6.883 Learning with Combinatorial Structure
Notes for Lecture 16
Author: Arpit Agarwal

1 Maximizing a Submodular Function

In the last lecture we looked at maximization of a monotone submodular function, i.e. submodular functions such that $F(S) \leq F(T)$ for all $S \subseteq T$, with cardinality constraints. We saw that the greedy algorithm, which picks elements in descending order of function value, gives good approximation guarantee.

Moreover, the approximation ratio depends on curvature c of a submodular function, defined as

$$c = \max_{e \in \mathcal{V}} 1 - \frac{F(\mathcal{V}) - F(\mathcal{V} \setminus e)}{F(e)} \, ,$$

where c captures the (non-)linearity of the submodular function, for $c = 0$ the function is linear and the approximation ratio is 1; while for $c = 0$ the function is far from linear and the approximation ratio is $(1 - \frac{1}{e})$.

The greedy algorithm also extends to the case where a function is approximately submodular. In this case we define submodularity ratio of a function as

$$\gamma_k = \min_{S \cap T = \emptyset, |S| \leq k} \frac{\sum_{e \in S} F(T \cup e) - F(T)}{F(T \cup S) - F(T)} \, ,$$

The greedy algorithm finds a set S_G such that [2]

$$F(S_G) \geq \big(1 - \frac{1}{e^{\gamma_k}}\big) F(S^*)$$

In this lecture we shift our attention towards maximization of monotone submodular functions under matroid constraints.

2 Maximization under Matroid Constraints

We look at the following maximization problem:

$$\text{(P1)} : \max_{S \in \mathcal{I}} F(S)$$

where F is monotone submodular and $\mathcal{I}$ is the independent set of a matroid $\mathcal{M}$. These type of problems are encountered in various applications:

Example 1 (Camera Network)**.** *In this problem given the set of all possible locations for installation of cameras and directions for each camera, one wants to choose a subset of the cameras and set their directions so as to maximize the total coverage. More formally, let $\mathcal{V} = \{1_a, 1_b, \cdots, n_a, n_b\}$ be the ground set of matroid $\mathcal{M}$, where n is the total number of cameras, i_a represents camera i looking in direction a. Let $P_i = \{i_a, i_b\}$, for $i \in \{1, \cdots, n\}$. The optimization problem is*

$$\begin{aligned} &\max F(S) \\ \text{s.t.} \quad &|S \cap P_i| \leq 1, \forall i \in \{1, \cdots, n\}, \end{aligned}$$

where F is a submodular function such that $F(S)$ denotes the total coverage of set S.

Example 2 (Welfare Maximization)**.** *In this example given a set of items and people, one wants to distribute the items to people so as to maximize the social welfare subject to constraints that one item is distributed only to one person. Formally, let there be n people and $\mathcal{V}$ denote the set of all items, the optimization problem is*

$$\begin{aligned} &\max_{S_1, \cdots, S_n \subseteq \mathcal{V}} \sum_{i=1}^{n} F_i(S_i) \\ \text{s.t.} \quad &S_i \cap S_j = \emptyset, \quad \forall i \neq j, \end{aligned}$$

where S_i denotes the set of items assigned to person i, and F_i is the welfare function of person i.

The greedy algorithm for the problem P1 provides the approximation guarantee of $1/2$, which is not optimal. Calinescu et al. [1] show that the following relaxation to this problem results in optimal approximation guarantee:

$$\text{(P2)} : \max_{x \in \text{conv}(\mathcal{I})} g(x).$$

The solution $\hat{x}$ to the above relaxed problem can be rounded-off to a binary solution y such that $g(y) \geq g(\hat{x})$.

The important question: How to find the function g for the relaxed problem? One possible solution might be to take g to be the Lovász extension of F. However, the Lovász extension is convex and maximizing a convex function might be hard. In the next section we see that the multilinear extension of F is a good choice for g.

