Lecture 10: discrete and continuous submodular function maximization

Dabeen Lee

Industrial and Systems Engineering, KAIST

2025 Winter Lecture Series on Combinatorial Optimization

January 16, 2025

Outline

- Submodular set function maximization
- Submodular function maximization subject to a matroid constraint
- Continuous DR-submodular maximization

Outline

- Submodular set function maximization
- Submodular function maximization subject to a matroid constraint
- Continuous DR-submodular maximization

Submodular set function

- A function $f : 2^V \to \mathbb{R}$ over the subsets of finite ground set V is submodular if

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B) \quad \forall A, B \subseteq V.$$

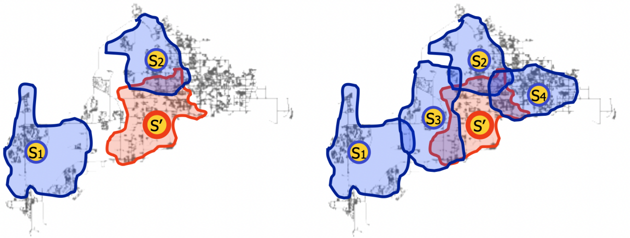
- Submodularity is equivalent to the diminishing (marginal) returns property.
- To be precise, $f : 2^V \to \mathbb{R}$ is submodular if and only if

$$\underbrace{f(A \cup \{v\}) - f(A)}_{\text{marginal return from adding } v \text{ to } A} \quad \geq \quad \underbrace{f(B \cup \{v\}) - f(B)}_{\text{marginal return from adding } v \text{ to } B} \qquad \forall A \subseteq B,\ v \notin B$$

- A common analogy: at a buffet restaurant, you feel less happy with the second plate than with the first plate.

Examples of submodular set functions

The area covered by sensors



1

Figure: Adding sensor s' to $\{s_1, s_2\}$ (left) and $\{s_1, s_2, s_3, s_4\}$ (right)

- Let A_v be the area covered by a sensor v.
- Then f defined as

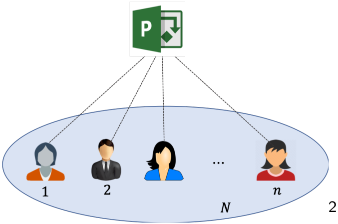
$$f(S) = \left| \bigcup_{v \in S} A_v \right|, \quad S \subseteq V$$

is submodular.

[1] Taken from [Krause and Golovin, 2014]

Examples of submodular set functions

Joint stochastic utility



2

Figure: Employees with random performance

- n employees with random performance values $X_1, \ldots, X_n$.
- Then

$$\mathbb{E}\left\{\max_{v \in S} X_v\right\}, \ \mathbb{E}\left\{\sqrt{\sum_{v \in S} X_v}\right\}, \ \mathbb{E}\left\{\sqrt{\sum_{v \in S} X_v^2}\right\}$$

are all submodular functions over $S \subseteq V$.

[2] Taken from Milan Vojnovic's slides.

Submodular set function maximization

Submodular function maximization (SFM)

$$\begin{aligned} \text{maximize} \quad & f(S) \\ \text{subject to} \quad & S \in \mathcal{F} \subseteq 2^V \end{aligned}$$

where

- f is submodular and monotone, i.e., $f(A) \leq f(B)$ for any $A \subseteq B$,
- $\mathcal{F}$ is the constraint set.

Constraint set examples:

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

Applications of SFM

Business operations

- Online freelancing platforms [Sekar et al., 2020].
- Team selection - sports teams, online gaming [Kleinberg and Raghu, 2015].
- Assortment selection in online shopping websites [Udwani, 2021].
- Influence maximization [Kempe et al., 2003].

Machine learning

- Feature and variable selection [Krause and Guestrin, 2005].
- Dictionary learning [Das and Kempe, 2011].
- Document summarization [Lin and Bilmes, 2010, 2011].
- Image summarization [Mirzasoleiman et al., 2018, Tschiatschek et al., 2014].
- Active set selection in non-parametric learning [Mirzasoleiman et al., 2016].

NP-hardness of SFM

- [Cornuéjols et al., 1977]

  SFM is **NP-hard** even with a monotone objective subject to a cardinality constraint.
- No polynomial time exact algorithm is known.
- If one exists, it would imply a polynomial time exact algorithm for traveling salesman problem (TSP).
- However, there exist polynomial time constant approximation algorithms.

Approximation algorithm

- 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 $\alpha > 0$.

