

Theorem 3.2. *Suppose $f_D : 2^V \to \mathbb{R}$ is monotone and has sensitivity λ. Then instantiating Algorithm 1 with $\mathcal{O} = \mathrm{EM}$ and parameter $\varepsilon_0 > 0$ provides $(\varepsilon = k\varepsilon_0, \delta = 0)$-differential privacy. It also provides (ε, δ)-differential privacy for every $\delta > 0$ with $\varepsilon = k\varepsilon_0^2/2 + \varepsilon_0 \cdot \sqrt{2k\ln(1/\delta)}$.*

Moreover, for every $D \in X^n$,

$$\mathbb{E}\left[f_D(S_k)\right] \geq \left(1 - \frac{1}{e}\right) \mathrm{OPT} - \frac{2\lambda k \ln |V|}{\varepsilon_0}$$

where $S_k \leftarrow_R \mathcal{G}^{\mathrm{EM}}(D)$.

Proof. The privacy guarantee of Theorem 3.2 follows immediately from the $(\varepsilon, 0)$-differential privacy of the exponential mechanism, together with Theorem 2.4.

To simplify notation in the utility proofs in this paper, we suppress the dependence of the submodular function of interest on D, i.e. we write $f = f_D$. We also introduce the notation $f_S(T) = f(S \cup T) - f(S)$ to denote the marginal gain by adding T to the set S.

To argue that the algorithm achieves good utility, recall that in each step i, the exponential mechanism guarantees a solution v_i with

$$\mathbb{E}[f_{S_{i-1}}(v_i)] \geq \max_{v \in V \setminus S_{i-1}} f_{S_{i-1}}(v) - \alpha \tag{1}$$

where $\alpha = 2\lambda \cdot \ln |V|/\varepsilon$.

Let S^* denote any set of size k with $f(S^*) = \mathrm{OPT}$. Below, let us condition on having obtained some set S_{i-1} of elements after the first $i-1$ iterations of our algorithm. Then

$$\begin{aligned}
\mathbb{E}[f_{S_i}(v_i)] &= \max_{v \in V \setminus S_{i-1}} f_{S_{i-1}}(v) - \alpha && \text{(by Condition (1))} \\
&\geq \frac{1}{k}\left(\sum_{v \in S^*} f_{S_{i-1}}(v)\right) - \alpha \\
&\geq \frac{f(S^* \cup S_{i-1}) - f(S_{i-1})}{k} - \alpha && \text{(by submodularity of } f\text{)} \\
&\geq \frac{\mathrm{OPT} - f(S_{i-1})}{k} - \alpha && \text{(by monotonicity of } f\text{)}
\end{aligned}$$

We now unfix from conditioning on having obtained a specific S_{i-1} by taking the expectation over all choices of such a set. This gives

$$\mathbb{E}[f_{S_{i-1}}(v_i)] \geq \frac{\mathrm{OPT} - \mathbb{E}[f(S_{i-1})]}{k} - \alpha$$

Rearranging yields

$$\mathrm{OPT} - \mathbb{E}[f(S_i)] \leq \left(1 - \frac{1}{k}\right)(\mathrm{OPT} - \mathbb{E}[f(S_{i-1})]) + \alpha$$

Recursively applying this bound yields

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

Hence, we conclude

$$\begin{aligned}
\mathbb{E}[f(S_k)] &\geq \left[1 - \left(1 - \frac{1}{k}\right)^k\right] \mathrm{OPT} - \alpha \\
&\geq \left(1 - \frac{1}{e}\right) \mathrm{OPT} - \alpha.
\end{aligned}$$

□

3.1 Matroid and p-Extendible System Constraints

We now show how to extend Algorithm 1 to privately maximize monotone submodular functions subject to more general constraints. To start, we review the definition of a p-extendible system. Consider a ground set V and a non-empty downward-closed family of subsets $\mathcal{I} \subseteq 2^V$ (i.e. if $T \in \mathcal{I}$, then $S \in \mathcal{I}$ for every $S \subseteq T$). Such an $\mathcal{I}$ is called a family of *independent sets*. The pair $(V, \mathcal{I})$ is said to be a *p-extendible system* (Mestre, 2006) if for all $S \subset T \in \mathcal{I}$, and $v \in V$ such that $S \cup \{v\} \in \mathcal{I}$, there exists a set $Z \subseteq (T \setminus S)$ such that $|Z| \leq p$ and $(T \setminus Z) \cup \{v\} \in \mathcal{I}$. Let $r(\mathcal{I})$ denote the size of the largest independent set in $\mathcal{I}$.

The definition of a *matroid* coincides with that of a 1-extendible system (with *rank* $r(\mathcal{I})$). For $p \geq 2$, the notion of a p-extendible system strictly generalizes that of an intersection of p matroids. A slight modification of Algorithm 1 gives a unified algorithm for privately maximizing a monotone submodular function subject to matroid and p-extendible system constraints, presented as Algorithm 2.

We obtain analogues of the results presented for cardinality constraints.

Theorem 3.3. *Suppose $f_D : 2^V \to \mathbb{R}$ is λ-decomposable (cf. Definition 2.1). Let $\delta > 0$ and let $\varepsilon_0 \geq 0$ be such that $\varepsilon = 2 \cdot \varepsilon_0 \cdot (e-1)\ln(3e/\delta) \leq 1$. Then instantiating Algorithm 2 with $\mathcal{O} = \mathrm{EM}$ and parameter $\varepsilon_0 > 0$ provides (ε, δ)-differential privacy. Moreover, for every $D \in X^n$,*

$$\mathbb{E}\left[f_D(S)\right] \geq \frac{1}{p+1} \cdot \mathrm{OPT} - \frac{p}{p+1}\left(\frac{2\lambda r(\mathcal{I}) \ln |V|}{\varepsilon_0}\right)$$

where $S \leftarrow_R \mathcal{G}^{\mathrm{EM}}(D)$.

Algorithm 2 Differentially Private Greedy (p-system) $\mathcal{G}^{\mathcal{O}}$

Input: Submodular function $f_D : 2^V \to \mathbb{R}$, dataset D, p-extendible family $(V, \mathcal{I})$, privacy parameters ε_0, δ_0
Output: Maximal independent subset of V

1. Initialize $S = \emptyset$
2. While $S \in \mathcal{I}$ is not maximal:
   - Define $q : (V \setminus S) \times X^n \to \mathbb{R}$ via $q(v, \tilde{D}) = f_{\tilde{D}}(S \cup \{v\}) - f_{\tilde{D}}(S)$
   - Compute $v_i \leftarrow_R \mathcal{O}(q, D; \varepsilon_0, \delta_0)$
   - Update $S \leftarrow (S \cup \{v_i\})$
3. Return S

Proof. The privacy guarantee of Theorem 3.3 follows from Gupta et al. (2010).

In our proof of utility, we again suppress the dataset D, and use the notation $f_S(T)$ to denote $f(S \cup T) - f(S)$. Our proof applies to any greedy algorithm that, in each round i, selects an item v_i with

$$\mathbb{E}\left[f_{S_{i-1}}(v_i)\right] \geq \max_{v : S_{i-1} \cup \{v\} \in \mathcal{I}} f_{S_{i-1}}(v_i) - \alpha \tag{2}$$

for some error term $\alpha > 0$.