3 Multilinear Extension

Given a submodular function F, the multilinear extension function g for $x \in [0,1]^n$ is defined as:

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

where the expectation is over random draws of sets $S \subseteq \mathcal{V}$, with each element $e \in \mathcal{V}$ drawn in S with probability x_e. Expanding the above expectation, we get

$$g(x) = \sum_{S \subseteq \mathcal{V}} F(S) \prod_{e \in S} x_e \prod_{e \notin S} (1 - x_e) \,.$$

We make the following observations:

Observation 1. *$g(1_S) = F(S)$, i.e. the multilinear extension agrees with F on all binary vectors.*

Observation 2. *$g(x) \geq f(x)$ where f is the Lovász extension of F, i.e. the multilinear extension forms an upper bound on the Lovász extension.*

Observation 3. *If F is monotone non-decreasing, then $\frac{\partial g}{\partial x_i} \geq 0, \forall i$.*

Proof. Let R be a random set, $R \subseteq \mathcal{V} \setminus i$, whose elements j appear with probability x_j. Now, observe that if a function is multilinear then its derivative is constant if other entries are fixed.

$$\begin{aligned}
\frac{\partial g}{\partial x_i} &= g(x_1, \cdots, x_{i-1}, 1, x_{i+1}, \cdots, x_n) - g(x_1, \cdots, x_{i-1}, 0, x_{i+1}, \cdots, x_n) \\
&= \mathbb{E}[F(R \cup i)] - \mathbb{E}[F(R)] \\
&\geq 0 \,.
\end{aligned}$$

The last inequality if due to the monotonicity of F.

□

Observation 4. *If F is submodular, then $\frac{\partial^2 g}{\partial x_i \partial x_j} \leq 0, \forall i, j$.*

Proof. Similar to the previous proof, let R be a random set, $R \subseteq \mathcal{V} \setminus \{i, j\}$, then the partial derivative can be computed as:

$$\begin{aligned} \frac{\partial^2 g}{\partial x_i \partial x_j} &= \big(\mathbb{E}[F(R \cup i \cup j)] - \mathbb{E}[F(R \cup i)]\big) - \big(\mathbb{E}[F(R \cup j)] - \mathbb{E}[F(R)]\big) \\ &\leq 0\,. \end{aligned}$$

The last inequality if due to the submodularity of F.

□

Corollary 1. *The following results hold:*

1. *If F is submodular, then g is non-decreasing in all directions.*
2. *If F is submodular, then g is concave along all directions $d \geq 0$.*
3. *If F is submodular, then g is convex in directions $1_i - 1_j$.*

These results can be proved by considering the function $\phi(\xi) = g(x + \xi d)$ and differentiating it.

The property of g that it is concave in all directions $d \geq 0$ makes it very appealing for efficient maximization. In the next section we look at a greedy algorithm that exploits this property of g by optimizing it along directions $d \geq 0$.

Note, also, the difference between multilinear extension and Lovász extension: the former is concave in certain direction and convex in others, while the latter is convex in all directions.

Evaluating the multilinear extension is difficult in general, as it requires summing over a potentially large number of sets. One possible solution might be to evaluate the function by performing sampling. To evaluate the function value at x, we randomly sample m sets $R_1, \cdots, R_m$ according to x. Then with probability $1 - e^{-m\epsilon^2/4}$

$$\Big|\frac{1}{m}\sum_{i=1}^{m} F(R_i) - g(x)\Big| \leq \epsilon \max_S F(S)$$

Algorithm 1 Continuous Greedy Algorithm

Initialize: $x^0 = 0$
for $t = 1$ **to** T **do**
  $s^t = \arg\max_{s \in \mathrm{conv}(\mathcal{I})} \langle s, \nabla g(x^t) \rangle$
  $x^{t+1} = x^t + \alpha_t s^t$
end for
Return: x^T

4 Continuous Greedy Algorithm

We now present a greedy algorithm for solving P2 with g being the multilinear extension of the monotone submodular function F. Algorithm 1 outlines the pseudo-code for the algorithm.

In each iteration the algorithm finds a direction s^t that best aligns with the gradient $\nabla g(x^t)$. The next iterate is generated by taking a linear combination of the current iterate and the direction s^t. A reasonable choice for the parameter α_t is $\alpha_t = \frac{1}{T}$, as we will later see. Note that this is very similar to the Frank-Wolfe algorithm studied in the earlier lectures.