Greedy algorithm

- Let us explain a simple greedy algorithm by [Nemhauser et al., 1978] 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 $v \in \arg\max\{f(S \cup \{v\}) - f(S) : \ v \in V \setminus S\}$
end while
Return S

Greedy algorithm

Theorem ([Nemhauser et al., 1978])

Let $\bar{S} \subseteq V$ be the outcome of Algorithm 1. Assume that $f(\emptyset) = 0$. Then

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

Greedy algorithm

Approximation algorithms for SFM

Monotone SFM

- [Nemhauser et al., 1978]

  $(1-1/e)$-approximation algorithm for a **cardinality constraint**, where $(1-1/e) \approx 0.63$. Moreover, the factor $(1-1/e)$ is tight.

- [Sviridenko, 2004]

  $(1-1/e)$-approximation algorithm for a **knapsack constraint**.

- [Calinescu et al., 2007, Vondrak, 2008]

  $(1-1/e)$-approximation algorithm for a **matroid constraint**.

- [Kulik et al., 2009]

  $(1-1/e)$-approximation algorithm for a **system of linear constraints**.

There are constant approximation algorithms for non-monotone SFM.

Approximation algorithm for a matroid constraint

Outline

- Obtain a continuous relaxation of (discrete) SFM.
- Solve the continuous relaxation.
- Round the (fractional) solution to obtain a (discrete) solution to SFM.

Approximation algorithm for a matroid constraint

Outline

- Obtain a continuous relaxation of (discrete) SFM.

  $\rightarrow$ **Multilinear extension** + **Polymatroid**

- Solve the continuous relaxation.

  $\rightarrow$ **Continuous greedy algorithm**

- Round the (fractional) solution to obtain a (discrete) solution to SFM.

  $\rightarrow$ **Pipage rounding**

Multilinear extension

Multilinear extension [Vondrak, 2008]

- A set function $f : 2^V \to \mathbb{R}$ can be turned into a function $f : \{0,1\}^V \to \mathbb{R}$ over binary variables.
- Given $f : \{0,1\}^V \to \mathbb{R}$,

$$F(x) = \sum_{S \subseteq V} f(S) \prod_{v \in S} x_v \prod_{v \notin S} (1 - x_v) \quad \text{for } x \in [0,1]^V$$

  is the multilinear extension of f.
- Note that

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

- Moreover,

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

  where $S \sim x$ means sampling S by selecting each v with probility x_v.

Polymatroid

Polymatroid [Edmonds, 1970]

- 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 $S \subseteq V$, $\text{rank}(S)$ is defined by the max size of an independent set in S.
- Then the polymatroid of matroid $\mathcal{M}$ is given by

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

- Here, $P \cap \{0,1\}^V = \mathcal{F}$.

Continuous relaxation

- Given the multilinear extension F and the polymatroid P,

$$\begin{aligned} \text{maximize} \quad & F(x) \\ \text{subject to} \quad & x \in P \subseteq [0,1]^V \end{aligned}$$

  is a continuous relaxation of SFM, given by

$$\begin{aligned} \text{maximize} \quad & f(S) \\ \text{subject to} \quad & S \in \mathcal{F} \subseteq 2^V. \end{aligned}$$

- In particular,

$$\max_{x \in P} F(x) \geq \max_{S \in \mathcal{F}} f(S).$$

Continuous greedy algorithm

Algorithm 2 Continuous greedy algorithm [Calinescu et al., 2007, Vondrak, 2008]

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 $x(1)$.

- We can discretize the algorithm.

Theorem ([Calinescu et al., 2007, Vondrak, 2008])

$x(1) \in P$ and

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

Pipage rounding

Theorem ([Ageev and Sviridenko, 2004, Calinescu et al., 2007])

There is a randomized polynomial time algorithm that given any $x \in P$, returns $S \in \mathcal{F}$ with

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

Together with the previous theorem,

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

Extension to the continuous domain

Extension

- There were two continuous components in the algorithm.

  1. Multilinear extension
  2. Continuous greedy algorithm

- We extend the multilinear extension to **continuous submodular functions**.

- We adapt the continuous greedy algorithm to develop the **conditional gradient method**, which is a first-order iterative algorithm.

Properties of the multilinear extension

- Recall that the multilinear extension is given by

$$F(x) = \sum_{S \subseteq V} f(S) \prod_{v \in S} x_v \prod_{v \notin S} (1 - x_v).$$

- It 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 $x, y \in [0, 1]^V$ with $x \leq y$.
- The multilinear extension is neither convex nor concave.
- But 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.
- 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.**

Continuous DR-submodular function

- 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 ([Bian et al., 2017])

If F is DR-submodular, then it is up-concave.

Equivalent definitions and examples

More properties of DR-submodular functions [Bian et al., 2017]

- 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

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

Examples

- 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 $w \geq 0$.
- $\operatorname{logdet}(\sum_i x_i A_i)$ where each A_i is positive definite and $x \geq 0$.

