Lecturer: Dabeen Lee

Outline

In this lecture, we consider the problem of maximizing a submodular set function. We first analyze the greedy algorithm for the cardinality constraint case. Then we introduce the continuous relaxation-based approach for the matroid constraint case. Lastly, we cover the first-order method for maximizing a continuous DR-submodular function.

1 Submodular function maximization

In this section, we consider the problem of maximizing a submodular function. Let $f : 2^{|E|} \to \mathbb{R}$ be a submodular set function. We focus on the case where f is **monotone**, meaning that $f(S) \leq f(T)$ if $S \subseteq T$. Then the problem is given by

$$\text{maximize} \quad f(S) \quad \text{subject to} \quad S \in \mathcal{F} \tag{10.1}$$

where $\mathcal{F} \subseteq 2^{|E|}$ is the collection of feasible subsets. What follows is a list of examples for $\mathcal{F}$.

- Cardinality constraint: $\mathcal{F} = \{S \subseteq E : \ |S| \leq k\}$.
- Knapsack constraint: $\mathcal{F} = \left\{S \subseteq E : \ \sum_{i \in S} c_i \leq B\right\}$.
- Matroid: $\mathcal{F} = \{S \subseteq E : \ S \text{ is an independent set of matroid } \mathcal{M}\}$.

There are a wide range of applications in business operations, such as online freelancing platforms [SVY20], team selection - sports teams, online gaming [KR15], assortment selection in online shopping websites [Udw21], and influence maximization [KKT03]. Recently, submodular function maximization is applied to many machine learning problems, including feature and variable selection [KG05], dictionary learning [DK11], document summarization [LB10, LB11], image summarization [TIWB14, MJK18], and active set selection in non-parametric learning [MKSK16].

Submodular function maximization (SFM) is NP-hard even with a monotone objective subject to a cardinality constraint [CFN77]. However, there exist polynomial time constant approximation algorithms. We say that a solution $\bar{S} \in \mathcal{F}$ is α-approximate for some $\alpha \in [0, 1]$ if

$$f(\bar{S}) \geq \alpha \cdot \max_{S \in \mathcal{F}} f(S).$$

An α-approximation algorithm would always find an α'-approximate solution for some $\alpha' \geq \alpha$ for every instance of SFM. In other words, the worst-case guarantee is always as good as α times the optimal value. A constant approximation algorithm is an α-approximation algorithm for some fixed $\alpha > 0$.

Let us explain a simple greedy algorithm by Nemhauser, Wolsey, and Fisher [NWF78] that guarantees an $(1 - 1/e)$-approximate solution. The idea is that until we reach the size limit, we take an element that achieves the maximum marginal return value.

Algorithm 1 Greedy algorithm for submodular maximization

Initialize $S = \emptyset$
while $|S| < k$ **do**
  Take an element $e \in \arg\max \{f(S \cup \{e\}) - f(S) : \ e \in E \setminus S\}$
end while
Return S

Theorem 10.1 (Nemhauser, Wolsey, and Fisher [NWF78]). *Let $\bar{S} \subseteq E$ be the outcome of Algorithm 1. Assume that $f(\emptyset) = 0$. Then*

$$f(\bar{S}) \geq \left(1 - \frac{1}{e}\right) \max \{f(S) : |S| \leq k\}.$$

Proof. Let S^* denote an optimal solution to SFM. Suppose that $e_1, \ldots, e_k$ is the sequence of elements selected by the algorithm. For $i \in \{1, \ldots, k\}$, we use notation $S_i = \{e_1, \ldots, e_i\}$. Then $\bar{S} = S_k$.