We follow the proof outlined by Călinescu et al. (2011). Fix an optimal solution $O \in \mathcal{I}$, i.e. $f(O) = \mathrm{OPT}$. Let $S_1, \ldots, S_r$ be any sequence representing the output of the algorithm, where $r = r(\mathcal{I})$. (If

the algorithm terminates in an earlier round $k < r$, then extend its output by setting $S_i = S_k$ for each $i = k+1, \ldots, r$.) To such a sequence, we define a partition $O_1, \ldots, O_r$ of O via the following algorithm.

Algorithm 3 Partition construction algorithm

Input: Optimal solution O, sets $S_1, \ldots, S_r$
Output: A partition $O_1, O_2, \ldots, O_r$ of O

1. Initialize $T_0 = O$
2. For $i = 1, 2, \ldots, r$:
   (a) If $v_i \in T_{i-1}$, set $O_i = \{v_i\}$;
   Else, let $O_i \subseteq T_{i-1}$ be the smallest subset s.t. $((S_{i-1} \cup T_{i-1}) \setminus O_i) \cup \{v_i\} \in \mathcal{I}$
   (b) Set $T_i = T_{i-1} \setminus O_i$
3. Return $O_1, O_2, \ldots, O_r$

To see that $O_1, \ldots, O_r$ is indeed a partition, observe that $S_i \cup T_i \in \mathcal{I}$ and $S_i \cap T_i = \emptyset$ for every i. Therefore, it must be the case that $T_r = \emptyset$, since $S_r \cup T_r \in \mathcal{I}$ and S_r is maximal when the algorithm terminates. Hence, the disjoint sets $O_1, \ldots, O_r$ do in fact exhaust O.

Lemma 3.4. *For every $i = 1, \ldots, r$, we have $\mathbb{E}\left[f_{S_{i-1}}(v_i)\right] \geq \frac{1}{p}\mathbb{E}\left[f_{S_{i-1}}(O_i)\right] - \alpha$.*

Before proving Lemma 3.4, we show how to use it to complete the proof of Threorem 3.3. Recursively applying the lemma shows that for every i,

$$\mathbb{E}\left[f(S_i)\right] \geq \frac{1}{p}\sum_{j=1}^{i}\mathbb{E}\left[f_{S_{j-1}}(O_j)\right] - i\alpha.$$

Hence, we obtain

$$\begin{aligned}
\mathbb{E}\left[f(S_r)\right] &\geq \frac{1}{p}\sum_{i=1}^{r}\mathbb{E}\left[f_{S_{i-1}}(O_i)\right] - r\alpha \\
&\geq \frac{1}{p}\sum_{i=1}^{r}\mathbb{E}\left[f_{S_r}(O_i)\right] - r\alpha && \text{(by submodularity)} \\
&\geq \frac{1}{p}\mathbb{E}\left[f_{S_r}(O)\right] - r\alpha && \text{(by linearity of expectation and submodularity)} \\
&\geq \frac{1}{p}\left(f(O) - \mathbb{E}\left[f(S_r)\right]\right) - r\alpha. && \text{(by monotonicity)}
\end{aligned}$$

Rearranging gives the desired result $\mathbb{E}\left[f(S_r)\right] \geq \frac{1}{p+1}f(O) - \frac{p}{p+1}r\alpha$.

□

Proof of Lemma 3.4. The partition construction algorithm that every set O_i satisfies $|O_i| \leq p$; this follows from the definition of p-extendibility and the fact that $S_{i-1} \cup \{v_i\} \in \mathcal{I}$. Moreover, any element in O_i is a candidate for inclusion in S_i, since $S_{i-1} \cup \{v\} \in \mathcal{I}$ for every $v \in O_i$.

It is also clear from the partition construction that for each $v \in O_i$, we have $S_{i-1} \cup \{v\} \in \mathcal{I}$. Below, fix a choice of i and condition on the algorithm's history up to iteration $i-1$. This fixes choices of the sets $S_1, \ldots, S_{i-1}$, as well as $T_1, \ldots, T_i$ and $O_1, \ldots, O_i$.

Then since $\mathbb{E}\left[f_{S_{i-1}}(v_i)\right] \geq f_{S_{i-1}}(v) - \alpha$ for every $v \in O_i$, we have

$$\begin{aligned}\mathbb{E}\left[f_{S_{i-1}}(v_i)\right] &\geq \frac{1}{|O_i|} f_{S_{i-1}}(O_i) - \alpha && \text{(by submodularity)} \\ &\geq \frac{1}{p} f_{S_{i-1}}(O_i) - \alpha.\end{aligned}$$

Taking the expectation over the conditioned event gives the asserted result.

□

Theorem 3.5. *Suppose $f_D : 2^V \to \mathbb{R}$ has sensitivity λ. Then instantiating Algorithm 2 with $\mathcal{O} = \mathrm{EM}$ and parameter $\varepsilon_0 > 0$ provides $(\varepsilon = r(\mathcal{I})\varepsilon_0, \delta = 0)$-differential privacy. It also provides (ε, δ)-differential privacy for every $\delta > 0$ with $\varepsilon = r(\mathcal{I})\varepsilon^2/2 + \varepsilon \cdot \sqrt{2r(\mathcal{I})\ln(1/\delta)}$.*

Moreover, for every $D \in X^n$,

$$\mathbb{E}\left[f_D(S)\right] \geq \frac{1}{p+1} \cdot \mathrm{OPT} - \frac{p}{p+1}\left(\frac{2\lambda r(\mathcal{I}) \ln |V|}{\varepsilon_0}\right)$$

where $S \leftarrow_R \mathcal{G}^{\mathrm{EM}}(D)$.

Proof. The proof of privacy follows from Theorem 2.4. The proof of utility is identical to that of the proof of Theorem 3.3. □

4 Non-Monotone Submodular Maximization

We now consider the problem of privately maximizing an arbitrary, possibly non-monotone, submodular function under a cardinality constraint. In general, the greedy algorithm presented in Section 3 fails to give any constant-factor approximation. Instead, our algorithm in this section will be based on the "stochastic greedy" algorithm first studied by Mirzasoleiman et al. (2015). In each round, the stochastic greedy algorithm first subsamples a random $\frac{1}{k} \cdot \ln(1/\alpha)$ fraction of the ground set for some $\alpha > 0$, and then greedily selects the item from this subsample that maximizes marginal gain. Mirzasoleiman et al. (2015) showed that for a monotone objective function f, this algorithm provides a $(1 - 1/e - \alpha)$-approximation to the optimal solution. Their original motivation was to improve the running time of the greedy algorithm: from $O(|V| \cdot k)$ evaluations of the objective function to linear $O(|V| \cdot \ln(1/\alpha))$.

Unfortunately, the stochastic greedy algorithm does not provide any approximation guarantee for non-monotone submodular functions. Buchbinder et al. (2014) instead proposed a "random greedy" algorithm that, in each iteration, randomly selects one of the k elements with the highest marginal gain. Buchbinder et al. (2014) showed that the random greedy algorithm achieves a $1/e$ approximation to the optimal solution (in expectation), using $k|V|$ function evaluations. However, it is not clear how to adapt this algorithm to accommodate differential privacy, since its analysis has a brittle dependence on the sampling procedure.

