1 Submodular Functions and Convexity

Let $f : 2^S \to \mathbb{R}$ be a submodular set function. We discuss a connection between submodular functions and convexity that was shown by Lovász [3].

Given an arbitrary (not necessarily submodular) set function $f : 2^S \to \mathbb{R}$, we can view it as assigning values to the integer vectors in the hypercube $[0,1]^n$ where $n = |S|$. That is, for each $U \subseteq S$, $f(\chi(U)) = f(U)$. We say that a function $\hat{f} : [0,1]^n \to \mathbb{R}$ is an *extension* of f if $\hat{f}(\chi(U)) = f(U)$ for all $U \subseteq S$; that is $\hat{f}$ assigns a value to each point in the hypercube and agrees with f on the characterstic vectors of the subsets of S. There are several ways to define an extension and we consider one such below.

Let $S = \{1, 2, \ldots, n\}$. Consider a vector $c = (c(1), \ldots, c(n))$ in $[0,1]^n$ and let $p_1 > p_2 > \ldots > p_k$ be the distinct values in $\{c(1), c(2), \ldots, c(n)\}$. Define $q_k = p_k$ and $q_j = p_j - p_{j+1}$ for $j = 1, \ldots, k-1$. For $1 \leq j \leq k$, we let $U_j = \{i \mid c(i) \geq p_j\}$. Define $\hat{f}$ as follows:

$$\hat{f}(c) = (1 - p_1) f(\emptyset) + \sum_{j=1}^{k} q_j f(U_j)$$

As an example, if $c = (0.75, 0.3, 0.2, 0.3, 0)$ then

$$\hat{f}(c) = 0.25 \cdot f(\emptyset) + 0.45 \cdot f(\{1\}) + 0.1 \cdot f(\{1,2,4\}) + 0.2 \cdot f(\{1,2,3,4,5\})$$

In other words c is expressed as a convex combination $\chi(\emptyset) + \sum_{j=1}^{k} q_j \chi(U_j)$ of vertices of the hypercube, and $\hat{f}(c)$ is the natural interpolation. It is typically assumed that $f(\emptyset) = 0$ (one can always shift any function to achieve this) and in this case we can drop the term $(1 - p_1) f(\emptyset)$; however, it is useful to keep in mind the implicit convex decomposition.

Lemma 1 *If f is submodular then $\hat{f}(c) = \max\{cx \mid x \in EP_f\}$.*

We leave the proof of the above as an exercise. It follows by considering the properties of the Greedy algorithm for maximizing over polymatroids that was discussed in the previous lecture.

Theorem 2 (Lovász) *A set function $f : 2^S \to \mathbb{R}$ with $f(\emptyset) = 0$ is submodular iff $\hat{f}$ is convex.*

Proof: Suppose f is submodular. Let $c_1, c_2 \in [0,1]^n$ and $t \in [0,1]$ and let $c = tc_1 + (1-t)c_2$. To show that $\hat{f}$ is convex we need to show that $\hat{f}(c) \leq \hat{f}(tc_1) + \hat{f}((1-t)c_2)$. This follows easily from Lemma 1. Let $x^* \in EP_f$ be such that $\hat{f}(c) = c \cdot x^* = tc_1 \cdot x^* + (1-t)c_2 \cdot x^*$. Then $\hat{f}(tc_1) \geq tc_1 \cdot x^*$ and $\hat{f}((1-t)c_2) \geq (1-t)c_2 \cdot x^*$ and we have the desired claim.

Now suppose $\hat{f}$ is convex. Let $A, B \subseteq S$. From the definition of $\hat{f}$ we note that $\hat{f}((\chi(A) + \chi(B))/2) = \hat{f}(\chi(A \cup B)/2) + \hat{f}(\chi(A \cap B)/2)$ (the only reason to divide by 2 is to ensure that we stay in $[0,1]^n$). On the other hand, by convexity of $\hat{f}$, $\hat{f}((\chi(A)+\chi(B))/2) \leq \hat{f}(\chi(A)/2) + \hat{f}(\chi(B)/2)$. Putting together these two facts, we have $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$, and hence f is submodular. $\square$

Corollary 3 *If f is submodular then* $\min_{U \subseteq S} f(S) = \min_{c \in [0,1]^n} \hat{f}(c)$.