We state the important theorem that this algorithm achieves good approximation ratio.

Theorem 1. *The continuous greedy algorithm with $\alpha_t = \frac{1}{T}$ finds a point $\hat{x}$ with*

$$g(\hat{x}) \geq \Big(1 - \frac{1}{e}\Big)\mathrm{OPT} \ - \epsilon$$

Before proving this theorem we state two lemmas that lead to the proof.

Lemma 1. *For any polytope $\mathcal{P} \subseteq [0,1]^n$, $x \in [0,1]^n$ and $\mathrm{OPT} = \max_{x \in \mathcal{P}} g(x)$, there exists $v \in \mathcal{P}$ such that*

$$\langle v, \nabla g(x) \rangle \geq \mathrm{OPT} - g(x)$$

Proof. Let v be the optimal solution, i.e. $g(v) = \mathrm{OPT}$. Consider the direction $d = (x \vee v) - x = (v - x) \vee 0$, where $\vee$ denotes the pointwise maximum operator. Observe that $d \geq 0$ and $d \leq v$.

Let $\phi(\eta) = g(x + \eta d)$. Now, ϕ is a concave function for $\eta \in [0,1]$ due to Corollary 1 and the fact that $d \geq 0$. This implies

$$\phi(1) \leq \phi(0) + \phi'(0) \tag{1}$$

Now $\phi(1) = g(x \vee v)$, $\phi(0) = g(x)$ and $\phi'(0) = \langle d, \nabla g(x) \rangle$. Also,

$$\text{OPT} = g(v) \leq g(x \vee v) \tag{2}$$

where the last inequality is due to the monotonicity of g. Combining the two equations above, we get

$$\langle d, \nabla g(x) \rangle \geq g(x \vee v) - g(x) \geq \text{ OPT} - g(x) \tag{3}$$

Now, the fact that $\nabla g(x)$ is non-negative and $d \leq v$ gives the result

$$\langle v, \nabla g(x) \rangle \geq \text{ OPT} - g(x)$$

□

Corollary 2. *For any polytope $\mathcal{P} \subseteq [0,1]^n$, $x^t \in [0,1]^n$, and $s^t = \arg\max_{\mathcal{P}} \langle s, \nabla g(x^t) \rangle$,*

$$\langle s^t, \nabla g(x^t) \rangle \geq \text{ OPT} - g(x^t)\,.$$

Lemma 2. *Taking $\alpha_t = \frac{1}{T}$,*

$$g(x^T) \geq \Big(1 - (1 - \frac{1}{T})^T\Big)\text{OPT} - \frac{C_g}{2T}\,,$$

for some constant C_g.

Proof. Taking the Taylor expansion on $g(x^{t+1})$

$$\begin{aligned}
g(x^{t+1}) &\geq g(x^t) + \alpha \langle s^t, \nabla g(x^t) \rangle - \frac{C_g}{2}\alpha^2 \\
&\geq g(x^t) + \alpha[\text{ OPT} - g(x^t)] - \frac{C_g}{2}\alpha^2 \qquad \text{Using Corollary 2} \\
&= (1-\alpha) g(x^t) + \alpha \text{ OPT} - \frac{C_g}{2}\alpha^2
\end{aligned}$$

Changing signs and adding OPT to both sides, we get

$$
\begin{aligned}
\mathrm{OPT} - g(x^{t+1}) &\leq (1-\alpha)[\,\mathrm{OPT} - g(x^t)] + \frac{C_g}{2}\alpha^2 \\
&\leq (1-\alpha)^{t+1}[\,\mathrm{OPT} - g(x^0)] + \frac{C_g}{2}\alpha^2(t+1)
\end{aligned}
$$

Taking $g(0) = 0$ and $\alpha = \frac{1}{T}$, proves the result

$$
g(x^T) \geq \Big(1 - \big(1 - \frac{1}{T}\big)^T\Big)\mathrm{OPT} - \frac{C_g}{2T}\,,
$$

$\square$

Proof. (of Theorem 1) The result follows immediately from Lemma 2, by letting $T \to \infty$. $\square$

Rounding: Now, given the output $\hat{x}$ of the algorithm, one needs to round it to find a solution of P1. The following procedure leads to a binary $y = 1_T$ with $F(T) = g(y) \geq g(\hat{x})$:

1. Pick two fractional entries $\hat{x}_i, \hat{x}_j$.
2. Move in one of the convex direction $\pm(1_i - 1_j)$. One of them will be without loss.