We make two main contributions to the analysis of the stochastic greedy and random greedy algorithms. First, we show that running the stochastic greedy algorithm on an exact $\frac{1}{k}$ fraction of the ground set per iteration still gives a (0.468)-approximation for monotone objectives, and moreover, gives a $\frac{1}{e}(1 - 1/e)$-approximation even for non-monotone objectives. Note that this algorithm evaluates the objective function on only $|V|$ elements, and still provides a constant factor approximation guarantee. This makes our

"subsample-greedy" algorithm the fastest algorithm for maximizing a general submodular function subject to a cardinality constraint (albeit with slightly worse approximation guarantees). Second, we show that the guarantees of this algorithm are robust to using a randomized greedy selection procedure (e.g. the exponential or large margin mechanism), and hence it can be adapted to ensure differential privacy.

We present the subsample-greedy algorithm as Algorithm 4 below. Assume that V is augmented by enough "dummy elements" to ensure that $|V|/k$ is an integer; each dummy element u is defined so that $f_D(S \cup \{u\}) = f_D(S)$ for every set S. We also explicitly account for an additional set U of k dummy elements, and ensure that at least one appears in every subsample.

Algorithm 4 Diff. Private "Subsample-Greedy" $\mathcal{SG}^{\mathcal{O}}$

Input: Submodular function $f_D : 2^V \to \mathbb{R}$, dataset D, cardinality constraint k, privacy parameters ε_0, δ_0
Output: Size k subset of V

1. Initialize $S_0 = \emptyset$, dummy elements $U = \{u^1, \ldots, u^k\}$
2. For $i = 1, \ldots, k$:
   - Sample $V_i \subset V$ a uniformly random subset of size $|V|/k$ and u_i a random dummy element
   - Define $q_i : (V_i \cup \{u_i\}) \times X^n \to \mathbb{R}$ via $q_i(v, \tilde{D}) = f_{\tilde{D}}(S_{i-1} \cup \{v\}) - f_{\tilde{D}}(S_{i-1})$
   - Compute $v_i \leftarrow_{\mathrm{R}} \mathcal{O}(q_i, D; \varepsilon_0, \delta_0)$
   - Update $S_i \leftarrow (S_{i-1} \cup \{v_i\})$
3. Return S_k with all dummy elements removed

Theorem 4.1. *Suppose $f_D : 2^V \to \mathbb{R}$ has sensitivity λ. Then instantiating Algorithm 4 with $\mathcal{O} = \mathrm{EM}$ provides (ε, δ)-differential privacy, and for every $D \in X^n$,*

$$\mathbb{E}\left[f_D(S)\right] \geq \frac{1}{e}\left(1 - \frac{1}{e}\right)\mathrm{OPT} - \frac{2\lambda k \ln |V|}{\varepsilon}$$

where $S \leftarrow_R \mathcal{SG}^{\mathrm{EM}}(D)$. Moreover, if f_D is monotone, then

$$\begin{aligned}\mathbb{E}\left[f_D(S)\right] &\geq \left(1 - e^{-(1-1/e)}\right)\mathrm{OPT} - \frac{2\lambda k \ln |V|}{\varepsilon} \\ &\approx 0.468\,\mathrm{OPT} - \frac{2\lambda k \ln |V|}{\varepsilon}.\end{aligned}$$

The guarantees of Theorem 4.1 are of interest even without privacy. Letting MAX denote the selection procedure which simply outputs the true maximizer (equivalently, which runs the exponential mechanism with $\varepsilon_0 = +\infty$), we obtain the following non-private algorithm for maximizing a submodular function f_D:

Corollary 4.2. *Let $f_D : 2^V \to \mathbb{R}$ be any submodular function. Instantiating Algorithm 4 with $\mathcal{O} = \mathrm{MAX}$ gives*

$$\mathbb{E}\left[f_D(S)\right] \geq \frac{1}{e}\left(1 - \frac{1}{e}\right)\mathrm{OPT}$$

where $S \leftarrow_R \mathcal{SG}^{\mathrm{MAX}}(D)$. Moreover, if f_D is monotone, then

$$\mathbb{E}\left[f_D(S)\right] \geq \left(1 - e^{-(1-1/e)}\right)\mathrm{OPT} \approx 0.468\,\mathrm{OPT}\,.$$

4.1 Proof of Theorem 4.1

The analysis below will work generally for any random selection procedure guaranteeing that in every round $i = 1, \ldots, k$,

$$\mathbb{E}\left[f_{S_{i-1}}(v_i)\right] \geq \max_{v \in (V_i \cup \{u_i\})} f_{S_{i-1}}(v) - \alpha$$

for some parameter $\alpha > 0$. We begin by fixing an optimal solution S^* with $f(S^*) = \mathrm{OPT}$.

Claim 4.3 ((Buchbinder et al., 2014, Observation 3.2))**.** *For every $i = 0, \ldots, k$, we have $\mathbb{E}\left[f(S^* \cup S_i)\right] \geq (1 - 1/k)^i \cdot \mathrm{OPT}$.*

Proof. For every iteration $i = 1, \ldots k$, the subsampling step ensures that every element in $V \cup U$ is selected for inclusion in S_i with probability at most $1/k$. Hence, for every $i = 0, 1, \ldots, k$, each element is included in S_i with probability at most $1 - (1 - 1/k)^i$. Define $g : 2^V \to \mathbb{R}$ by $g(S) = g(S^* \cup S)$. Then g is a submodular function, and

$$\mathbb{E}\left[f(S^* \cup S_i)\right] = \mathbb{E}\left[g(S_i \setminus S^*)\right] \geq (1 - 1/k)^i g(\emptyset) = (1 - 1/k)^i \,\mathrm{OPT}\,.$$

The inequality here follows from the fact that for any set T and any random subset $T' \subseteq T$ that includes every element of T with probability p, we have $\mathbb{E}\left[g(T')\right] \geq (1 - p) \cdot g(\emptyset) + p \cdot g(T)$ for any submodular function g Feige et al. (2007). □

Claim 4.4.

$$\mathbb{E}[f_{S_{i-1}}(v_i)] \geq \left(1 - \frac{1}{e}\right) \cdot \left(\frac{\mathbb{E}[f(S^* \cup S_{i-1})] - \mathbb{E}[f(S_{i-1})]}{k}\right) - \alpha.$$

Proof. Begin by conditioning on a fixed choice of the set S_{i-1}. Let $M \subseteq (V \cup U)$ denote a set of k items which maximizes the quantity $\sum_{v \in M} f_{S_{i-1}}(v)$. That is, M consists of the k items in $(V \cup U)$ which result in the largest marginal gain for $f_{S_{i-1}}$.

Let G denote the event that the subsampled set $V_i \cup \{u_i\}$ contains at least one element in M. Observe that even if G does not occur, we have

$$\mathbb{E}\left[f_{S_{i-1}}(v_i) | \overline{G}\right] \geq f_{S_{i-1}}(u_i) - \alpha \geq -\alpha. \tag{3}$$

We claim moreover that

$$\mathbb{E}\left[f_{S_{i-1}}(v_i) | G\right] \geq \frac{1}{k} \sum_{v \in M} f_{S_{i-1}}(v) - \alpha. \tag{4}$$

To see this, sort the items in $V \cup U$ as $v^{(1)}, \ldots, v^{(m)}, v^{(m+1)}, \ldots, v^{(m+k)}$, where $m = |V|$ and $f_{S_{i-1}}(v^{(j)}) \geq f_{S_{i-1}}(v^{(j+1)})$ for every $j = 1, \ldots, m + k - 1$. Break ties in such a way that $M = \{v^{(1)}, \ldots, v^{(k)}\}$, and that there is some $t \in \{0, \ldots, k\}$ such that $v^{(1)}, \ldots, v^{(t)} \in V$ and $v^{(t+1)}, \ldots, v^{(k)} \in U$ (that is, real elements come before dummy elements).