Applications of DR-submodular functions

- Isotonic regression [Bach, 2018].
- Robust budget allocation [Soma et al., 2014, Staib and Jegelka, 2017].
- Online resource allocation [Eghbali and Fazel, 2016].
- Adwords for e-commerce and advertising [Devanur and Jain, 2012, Mehta et al., 2005].

Continuous submodular maximization

Continuous submodular maximization

$$\begin{aligned} \text{maximize} \quad & F(x) \\ \text{subject to} \quad & x \in C \end{aligned}$$

where

- F is continuous DR-submodular,
- F is monotone, i.e. $F(x) \leq F(y)$ for any x, y with $x \leq y$,
- C is a convex constraint set,
- $0 \in C$,
- C is **down-closed**, i.e. $y \in C$ and $0 \leq x \leq y$ implies $x \in C$.

Conditional gradient method [Bian et al., 2017]

- Extension of the continuous greedy algorithm.
- Often called the Frank-Wolfe algorithm for submodular maximization.

Conditional gradient method

Recall

Algorithm 2 Continuous greedy algorithm [Calinescu et al., 2007, Vondrak, 2008]

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 $x(1)$.

The conditional gradient method for SFM is given by

Algorithm 3 Conditional gradient method [Bian et al., 2017]

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.

Conditional gradient method

Theorem ([Bian et al., 2017])

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 $F(0) = 0$.

Conditional gradient method

Conditional gradient method

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

F. 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. URL https://proceedings.neurips.cc/paper/2018/file/6ea9ab1baa0efb9e19094440c317e21b-Paper.pdf.

A. A. Bian, B. Mirzasoleiman, J. Buhmann, and A. Krause. Guaranteed Non-convex Optimization: Submodular Maximization over Domains. In A. Singh and J. 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.

G. Calinescu, C. Chekuri, M. Pál, and J. Vondrák. Maximizing a submodular set function subject to a matroid constraint (extended abstract). In M. Fischetti and D. P. Williamson, editors, Integer Programming and Combinatorial Optimization, pages 182–196, 2007.

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

A. Das and D. 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.

N. R. Devanur and K. 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.

J. Edmonds. Submodular functions, matroids, and certain polyhedra. In Combinatorial Structures and Their Applications, pages 69–87. Gordon and Breach, 1970.

R. Eghbali and M. 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. URL https://proceedings.neurips.cc/paper/2016/file/3c1e4bd67169b8153e0047536c9f541e-Paper.pdf.

D. Kempe, J. Kleinberg, and E. 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.

J. Kleinberg and M. Raghu. Team performance with test scores. In ACM Conference on Economics and Computation (EC), pages 511–528, 2015.

A. Krause and D. Golovin. Submodular Function Maximization, page 71–104. Cambridge University Press, 2014. doi: 10.1017/CBO9781139177801.004.

A. Krause and C. Guestrin. Near-optimal nonmyopic value of information in graphical models. In Conference on Uncertainty in Artificial Intelligence (UAI), pages 324–331, 2005.

A. Kulik, H. Shachnai, and T. Tamir. Maximizing submodular set functions subject to multiple linear constraints. In Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '09, page 545–554, 2009.

H. Lin and J. 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.

H. Lin and J. 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.

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. doi: 10.1109/SFCS.2005.12.

B. Mirzasoleiman, A. Karbasi, R. Sarkar, and A. Krause. Distributed submodular maximization. Journal of Machine Learning Research, 17(1): 1–44, 2016.

B. Mirzasoleiman, S. Jegelka, and A. Krause. Streaming non-monotone submodular maximization: Personalized video summarization on the fly. In AAAI Conference on Artificial Intelligence (AAAI), pages 1379–1386, 2018.

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

S. Sekar, M. Vojnovic, and S.-Y. Yun. A test score based approach to stochastic submodular optimization. Management Science, 2020.

T. Soma, N. Kakimura, K. Inaba, and K.-i. Kawarabayashi. Optimal budget allocation: Theoretical guarantee and efficient algorithm. In International Conference on Machine Learning (ICML), pages 351–359, 2014.

M. Staib and S. Jegelka. Robust budget allocation via continuous submodular functions. In D. Precup and Y. W. Teh, editors, International Conference on Machine Learning (ICML), volume 70 of Proceedings of Machine Learning Research, pages 3230–3240, 06–11 Aug 2017.

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

S. Tschiatschek, R. Iyer, H. Wei, and J. Bilmes. Learning mixtures of submodular functions for image collection summarization. In Advances in Neural Information Processing Systems, pages 1413–1421, 2014.

R. Udwani. Submodular order functions and assortment optimization. Technical report, University of California, Berkeley, 2021.

J. 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.