Let us prove by induction that

$$f(S^*) - f(S_i) \leq \left(1 - \frac{1}{k}\right)^i f(S^*)$$

for all $i \in \{0, 1, \ldots, k\}$. The claim trivially holds when $i = 0$. Suppose that $f(S^*) - f(S_{i-1}) \leq (1 - 1/k)^{i-1} f(S^*)$ for some $i \in \{1, \ldots, k\}$. Since f is submodular, we have

$$f(S^*) - f(S_{i-1}) \leq \sum_{e \in S^* \setminus S_{i-1}} \left(f(S_{i-1} \cup \{e\}) - f(S_{i-1})\right).$$

Since e_i has the maximum marginal return after S_{i-1}, it follows that

$$\begin{aligned}\sum_{e \in S^* \setminus S_{i-1}} \left(f(S_{i-1} \cup \{e\}) - f(S_{i-1})\right) &\leq |S^* \setminus S_{i-1}| \left(f(S_i) - f(S_{i-1})\right) \\ &\leq k \left(f(S_i) - f(S_{i-1})\right).\end{aligned}$$

Then it follows that

$$\begin{aligned}f(S^*) - f(S_i) &= f(S^*) - f(S_{i-1}) - \left(f(S_i) - f(S_{i-1})\right) \\ &\leq \left(1 - \frac{1}{k}\right) \left(f(S^*) - f(S_{i-1})\right) \\ &\leq \left(1 - \frac{1}{k}\right)^i f(S^*)\end{aligned}$$

where the second inequality is due to the induction hypothesis.

As a result, we have

$$f(\bar{S}) = f(S_k) \geq \left(1 - \left(1 - \frac{1}{k}\right)^k\right) f(S^*) \geq \left(1 - \frac{1}{e}\right) f(S^*),$$

as required. □

In fact, the greedy algorithm can be extended to an $(1 - 1/e)$-approximation algorithm for the knapsack constraint setting [Svi04].

2 Matroid constraint

In this section, we explain an $(1-1/e)$-approximation algorithm for the matroid constraint setting [CCPV07, Von08]. The outline of the algorithm works as follows.

1. First, obtain a continuous relaxation of the given submodular set function (multilinear relaxation).
2. Next, take a continuous relaxation of the feasible region (polymatroid).
3. Then solve the resulting continuous relaxation of the submodular function maximization problem (continuous greedy algorithm).
4. Lastly, round the fractional solution to obtain a discrete solution (pipage rounding).

Let us explain one by one. Given a submodular set function $f:\{0,1\}^{|E|}\to\mathbb{R}$, we take the **multilinear extension** of f given by

$$F(x)=\sum_{S\subseteq E} f(S)\prod_{e\in S} x_e\prod_{e\notin S}(1-x_e)\quad \text{for } x\in[0,1]^{|E|}.$$

Note that we have

$$F(\mathbf{1}_S)=f(S) \text{ for every } S\subseteq E.$$

Moreover,

$$F(x)=\mathbb{E}_{S\sim x}\left[f(S)\right]$$

where $S\sim x$ means sampling S by selecting each e with probility x_e. The matroid constraint set is given by

$$\mathcal{F}=\{S\subseteq V:\ S \text{ is an independent set of matroid } \mathcal{M}\}.$$

What is the right continuous relaxation for $\mathcal{F}$? For $S\subseteq E$, $\text{rank}(S)$ is defined by the max size of an independent set in S. Then the **polymatroid** of matroid $\mathcal{M}$ is given by

$$P=\left\{x\in[0,1]^{|E|}:\ \sum_{e\in S} x_e\leq \text{rank}(S)\quad \forall S\subseteq E\right\}.$$

Here, $P\cap\{0,1\}^{|E|}=\mathcal{F}$. Given the multilinear extension of f and the polymatroid P, we obtain the continuous relaxation given by

$$\text{maximize}\quad F(x)\quad \text{subject to}\quad x\in P.$$

Let us present an algorithm that solves the continuous relaxation.

Algorithm 2 Continuous greedy algorithm [CCPV07, Von08]

Input: multilinear extension F and polymatroid P
Start with $x(0)=0$.
Let $v(x)=\arg\max_{v\in P}\left\{\nabla F(x)^\top v\right\}$.
Set $dx/dt=v(x)$.
Output $x(1)$.