Let A_j denote the event that j is the smallest index such that $v^{(j)} \in V_i \cup \{u_i\}$. Then the events $A_1, \ldots, A_{m+k}$ are mutually exclusive and exhaustive. Moreover, by the definition of G, we have $\sum_{j=1}^{k} \Pr[A_j] = \Pr[G]$.

It is easy to see that

$$\begin{aligned}
\Pr[A_1] &= \frac{1}{k} \\
\Pr[A_j] &= \frac{\binom{m-j}{m/k-1}}{\binom{m}{m/k}} && j = 2, \ldots, t, \\
\Pr[A_j] &= \frac{\binom{m-t}{m/k}}{\binom{m}{m/k}} \cdot \frac{1}{k} \cdot \left(1 - \frac{1}{k}\right)^{j-t-1} && j = t+1, \ldots, k.
\end{aligned}$$

Moreover, $\Pr[A_j]$ is a decreasing function $j = 1, \ldots, k$. Hence, $\Pr[A_j|G] = \Pr[A_j]/\Pr[G]$ is a decreasing function of $j = 1, \ldots, k$ as well. Moreover, $\Pr[A_1|G] \geq \Pr[A_1] = 1/k$. This allows us to calculate

$$\begin{aligned}
\mathbb{E}\left[f_{S_{i-1}}(v_i)|G\right] &= \sum_{j=1}^{k} \mathbb{E}\left[f_{S_{i-1}}(v_i)|A_j\right] \cdot \Pr[A_j|G] \\
&\geq \frac{1}{k} \sum_{j=1}^{k} \left(f_{S_{i-1}}(v^{(j)}) - \alpha\right) && \text{(by Chebyshev's sum inequality)} \\
&= \frac{1}{k} \sum_{v \in M} f_{S_{i-1}}(v) - \alpha.
\end{aligned}$$

This establishes the claimed inequality (4).

To estimate $\mathbb{E}\left[f_{S_{i-1}}(v_i)|G\right]$, it remains to calculate $\Pr[G]$. Suppose M consists of t elements from V and $k - t$ dummy elements from U. Then

$$\begin{aligned}
\Pr[G] &= 1 - \Pr[M \cap (V_i \cup \{u_i\}) = \emptyset] \\
&= 1 - \frac{\binom{m-t}{m/k}}{\binom{m}{m/k}} \cdot \frac{t}{k} \\
&= 1 - \frac{(m - (m/k))(m - (m/k) - 1) \ldots (m - (m/k) - t + 1)}{m(m-1)\ldots(m-t+1)} \cdot \frac{t}{k} \\
&= 1 - \left(1 - \frac{1}{k}\right)\left(1 - \frac{1}{k} \cdot \frac{m}{m-1}\right) \ldots \left(1 - \frac{1}{k} \cdot \frac{m}{m-t+1}\right) \cdot \frac{t}{k} \\
&\geq 1 - \left(1 - \frac{1}{k}\right)^{t} \cdot \frac{t}{k} \\
&\geq 1 - \frac{te^{-t/k}}{k} \\
&\geq 1 - \frac{1}{e},
\end{aligned}$$

where the last inequality follows from the fact that the function $r(x) = xe^{-x}$ is maximized at $x = 1$, where it takes the value $1/e$.

Let M' be the set containing $S^* \setminus S_{i-1}$ together with enough dummy elements to have size exactly k. We conclude that

$$\begin{aligned}
\mathbb{E}\left[f_{S_{i-1}}(v_i)\right] &= \Pr[G] \cdot \mathbb{E}\left[f_{S_{i-1}}(v_i)|G\right] + (1 - \Pr[G]) \cdot \mathbb{E}\left[f_{S_{i-1}}(v_i)|\bar{G}\right] \\
&\geq \left(1 - \frac{1}{e}\right)\left(\frac{1}{k}\sum_{v \in M} f_{S_{i-1}}(v) - \alpha\right) - \frac{1}{e} \cdot \alpha && \text{(by (4) and (3))} \\
&\geq \left(1 - \frac{1}{e}\right)\left(\frac{1}{k}\sum_{v \in M'} f_{S_{i-1}}(v)\right) - \alpha && \text{(by definition of } M\text{)} \\
&\geq \left(1 - \frac{1}{e}\right)\left(\frac{f(S^* \cup S_{i-1}) - f(S_{i-1})}{k}\right) - \alpha. && \text{(by submodularity)}
\end{aligned}$$

Unconditioning from S_{i-1} by taking the expectation over its choice proves the claim. □

Proof of Theorem 4.1. Let f be any (possibly non-monotone) submodular function. We show by induction that for every $i = 0, \ldots, k$, we have

$$\mathbb{E}\left[f(S_i)\right] \geq \left(1 - \frac{1}{e}\right) \cdot \frac{i}{k} \cdot \left(1 - \frac{1}{k}\right)^{i-1} \cdot \mathrm{OPT} - i\alpha. \tag{5}$$

This clearly holds for the base case of $i = 0$. Assuming it holds in iteration $i - 1$, we calculate

$$\begin{aligned}
\mathbb{E}[f(S_i)] &= \mathbb{E}[f(S_{i-1})] + \mathbb{E}[f_{v_i}(S_{i-1})] \\
&\geq \mathbb{E}[f(S_{i-1})] + \left(1 - \frac{1}{e}\right)\left(\frac{\mathbb{E}[f(S^* \cup S_{i-1})] - \mathbb{E}[f(S_{i-1})]}{k}\right) - \alpha && \text{(by Claim 4.4)} \\
&\geq \mathbb{E}[f(S_{i-1})] + \left(1 - \frac{1}{e}\right)\left(\frac{(1 - \frac{1}{k})^{i-1}\,\mathrm{OPT} - \mathbb{E}[f(S_{i-1})]}{k}\right) - \alpha && \text{(by Claim 4.3)} \\
&= \left(1 - \frac{1}{k}\right)\mathbb{E}[f(S_{i-1})] + \left(1 - \frac{1}{e}\right) \cdot \left(1 - \frac{1}{k}\right)^{i-1} \cdot \frac{1}{k} \cdot \mathrm{OPT} - \alpha \\
&\geq \left(1 - \frac{1}{k}\right)\left(1 - \frac{1}{e}\right) \cdot \frac{i-1}{k} \cdot \left(1 - \frac{1}{k}\right)^{i-2} \cdot \mathrm{OPT} + \left(1 - \frac{1}{e}\right) \cdot \left(1 - \frac{1}{k}\right)^{i-1} \cdot \frac{1}{k} \cdot \mathrm{OPT} - i\alpha \\
&\qquad \text{(by the inductive hypothesis)} \\
&= \left(1 - \frac{1}{e}\right) \cdot \frac{i}{k} \cdot \left(1 - \frac{1}{k}\right)^{i-1} \cdot \mathrm{OPT} - i\alpha.
\end{aligned}$$

Hence, in iteration k, we have

$$\mathbb{E}\left[f(S_k)\right] \geq \left(1 - \frac{1}{e}\right) \cdot \left(1 - \frac{1}{k}\right)^{k-1} \cdot \mathrm{OPT} - k\alpha \geq \left(1 - \frac{1}{e}\right) \cdot \frac{1}{e} \cdot \mathrm{OPT} - k\alpha$$

as we wanted to show.