Proof: Clearly $\min_{c \in [0,1]^n} \hat{f}(c) \leq \min_{U \subseteq S} f(S)$. To see the converse, let $c^* \in [0,1]^n$ achieve the minimum of $\min_{c \in [0,1]^n} \hat{f}(c)$. Then one of the sets in the convex combination of c^* in the definition of the extension achieves a value equal to $\hat{f}(c^*)$. □

The above shows that submodular function minimization can be reduced to convex optimization problem in a natural fashion. One advantage of an extension as above is that one can use it as a relaxation in optimization problems involving submodular functions and additional constraints. For example we may want to solve $\min_{U \subseteq S} f(S)$ subject to U satisfying some additional constraints that could perhaps be modeled as $x(S) \in P$ for some convex set P. Then we could solve $\min\{\hat{f}(x) \mid x \in P\}$ as a relaxation and round the solution in some fashion. There are several examples of this in the literature.

2 Combinatorial Algorithms for Submodular Function Minimization

We saw in last lecture an algorithm for solving the submodular function minimization problem (SFM): given f as a value oracle, find $\min_{U \subseteq S} f(S)$. The algorithm was based on solving a linear program via the ellipsoid method and has a strongly polynomial running time. A question of interest is whether there is a polynomial time "combinatorial" algorithm for this problem. Although there is no clear-cut and formal definition of a combinatorial algorithm, typically it is an algorithm whose operations have some combinatorial meaning in the underlying structure of the problem. Cunningham [**?**] gave a pseudo-polynomial time algorithm for this problem in 1985. It is only in 2000 that Schrijver [**?**] and independently Iwata, Fleischer and Fujishige gave polynomial time combinatorial algorithms for SFM. There have been several papers that followed these two; we mention the algorithm(s) of Iwata and Orlin [2] that have perhaps the shortest proofs. All the algorithms follow the basic outline of Cunningham's approach which was orginally developed by him for the special case of SFM that arises in the separation oracle for the matroid polytope.

Two excellent articles on this subject by Fleischer [1] and Toshev [5]. We set up the min-max result on which the algorithms are based and reader should refer to [1, 5, 4, 2] for more details.

2.1 Base Polytope and Extreme Bases via Linear Orders

Recall that $EP_f = \{x \in \mathbb{R}^S \mid x(U) \leq f(U) \forall U \subseteq S\}$. We obtain the *base polytope* by adding the constraint $x(S) = f(S)$ to EP_f.

$$B_f = \{x \in \mathbb{R}^S \mid x(U) \leq f(U) \ \forall U \subseteq S, \ x(S) = f(S)\}.$$

A vector x in B_f is called a *base vector* or simply a base of EP_f (or of f). A *base vector* of f is a base vector of EP_f. Note that B_f is a face of EP_f. B_f is a polytope, since $f(\{s\}) \geq x_s = x(S) - x(S \setminus \{s\}) \geq f(S) - f(S \setminus \{s\})$ for each $s \in S$.

An extreme point of the base polytope is called an extreme base. What are the extreme bases? We note that the greedy algorithm for $\max\{wx \mid x \in EP_f\}$ generates a base whenever $w \geq 0$ (if $w(v) < 0$ for some v then the optimum value is unbounded). In fact the greedy algorithm uses w only to sort the elements and then ignores the weights. Thus, any two weight vectors that result in the same sorted order give rise to the same base. We set up notation for this. Let $L = v_1, v_2, \ldots, v_n$ be a total order on S, in other words a permutation of S. We say $u \prec_L v$ if u comes before v in L; we use $\preceq_L$ if u, v need not be distinct. Let $L(v)$ denote $\{u \in S \mid u \preceq_L v\}$. Given a total order L the greedy algorithm produces a base vector b_L where for each $v \in S$,

$$b_L(v) = f(L(v)) - f(L(v) \setminus \{v\}).$$

Lemma 4 *For each linear order L on S the vector b_L is an extreme base. Moreover, each extreme base x there is a linear order L (could be more than one) such that $x = b_L$.*