5 Non-monotone Functions

There might be some applications where the submodular function is non-monotone, i.e. it might not be the case that $F(S) \leq F(T)$ for $S \subseteq T$. Examples of this include the graph cut function where the cut size might reduce as we add more nodes in the set; mutual information etc. We might still assume that $F(S) \geq 0, \forall S$.

The greedy approach can fail in this setting as can be seen by a simple example:

Suppose we have a set of sensors, with each sensor a having a coverage defined by area(a), and cost defined by $c(a)$; the submodular function F is defined as

$$
F(A) = \big|\cup_{a\in A}\,\text{area}(a)\big| - \sum_{a\in A} c(a)
$$

The problem is to find a subset A of sensors which maximize the above function. Figure 1 gives an example where if we try solve this problem using a greedy approach by picking

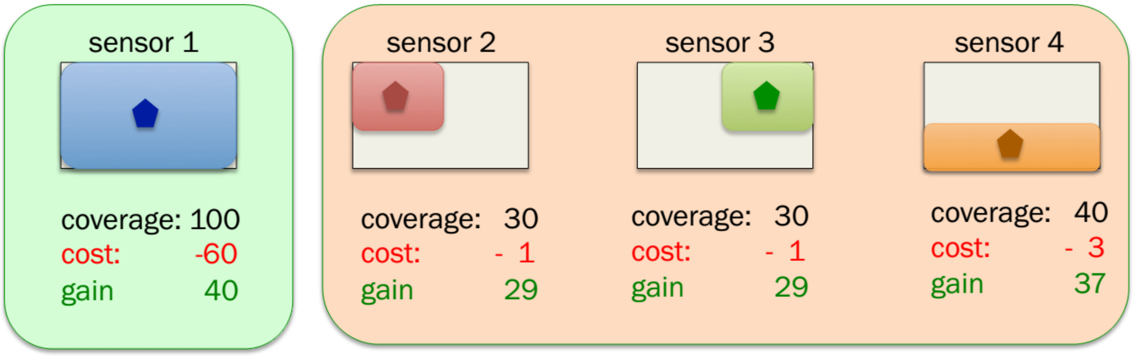

Figure 1: Example that shows greedy algorithm fails for non-monotone submodular maximization.

the sensors in descending order of coverage minus cost, we would not be able to find the optimal set of sensors. In this example we would pick the sensor with gain 40 and stop with function value $F(A) = 40$. The optimal strategy, however, is to pick the other three sensors with function value $F(A^*) = 95$.

Picking at random: For a non-monotone non-negative submodular function F, let us construct a set R by picking elements from $\mathcal{V}$ uniformly at random:

$$\Pr[\,\text{include } i] = \frac{1}{2} \qquad\qquad \text{for all i}\,.$$

This random set of items has a surprising expected approximation ratio:

$$\mathbb{E}[F(R)] \geq \frac{1}{4} F(S^*)\,.$$

Moreover, if the function F is symmetric then

$$\mathbb{E}[F(R)] \geq \frac{1}{2} F(S^*)\,.$$

However, the same random picking will not work for monotone submodular maximization with cardinality constrains.

In the next class we will look at methods for maximizing non-monotone submodular functions in more detail.

References

[1] Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a submodular set function subject to a matroid constraint. In *Integer programming and combinatorial optimization*, pages 182–196. Springer, 2007.

[2] Abhimanyu Das and David Kempe. Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection. *arXiv preprint arXiv:1102.3975*, 2011.

[3] Jan Vondrák. Lecture notes for CS 369P, 2010.