Now we consider the special case where f is monotone. In this case, we have

$$\begin{aligned}
\mathbb{E}[f_{v_i}(S_{i-1})] &\geq \left(1 - \frac{1}{e}\right)\left(\frac{\mathbb{E}[f(S^* \cup S_{i-1})] - \mathbb{E}[f(S_{i-1})]}{k}\right) - \alpha && \text{(by Claim 4.4)} \\
&\geq \left(1 - \frac{1}{e}\right)\left(\frac{\mathrm{OPT} - \mathbb{E}[f(S_{i-1})]}{k}\right) - \alpha && \text{(by monotonicity).}
\end{aligned}$$

Rearranging gives us

$$\mathrm{OPT} - \mathbb{E}\left[f(S_i)\right] \leq \left(1 - \frac{(1-1/e)}{k}\right) \cdot \left(\mathrm{OPT} - \mathbb{E}\left[f(S_{i-1})\right]\right) + \alpha.$$

Recursively applying this bound yields

$$\begin{aligned}\mathrm{OPT} - \mathbb{E}[f(S_i)] &\leq \left(1 - \frac{(1-1/e)}{k}\right)^i \left(\mathrm{OPT} - \mathbb{E}[f(S_0)]\right) + \sum_{j=0}^{i-1} \left(1 - \frac{1}{k}\right)^j \alpha \\ &\leq \left(1 - \frac{(1-1/e)}{k}\right)^i \mathrm{OPT} + i\alpha.\end{aligned}$$

Hence, we conclude

$$\begin{aligned}\mathbb{E}[f(S_k)] &\geq \left[1 - \left(1 - \frac{(1-1/e)}{k}\right)^k\right] \mathrm{OPT} - k\alpha \\ &\geq \left(1 - e^{-(1-1/e)}\right) \mathrm{OPT} - k\alpha.\end{aligned}$$

□

5 The Large Margin Mechanism

The accuracy guarantee of the exponential mechanism can be pessimistic on datasets where $q(\cdot, D)$ exhibits additional structure. For example, suppose that when the elements of V are sorted so that $q(v_1, D) \geq q(v_2, D) \geq \cdots \geq q(v_{|V|}, D)$, there exists an ℓ such that $q(v_1, D) \gg q(v_{\ell+1}, D)$. Then only the top ℓ ground set items are relevant to the optimization problem, so running the exponential mechanism on these should maintain differential privacy, but with error proportional to $\ln \ell$ rather than to $\ln |V|$. The *large margin mechanism* of Chaudhuri et al. (2014), like the exponential mechanism, generically solves discrete optimization problems. However, it automatically leverages this additional margin structure whenever it exists. Asymptotically, the error guarantee of the large margin mechanism is always at most that of the exponential mechanism, but can be much smaller when the data exhibits a margin for small ℓ.

Formally, given a quality function $q : V \times X^n \to \mathbb{R}$ and parameters $\ell \in \mathbb{N}, \gamma > 0$, a dataset D satisfies the (ℓ, γ)-margin condition if $q(v_{\ell+1}, D) < q(v_1, D) - \gamma$.

For each $\ell = 1, \ldots, |V|$, define

$$\begin{aligned}g_\ell &= \lambda \cdot \left(3 + \frac{4\ln(2\ell/\delta)}{\varepsilon}\right) \\ G_\ell &= \frac{8\lambda \ln(2/\delta)}{\varepsilon} + \frac{16\lambda \ln(7\ell^2/\delta)}{\varepsilon} + g_\ell.\end{aligned}$$

Recall that the Laplace distribution $\mathrm{Lap}(b)$ is specified by the density function $\frac{1}{2b}\exp(-|x|/b)$. , and a sample $Z \sim \mathrm{Lap}(b)$ obeys the tail bound $\Pr[Z > t] = \frac{1}{2}\exp(-t/b)$ for all $t > 0$.

Proposition 5.1. *Let $\varepsilon, \delta > 0$. Consider the large margin mechanism described in Algorithm 5. Then*

- *Algorithm* LMM *is (ε, δ)-differentially private.*

Algorithm 5 Large Margin Mechanism (LMM)

Input: Quality function $q : V \times X^n \to \mathbb{R}$, dataset D, privacy parameters $\varepsilon, \delta > 0$
Output: Item $\hat{v} \in V$

1. Sort the elements of V so that $q(v_1, D) \geq \cdots \geq q(v_{|V|}, D)$
2. Let $m = q(v_1, D) + Z$ for $Z \sim \mathrm{Lap}(8\lambda/\varepsilon)$
3. For $\ell = 1, \ldots, |V|$:
   - Sample $Z_\ell \sim \mathrm{Lap}(16\lambda/\varepsilon)$
   - If $m - q(v_{\ell+1}, D) > G_\ell + Z_\ell$: Report ℓ and break
4. Return $\hat{v} \in \{v_1, \ldots, v_\ell\}$ sampled w.p. $\propto \exp(\varepsilon q(v_i, D)/4\lambda)$.

- *Suppose $D \in X^n$ satisfies the (ℓ, γ)-margin condition for*

$$\gamma = \frac{24\lambda \ln(1/\beta)}{\varepsilon} + G_\ell$$

for some $\beta > 0$. Then there exists an event E with $\Pr[E] \geq 1 - \beta$ such that

$$\mathbb{E}[q(\hat{v}, D)|E] \geq \mathrm{OPT} - \frac{4\lambda \cdot \ln \ell}{\varepsilon},$$

where $\hat{v}$ is the output of $\mathrm{LMM}(D)$.

Our presentation of Algorithm 5 differs slightly from that of Chaudhuri et al. (2014). Namely, we simplify the choice of the noisy maximum m, and redistribute the algorithm's use of the privacy budget ε with an eye toward better performance in applications. Because of these small changes, we sketch the proof of Proposition 5.1 for completeness.

Privacy Analysis of Proposition 5.1. Algorithm 5 can be thought of as releasing two items in stages: First, the margin parameter ℓ in Step 3, and second, the item $\hat{v}$ sampled via the exponential mechanism in Step 4. We first claim that releasing the margin parameter ℓ guarantees $(\varepsilon/2, 0)$-differential privacy. This follows because Steps 2 and 3 taken together are an instantiation of the "AboveThreshold" algorithm, as presented by Dwork and Roth (Dwork & Roth, 2014, Theorem 3.23), with respect to the sensitivity-(2λ) functions $q(v_1, D) - q(v_{\ell+1}, D)$. Denote the output ℓ of the algorithm at Step 3 by $S(D)$.