Theorem 10.2 ([CCPV07, Von08]). *Let $x(t)$ for $t \in [0, 1]$ denote the trajectory of the continuous greedy algorithm (2). Then $x(1) \in P$ and*

$$F(x(1)) \geq \left(1 - \frac{1}{e}\right) \max_{S \in \mathcal{F}} f(S).$$

By the theorem, the objective value by Algorithm 2 for the continuous relaxation is at least an $(1 - 1/e)$-approximation of the maximum value of the submodular set function subject to the matroid constraint.

The last step is to round the solution $x(1)$ which potentially has fractional components.

Theorem 10.3 ([AS04, CCPV07]). *There is a randomized polynomial time algorithm that given any $x \in P$, returns $S \in \mathcal{F}$ with*

$$\mathbb{E}\left[f(S)\right] \geq F(x).$$

Together with the previous theorem,

$$\mathbb{E}\left[f(S)\right] \geq F(x(1)) \geq \left(1 - \frac{1}{e}\right) \max_{S \in \mathcal{F}} f(S).$$

3 Continuous submodular functions

In this section, we consider a continuous extension of the discrete submodular function maximization problem. It is not difficult to observe that the multilinear extension of a submodular set function satisfies the following **diminishing returns (DR) property**:

$$F(x + \delta e_i) - F(x) \geq F(y + \delta e_i) - F(y)$$

for any $\delta \geq 0$ and $x, y \in [0, 1]^{|E|}$ with $x \leq y$. The multilinear extension is neither convex nor concave. However, it is **concave along any nonnegative direction (up-concave)** $v \geq 0$, i.e.,

$$g(t) = F(x + t \cdot v)$$

is concave with respect to t. Moreover, the multilinear extension is **smooth**, i.e. there exists $\beta > 0$ such that

$$\|\nabla F(x) - \nabla F(y)\|_2 \leq \beta \|x - y\|_2.$$

Based on these properties, we extend submodularity to continuous functions.

We say that a function $F : \mathbb{R}^d \to \mathbb{R}$ is **(continuous) DR-submodular** if it satisfies the diminishing returns (DR) property. For some domain C,

$$F(x + \delta e_i) - F(x) \geq F(y + \delta e_i) - F(y)$$

for any $\delta \geq 0$ and $x, y \in C$ with $x \leq y$.

Lemma 10.4 ([BMBK17]). *If F is DR-submodular, then it is up-concave.*

In addition, DR-submodular functions satisfy the following properties.

- A differentiable function is DR-submodular if and only if

$$\nabla F(x) \geq \nabla F(y) \quad \text{for any } x, y \text{ with } x \leq y.$$

- A twice-differentiable function is DR-submodular if and only if

$$\nabla^2 F(x) = \left(\frac{\partial^2 F(x)}{\partial x_i \partial x_j}\right) \leq 0.$$

On top of the multilinear extension of a submodular set function, there exist other examples of DR-submodular functions.

- Quadratic functions $x^\top Ax/2 + b^\top x + c$ with $A \leq 0$.
- $\sum_{i,j} \varphi_{i,j}(x_i - x_j)$ where $\varphi_{i,j}$ is convex for every i, j.
- $g(\sum_i w_i x_i)$ where g is concave and $w \geq 0$.
- $\text{logdet}(\sum_i x_i A_i)$ where each A_i is positive definite and $x \geq 0$.

Continuous DR-submodular arise in isotonic regression [Bac18], robust budget allocation [SJ17, SKIK14], online resource allocation [EF16], and adwords for e-commerce and advertising [DJ12, MSVV05]. Then we consider the following continuous DR-submodular maximization problem.

$$\text{maximize} \quad F(x) \quad \text{subject to} \quad x \in C$$

where F is continuous DR-submodular, F is monotone, i.e. $F(x) \leq F(y)$ for any x, y with $x \leq y$, and C is a convex constraint set. We further assume that $0 \in C$, C is bounded and **down-closed**, i.e., if $y \in C$, then any $0 \leq x \leq y$ belongs to C. if To solve the problem, we may the continuous greedy algorithm. We present the **conditional gradient method** due to Bian et al. [BMBK17], which is also referred to as the Frank-Wolfe algorithm for submodular maximization.