We now establish that Step 4 provides differential privacy. Following Chaudhuri et al. (2014), we let $A(\ell, D)$ capture the behavior of the algorithm in Step 4, where on receiving ℓ from Step 3, it samples from the exponential mechanism on the top ℓ elements. They proved the following lemma about $A(\ell, D)$:

Lemma 5.2 ((Chaudhuri et al., 2014, Lemma 5))**.** *If D satisfies the (ℓ, γ)-margin condition with*

$$\gamma \geq 2\lambda \left(1 + \frac{2\ln(\ell/\delta')}{\varepsilon}\right)$$

for some $\delta' > 0$, then for every neighbor $D' \sim D$ and any $T \subseteq V$, we have

$$\Pr[A(\ell, D) \in T] \leq e^{\varepsilon/2} \Pr[A(\ell, D') \in T] + \delta'.$$

Now fix neighboring datasets $D \sim D'$. Let $\mathcal{L}$ denote the set of ℓ for which $q(v_1, D) - q(v_{\ell+1}, D) \geq g_\ell$. By definition, if $\ell = S(D) \in \mathcal{L}$, then D indeed satisfies the (ℓ, g_ℓ)-margin condition. Moreover, by tail bounds on the Laplace distribution,

$$\begin{aligned}
\Pr[S(D) \notin \mathcal{L}] &\leq \Pr\left[Z > 8\lambda \ln(2/\delta)/\varepsilon \vee \left(\exists \ell \in \{1, \ldots, |V|\} : Z_\ell < -16\lambda \ln(7\ell^2/\delta)/\varepsilon\right)\right] \\
&\leq \frac{\delta}{4} + \sum_{\ell=1}^{|V|} \frac{6\delta}{4\pi^2 \ell^2} \\
&\leq \frac{\delta}{2}.
\end{aligned}$$

Hence, we have that for any $T \subseteq V$,

$$\begin{aligned}
\Pr[\mathrm{LMM}(D) \in T] &\leq \sum_{\ell \in \mathcal{L}} \Pr[\mathrm{LMM}(D) \in T | S(D) = \ell] \cdot \Pr[S(D) = \ell] + \Pr[S(D) \notin \mathcal{L}] \\
&\leq \sum_{\ell \in \mathcal{L}} \Pr[\mathrm{LMM}(D) \in T | S(D) = \ell] \cdot e^{\varepsilon/2} \Pr[S(D') = \ell] + \frac{\delta}{2} \\
&\leq \sum_{\ell \in \mathcal{L}} (e^{\varepsilon/2} \Pr[\mathrm{LMM}(D') \in T | S(D') = \ell] + e^{-\varepsilon/2}\frac{\delta}{2}) \cdot e^{\varepsilon/2} \Pr[S(D') = \ell] + \frac{\delta}{2} \quad \text{by Lemma 5.2} \\
&\leq e^{\varepsilon} \Pr[\mathrm{LMM}(D') \in T] + \delta
\end{aligned}$$

This completes the privacy proof of Proposition 5.1.

□

Utility Analysis of Proposition 5.1. Suppose D satisfies the (ℓ, β)-margin condition with

$$\gamma \geq \frac{24\lambda \ln(1/\beta)}{\varepsilon} + G_\ell,$$

for some $\beta > 0$. By the tail bound for the Laplace distribution and a union bound, we have that with probability at least $1 - \beta$,

$$Z \geq \frac{8}{\lambda} \ln \frac{1}{\beta} \quad \text{and} \quad Z_\ell \leq \frac{16}{\lambda} \ln \frac{1}{\beta}.$$

Let E be the event where this occurs. If E occurs, then indeed we have

$$(q(v_1, D) + Z) - q(v_{\ell+1}, D) > G_\ell + Z_\ell,$$

and hence Step 3 terminates outputting some $\ell' \leq \ell$. By Proposition 2.5, it follows that

$$\mathbb{E}[q(\hat{v}, D) | E] \geq \mathrm{OPT} - \frac{4\lambda \cdot \ln \ell}{\varepsilon}.$$

□

Replacing the exponential mechanism with the large margin mechanism gives analogues of our results for monotone submodular maximization with a cardinality constraint, monotone submodular maximization over a p-extendible system, and non-monotone submodular maximization with a cardinality constraint:

Theorem 5.3. *Suppose $f_D : 2^V \to \mathbb{R}$ is monotone and has sensitivity λ. Then instantiating Algorithm 1 with $\mathcal{O} = \mathrm{LMM}$ and parameters $\varepsilon_0, \delta_0 = 0$ provides $(k\varepsilon_0, k\delta_0)$-differential privacy. It also provides $(\varepsilon, \delta' + k\delta_0)$-differential privacy for every $\delta' > 0$ with $\varepsilon = k\varepsilon^2/2 + \varepsilon \cdot \sqrt{2k \ln(1/\delta')}$.*

Moreover, for every $D \in X^n$, there exists an event E with $\Pr[E] \geq 1 - \beta$ such that

$$\mathbb{E}\left[f_D(S_k)|E\right] \geq \left(1 - \frac{1}{e}\right) \mathrm{OPT} - \sum_{i=1}^{k} \frac{4\lambda \ln \ell_i}{\varepsilon_0}$$

where $S_k \leftarrow_R \mathcal{G}^{\mathrm{LMM}}(D)$, and D satisfies the (ℓ_i, γ_i)-margin condition with respect to every function of the form $q_i(v, D) = f_D(\hat{S}_{i-1} \cup \{v\}) - f_D(\hat{S}_{i-1})$, with $\gamma_i = 24\lambda \ln(k/\beta)/\varepsilon + G_{\ell_i}$.

Theorem 5.4. *Instantiating Algorithm 2 with $\mathcal{O} = \mathrm{LMM}$ under all of the conditions of Theorem 5.3 gives the same privacy guarantee (replacing k with $r(\mathcal{I})$) and gives*

$$\mathbb{E}\left[f_D(S)|E\right] \geq \frac{1}{p+1} \cdot \mathrm{OPT} - \sum_{i=1}^{r(\mathcal{I})} \frac{4\lambda \ln \ell_i}{\varepsilon_0}.$$

Theorem 5.5. *Instantiating Algorithm 4 with $\mathcal{O} = \mathrm{LMM}$ under all of the conditions of Theorem 5.3 gives the same privacy guarantee and gives*

$$\mathbb{E}\left[f_D(S_k)|E\right] \geq \frac{1}{e}\left(1 - \frac{1}{e}\right) \mathrm{OPT} - \sum_{i=1}^{k} \frac{4\lambda \ln \ell_i}{\varepsilon_0}.$$

Moreover, if f_D is monotone, then

$$\mathbb{E}\left[f_D(S_k)|E\right] \geq 0.468\,\mathrm{OPT} - \sum_{i=1}^{k} \frac{4\lambda \ln \ell_i}{\varepsilon_0}.$$

6 Experimental Results

In this section we describe two concrete applications of our mechanisms.

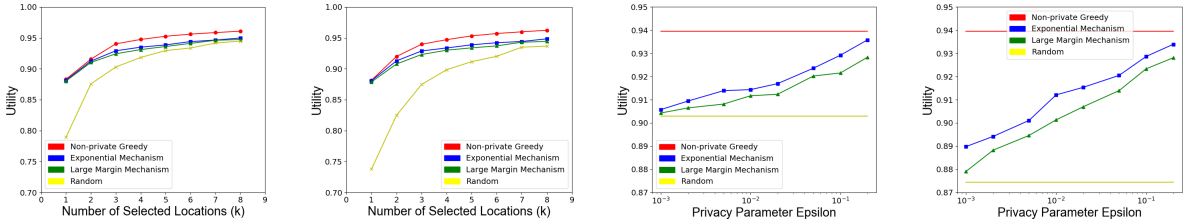
6.1 Location Privacy

We analyze a dataset of 10,000 Uber pickups in Manhattan in April 2014 (UberDataset). Each individual entry in the dataset consists of the longitude and latitude coordinates of the pickup location. We want to use this dataset to select k public locations as waiting spots for idle Uber drivers, while also guaranteeing differential privacy for the passengers whose locations appear in this dataset.[1] We consider two different public sets of locations L:

- $L_{Popular}$ is a set of 33 popular locations in Manhattan.
- L_{Grid} is a set of 33 locations spread evenly across Manhattan in a grid-like manner.

We define a utility function $M(i, j)$ to be the normalized Manhattan distance between a pickup location i and the waiting location j. That is, if pickup location i is located at coordinates (i_1, i_2) and the waiting location j is located at coordinates (j_1, j_2), then $M(i, j) = \dfrac{|i_1 - j_1| + |i_2 - j_2|}{m}$, where $m = 0.266$ is simply the Manhattan distance between the two furthest spread apart points in Manhattan. This normalization

[1] Under the assumption that each pickup corresponds to a unique individual.



(a) $L_{Popular}$ (b) L_{Grid} (c) $L_{Popular}$ (d) L_{Grid}

(e) $L_{Popular}$: Non-private greedy (f) $L_{Popular}$: EM-based greedy (g) L_{Grid}: Non-private greedy (h) L_{Grid}: EM-based greedy

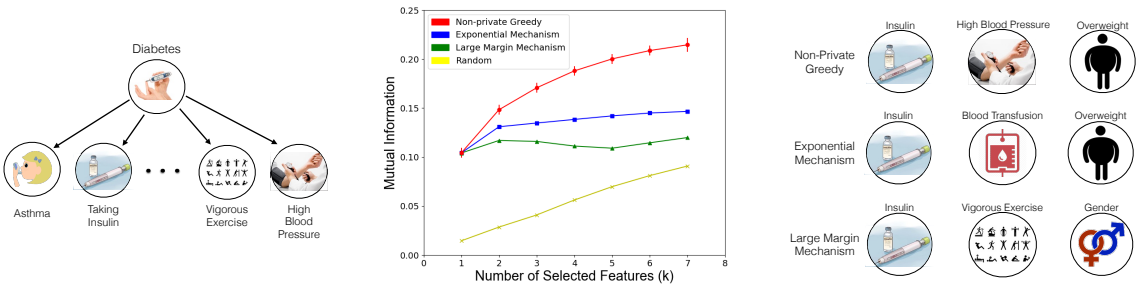
Figure 1: (a) and (b) show utility for various cardinalities (k). (c) and (d) fix $k = 3$ and show utility for various privacy parameters (ϵ). Utility is normalized to be between 0 and 1. (e) - (h) shows a representative top 3 set under various settings.

ensures that $0 \leq M(i,j) \leq 1$, for all i, j. In order to make sure we have a maximization problem, we define the following objective function: $f_D(S) = n - \sum_{i \in D} \min_{j \in S} M(i,j)$, where $n = |D| = 10000$.

Observation 6.1. *The function f_D is λ-decomposable for $\lambda = 1$ (and hence has sensitivity 1).*

This form of objective function is known to be monotone submodular and so we can use the greedy algorithms studied in this paper. We use $\epsilon = 0.1$ and $\delta = 2^{-20}$. For our settings of parameters, "basic composition" outperforms "advanced composition," so the privacy budget of $\epsilon = 0.1$ is split equally across the k iterations, meaning the mechanism at each iteration uses $\epsilon_0 = \frac{\epsilon}{k}$. Our figures plot the average utility across 100 simulations.

From Figures 1(a) and (b) we see that the results for both $L_{Popular}$ and L_{Grid} are relatively similar and unsurprising. The non-private greedy algorithm achieves the highest utility, but both the exponential mechanism (EM)-based greedy and large margin mechanism (LMM)-based greedy algorithms exhibit comparable utility while preserving a high level of privacy. Interestingly, we also see that the utilities of the EM-based and LMM-based algorithms are almost identical for both $L_{Popular}$ and L_{Grid}. This indicates that our mechanisms are actually selecting good locations, rather than just getting lucky because there are a lot of good locations to choose from.



(a) Graphical model of Naive Bayes (b) Expected mutual information (c) Representative top 3 features

Figure 2: Privately selecting health features, from national health examination surveys, that correlate most with diabetes.

Figures 1(c) and (d) show how the utility of the EM-based and LMM-based algorithms vary with the privacy parameter ϵ. We can also think of this as varying the dataset size for a fixed ϵ. We fix $k = 3$ and take the average of 100 simulations for each value of ϵ. We see that even for very small ϵ, our algorithms outperform fully random selection. As ϵ increases, so does the utility. It is not shown in this figure, but varying δ has very little effect.

From Figures 1(e) - (h), we see that the both the non-private and private algorithms select public locations that are relatively close to each other. For example, for the $L_{Popular}$ set of locations, the Empire State Building is close to the New York Public Library, the Soho Grand Hotel is close to NYU, and the Grand Army Plaza is close to the UN Headquarters. As a result, the private mechanisms manage to achieve comparable utility, while also masking the users' exact locations.

The theory described in Section 5 suggests that, at least asymptotically, the large margin mechanism-based algorithm should outperform the exponential mechanism-based algorithm. However, in our experiments, we find that the large margin mechanism is generally only able to find a margin in the first iteration of the greedy algorithm. This is because the threshold for finding a margin depends only on ϵ, δ, and n and

thus it stays the same across all k iterations. On the other hand, the marginal gain at each iteration drops very quickly, so the mechanism fails to find a margin and thus samples from all remaining locations. However, since the large margin mechanism spends half of its privacy budget to try to find a margin, the sampling step gives slightly worse guarantees than does the plain exponential mechanism, thus giving us the slightly weaker results we see in the figures.

6.2 Feature Selection Privacy

We analyze a dataset created from a combination of National Health Examination Surveys ranging from 2007 to 2014 (NHANESDataset). There are $n = 23,876$ individuals in the dataset with information on whether or not they have diabetes, along with $m = 23$ other potentially related binary health features. Our goal is to privately select k of these features that provide as much information about the diabetes class variable as possible.

More specifically, our goal is to maximize the mutual information between Y and X_S, where Y is a binary random variable indicating whether or not an individual has diabetes and X_S is a random variable that represents a set S of k binary health features. Mutual information takes the form:

$$I(Y;X) = \sum_{y \in \mathcal{Y}} \sum_{x \in \mathcal{X}} p(x,y) \log_2 \left(\frac{p(x,y)}{p(x)p(y)} \right).$$

Under the Naive Bayes assumption, we suppose the joint distribution on $(Y, X_1, \ldots, X_k)$ takes the form $p(y, x_1, \ldots, x_k) = p(y) \prod_{i=1}^{k} p(x_i \mid y)$. Therefore, we can easily specify the entire probability distribution by finding each $p(x_i \mid y)$. We estimate each $p(x_i \mid y)$ by counting frequencies in the dataset.

Our goal is to choose a size k subset S of the features in order to maximize $f_D(S) = I(Y; X_S)$. Mutual information (under the Naive Bayes assumption) for feature selection is known to be monotone submodular in S (Krause & Guestrin, 2005), and thus we can apply the greedy algorithms described in this paper.

Claim 6.2. *In iteration i of the greedy algorithm, the sensitivity of $f_D(S)$ is $\frac{(2i+1)\log_2(n)}{n}$.*

We run 1,000 simulations with $\epsilon = 1.0$ and $\delta = 2^{-20}$. As we can see from Figure 2(b), our private mechanisms maintain a comparable utility relative to the non-private algorithm. We also observe an interesting phenomenon where the expected utility obtained by our mechanism is not necessarily monotonically increasing with the number of features selected. This is an artifact of the fact that if we are selecting k features, then composition requires us to divide ϵ so that each iteration uses privacy budget $\frac{\epsilon}{k}$. This is problematic for this particular application because there happens to be one feature (insulin administration) that has much higher value than the rest. Therefore, the reduced probability of picking this single best feature (as a result of the lower privacy parameter $\frac{\epsilon}{k}$) is not compensated for by selecting more features.