Algorithm 3 Conditional gradient method

Start with $x_0 = 0$.
for $t = 1, \ldots, T$ **do**
  Obtain $v_t \in \arg\max_{v \in C} \left\{\nabla F(x_{t-1})^\top v\right\}$.
  Update $x_t = x_{t-1} + (1/T)v_t$.
end for
Output x_T.

Theorem 10.5 ([BMBK17]). *Assume that F is monotone, β-smooth in the ℓ_2-norm, and DR-submodular. We further assume that $0 \in C$, C is down-closed, and $\|v\|_2 \leq R$ for any $v \in R$. Then x_T returned by conditional gradient (Algorithm 3) satisfies*

$$F(x_T) \geq \left(1 - \frac{1}{e}\right) \max_{x \in C} F(x) - \frac{\beta R^2}{2T}$$

under $F(0) = 0$.

Proof. Since F is β-smooth, we have

$$\begin{aligned} F(x_t) &\geq F(x_{t-1}) + \nabla F(x_{t-1})^\top (x_t - x_{t-1}) - \frac{\beta}{2}\|x_t - x_{t-1}\|_2^2 \\ &= F(x_{t-1}) + \frac{1}{T}\nabla F(x_{t-1})^\top v_t - \frac{\beta}{2T^2}\|v_t\|_2^2. \end{aligned} \tag{10.2}$$

Let x^* be an optimal solution, and let v^* be defined as

$$v^* := (x^* \vee x) - x = (x^* - x) \vee 0 \geq 0$$

where $p \vee q$ for two vectors p and q takes the coordinate-wise maximum values of p and q. Then we have $0 \leq v^* \leq x^*$, and the down-closedness of C implies that $v^* \in C$. Moreover, it follows from the monotonicity of F that

$$F(x + v^*) = F(x^* \vee x) \geq F(x^*) \tag{10.3}$$

. Note that

$$\nabla F(x_{t-1})^\top v_t \geq \nabla F(x_{t-1})^\top v^* \geq F(x_{t-1} + v^*) - F(x_{t-1}) \geq F(x^*) - F(x_{t-1})$$

where the first inequality is due to our choice of v_T, the second inequality holds because $v^* \geq 0$ and F is up-concave, and the third one comes from (10.3). Combined with (10.2), we obtain

$$F(x_t) \geq \left(1 - \frac{1}{T}\right) F(x_{t-1}) + \frac{1}{T} F(x^*) - \frac{\beta}{2T^2} R^2.$$

This implies that

$$F(x^*) - F(x_t) \leq \left(1 - \frac{1}{T}\right) (F(x^*) - F(x_{t-1})) + \frac{\beta R^2}{2T^2}$$

Furthermore, it follows that

$$\begin{aligned} F(x^*) - F(x_T) &\leq \left(1 - \frac{1}{T}\right)^T (F(x^*) - F(x_0)) + \frac{\beta R^2}{2T^2} \sum_{t=1}^{T} \left(1 - \frac{1}{T}\right)^{t-1} \\ &\leq \frac{1}{e} (F(x^*) - F(x_0)) + \frac{\beta R^2}{2T}, \end{aligned}$$

as required. □

References

[AS04] A.A. Ageev and M.I. Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization*, 8:307–328, 2004. 10.3

[Bac18] Francis Bach. Efficient algorithms for non-convex isotonic regression through submodular optimization. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 3

[BMBK17] Andrew An Bian, Baharan Mirzasoleiman, Joachim Buhmann, and Andreas Krause. Guaranteed Non-convex Optimization: Submodular Maximization over Domains. In Aarti Singh and Jerry Zhu, editors, *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 54 of *Proceedings of Machine Learning Research*, pages 111–120, 20–22 Apr 2017. 10.4, 3, 10.5