From Figure 2(c), we see that both the private and non-private mechanisms generally select insulin administration as the top feature. However, while all three of the top features selected by the non-private algorithm are clearly related to diabetes, the non-private mechanisms tend to select one feature (in our case, gender or having received a blood transfusion) that may not be quite as relevant.

7 Conclusion

We have presented a general framework for maximizing submodular functions while guaranteeing differential privacy. Our results demonstrate that simple and flexible greedy algorithms can preserve privacy while

achieving competitive guarantees for a variety of submodular maximization problems: for all functions under cardinality constraints, as well as for monotone functions under matroid and p-extendible system constraints. Via our motivation to identify algorithms that could be made differentially private, we discovered a non-monotone submodular maximization algorithm that achieves guarantees that are novel even without concern for privacy. Finally, our experiments show that our algorithms are indeed competitive with their non-private counterparts.

Acknowledgments. This work was supported by DARPA Young Faculty Award (D16AP00046), Simons-Berkeley fellowship, and ERC StG 307036. This work was done in part while Amin Karbasi and Andreas Krause were visiting the Simons Institute for the Theory of Computing.

References

Bassily, Raef, Smith, Adam D., and Thakurta, Abhradeep. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *FOCS*, pp. 464–473, 2014.

Bassily, Raef, Nissim, Kobbi, Smith, Adam D., Steinke, Thomas, Stemmer, Uri, and Ullman, Jonathan. Algorithmic stability for adaptive data analysis. In *STOC*, pp. 1046–1059, 2016.

Beimel, Amos, Nissim, Kobbi, and Stemmer, Uri. Private learning and sanitization: Pure vs. approximate differential privacy. *Theory of Computing*, 12(1):1–61, 2016.

Buchbinder, Niv, Feldman, Moran, Naor, Joseph, and Schwartz, Roy. Submodular maximization with cardinality constraints. In *SODA*, pp. 1433–1452, 2014.

Bun, Mark and Steinke, Thomas. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *TCC*, pp. 635–658, 2016.

Bun, Mark, Nissim, Kobbi, Stemmer, Uri, and Vadhan, Salil P. Differentially private release and learning of threshold functions. In *FOCS*, pp. 634–649, 2015.

Călinescu, Gruia, Chekuri, Chandra, Pál, Martin, and Vondrák, Jan. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 2011.

Chaudhuri, Kamalika, Hsu, Daniel J., and Song, Shuang. The large margin mechanism for differentially private maximization. In *NIPS*, pp. 1287–1295, 2014.

Cheraghchi, Mahdi, et al. Submodular functions are noise stable. In *SODA*, 2012.

Dwork, Cynthia and Lei, Jing. Differential privacy and robust statistics. In *STOC*, pp. 371–380, 2009.

Dwork, Cynthia and Roth, Aaron. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

Dwork, Cynthia, McSherry, Frank, Nissim, Kobbi, and Smith, Adam D. Calibrating noise to sensitivity in private data analysis. In *TCC*, pp. 265–284, 2006.

Dwork, Cynthia, Rothblum, Guy N., and Vadhan, Salil P. Boosting and differential privacy. In *FOCS*, pp. 51–60, 2010.

Feige, U., Mirrokni, V., and Vondrak, J. Maximizing non-monotone submodular functions. In *FOCS*, 2007.

Feldman, Moran, Naor, Joseph, and Schwartz, Roy. A unified continuous greedy algorithm for submodular maximization. In *FOCS*, 2011.

Feldman, Moran, Harshaw, Christopher, and Karbasi, Amin. Greed is good: Near-optimal submodular maximization via greedy optimization. In *COLT*, 2017.

Fisher, Marshall L., Nemhauser, George L., and Wolsey, Laurence A. An analysis of approximations for maximizing submodular set functions - II. *Mathematical Programming Study*, (8), 1978.

Gupta, Anupam, Ligett, Katrina, McSherry, Frank, Roth, Aaron, and Talwar, Kunal. Differentially private combinatorial optimization. In *SODA*, pp. 1106–1125, 2010.

Hassidim, Avinatan and Singer, Yaron. Submodular optimization under noise. *CoRR*, abs/1601.03095, 2016. URL `http://arxiv.org/abs/1601.03095`.

Jenkyns, T. A. The efficacy of the "greedy" algorithm. In *South Eastern Conference on Combinatorics, Graph Theory and Computing*, 1976.

Kempe, David, Kleinberg, Jon, and Tardos, Éva. Maximizing the spread of influence through a social network. In *KDD*, 2003.

Kirchhoff, Katrin and Bilmes, Jeff. Submodularity for data selection in statistical machine translation. In *EMNLP*, 2014.

Krause, A. and Guestrin, C. Near-optimal nonmyopic value of information in graphical models. In *UAI*, 2005.

Krause, Andreas and Gomes, Ryan G. Budgeted nonparametric learning from data streams. In *ICML*, 2010.

Lin, Hui and Bilmes, Jeff. A class of submodular functions for document summarization. In *ACL*, 2011.

McSherry, Frank and Talwar, Kunal. Mechanism design via differential privacy. In *FOCS*, pp. 94–103, 2007.

Mestre, Julián. Greedy in approximation algorithms. In *ESA*, pp. 528–539, 2006.

Mirzasoleiman, Baharan, Badanidiyuru, Ashwinkumar, Karbasi, Amin, Vondrak, Jan, and Krause, Andreas. Lazier than lazy greedy. In *AAAI*, 2015.

Mirzasoleiman, Baharan, Badanidiyuru, Ashwinkumar, and Karbasi, Amin. Fast constrained submodular maximization: Personalized data summarization. In *ICML*, 2016a.

Mirzasoleiman, Baharan, Zadimoghaddam, Morteza, and Karbasi, Amin. Fast distributed submodular cover: Public-private data summarization. In *NIPS*, 2016b.

Nemhauser, George L., Wolsey, Laurence A., and Fisher, Marshall L. An analysis of approximations for maximizing submodular set functions - I. *Mathematical Programming*, 1978.

NHANESDataset. National health and nutrition examination survey (2007 - 2014). URL `https://wwwn.cdc.gov/nchs/nhanes/default.aspx`.

Papadimitriou, Christos H., Schapira, Michael, and Singer, Yaron. On the hardness of being truthful. In *FOCS*, pp. 250–259, 2008.

Singla, Adish, Bogunovic, Ilija, Bartók, Gábor, Karbasi, Amin, and Krause, Andreas. Near-optimally teaching the crowd to classify. In *ICML*, 2014.

Sipos, Ruben, Swaminathan, Adith, Shivaswamy, Pannaga, and Joachims, Thorsten. Temporal corpus summarization using submodular word coverage. In *CIKM*, 2012.

Song, Shuang, Chaudhuri, Kamalika, and Sarwate, Anand D. Stochastic gradient descent with differentially private updates. In *GlobalSIP*, pp. 245–248, 2013.

UberDataset. Uber pickups in new york city. URL `https://www.kaggle.com/fivethirtyeight/uber-pickups-in-new-york-city`.