[CCPV07] Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a submodular set function subject to a matroid constraint (extended abstract). In Matteo Fischetti and David P. Williamson, editors, *Integer Programming and Combinatorial Optimization*, pages 182–196, 2007. 2, 2, 10.2, 10.3

[CFN77] G. Cornuéjols, M.L. Fisher, and G.L. Nemhauser. Location of bank accounts to optimize float: An analytic study of exact and approximate algorithm. *Management Science*, 23:789–810, 1977. 1

[DJ12] Nikhil R. Devanur and Kamal Jain. Online matching with concave returns. In *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing*, STOC '12, page 137–144, 2012. 3

[DK11] Abhimanyu Das and David Kempe. Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection. In *International Conference on International Conference on Machine Learning (ICML)*, pages 1057–1064, 2011. 1

[EF16] Reza Eghbali and Maryam Fazel. Designing smoothing functions for improved worst-case competitive ratio in online optimization. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 3

[KG05] Andreas Krause and Carlos Guestrin. Near-optimal nonmyopic value of information in graphical models. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 324–331, 2005. 1

[KKT03] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 137–146, 2003. 1

[KR15] Jon Kleinberg and Maithra Raghu. Team performance with test scores. In *ACM Conference on Economics and Computation (EC)*, pages 511–528, 2015. 1

[LB10] Hui Lin and Jeff Bilmes. Multi-document summarization via budgeted maximization of submodular functions. In *Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (HLT)*, pages 912–920, 2010. 1

[LB11] Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In *Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (HLT)*, pages 510–520, 2011. 1

[MJK18] Baharan Mirzasoleiman, Stefanie Jegelka, and Andreas Krause. Streaming non-monotone submodular maximization: Personalized video summarization on the fly. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 1379–1386, 2018. 1

[MKSK16] Baharan Mirzasoleiman, Amin Karbasi, Rik Sarkar, and Andreas Krause. Distributed submodular maximization. *Journal of Machine Learning Research*, 17(1):1–44, 2016. 1

[MSVV05] A. Mehta, A. Saberi, U. Vazirani, and V. Vazirani. Adwords and generalized on-line matching. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pages 264–273, 2005. 3

[NWF78] G.L. Nemhauser, L.A. Wolsey, and M.L. Fisher. An analysis of approximations for maximizing submodular set functions - i. *Mathematical Programming*, 14:265–294, 1978. 1, 10.1

[SJ17] Matthew Staib and Stefanie Jegelka. Robust budget allocation via continuous submodular functions. In Doina Precup and Yee Whye Teh, editors, *International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pages 3230–3240, 06–11 Aug 2017. 3

[SKIK14] Tasuku Soma, Naonori Kakimura, Kazuhiro Inaba, and Ken-ichi Kawarabayashi. Optimal budget allocation: Theoretical guarantee and efficient algorithm. In *International Conference on Machine Learning (ICML)*, pages 351–359, 2014. 3

[Svi04] M. Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Operations Research Letters*, 32:41–43, 2004. 1

[SVY20] Shreyas Sekar, Milan Vojnovic, and Se-Young Yun. A test score based approach to stochastic submodular optimization. *Management Science*, 2020. 1

[TIWB14] Sebastian Tschiatschek, Rishabh Iyer, Haochen Wei, and Jeff Bilmes. Learning mixtures of submodular functions for image collection summarization. In *Advances in Neural Information Processing Systems*, pages 1413–1421, 2014. 1

[Udw21] Rajan Udwani. Submodular order functions and assortment optimization. Technical report, University of California, Berkeley, 2021. 1

[Von08] Jan Vondrak. Optimal approximation for the submodular welfare problem in the value oracle model. In *ACM Symposium on Theory of Computing (STOC)*, page 67–74, 2008. 2, 2, 10.2