2.2 A Min-Max Theorem

Recall that the linear programming based algorithm for SFM was based on the following theorem of Edmonds.

Theorem 5 *For a submodular function $f : 2^S \to \mathbb{R}$ with $f(\emptyset) = 0$,*

$$\min_{U \subseteq S} f(U) = \max\{x(S) \mid x \in EP_f, x \leq 0\}.$$

A related theorem that one can prove from the above is the following. For a vector $z \in \mathbb{R}^S$ and $U \subseteq S$ we define $z^-(U)$ as $\sum_{v \in U : z(v) < 0} z(v)$. Alternatively, $z^-(v) = \min\{0, z(v)\}$.

Theorem 6 *For a submodular function $f : 2^S \to \mathbb{R}$ with $f(\emptyset) = 0$,*

$$\min_{U \subseteq S} f(U) = \max\{x^-(S) \mid x \in B_f\}.$$

We give direct proof of this which underlies the algorithmic aspects.

Proof: For any $x \in \mathbb{R}^S$ and $U \subseteq S$ we have $x^-(S) \leq x(U)$. If in addition $x \in B_f$ then $x^-(S) \leq x(U) \leq f(U)$. Since this holds for any $U \subseteq S$ we have that $\min_{U \subseteq S} f(U) \geq \max\{x^-(S) \mid x \in B_f\}$.

For the converse direction, let x be an optimum solution to $\max\{x^-(S) \mid x \in B_f\}$. Let $N = \{u \in S \mid x(u) < 0\}$ and $P = \{u \in S \mid x(u) > 0\}$. We observe that for any $v \in S \setminus (N \cup P)$, $x(v) = 0$. We say that a set U is tight with respect to x if $x(U) = f(U)$. Recall that tight sets uncross, in other words the set of all tight sets are closed under intersection and union.

Claim 7 *For any $u \in N$ and $v \in P$, there exists a tight set Y_{uv} where $u \in Y_{uv}$ and $v \notin Y_{uv}$.*

Assuming the claim above we finish the proof as follows. For $u \in N$, let $Y_u = \cap_{v \in P} Y_{uv}$. We note that Y_u is tight and $Y_u \cap P = \emptyset$. Let $Z = \cup_{u \in N} Y_u$. The set Z is tight and $N \subseteq Z$ and $Z \cap P = \emptyset$. Therefore, $x^-(S) = x(Z) = f(Z)$ and we are done.

Now we prove the claim by contradiction. Suppose it is not true. Then there is a $u \in N$ and $v \in P$ such that for all A where $u \in A$ and $v \notin A$ we have $x(A) < f(A)$. Let $\epsilon = \min\{f(A) - x(A) \mid u \in A, v \notin A\}$; we have $\epsilon > 0$. Let $\epsilon' = \min(\epsilon, |x(u)|, |x(v)|)$. We obtain a new vector $x' \in B_f$ as $x' = x + \epsilon'(\chi(u) - \chi(v))$, that is we add ϵ' to $x(u)$ and subtract ϵ' from $x(v)$. The new vector x' contradicts the optimality of x since $x'^-(S) > x^-(S)$. □

The above proof suggests the following definition.

Definition 8 *Given a vector $x \in B_f$ and $u, v \in S$, the* exchange capacity *of u, v with respect to x, denoted by $\alpha(x, v, u)$, is* $\min\{f(A) - x(A) \mid u \in A, v \notin A\}$.

A corollary that follows from the proof of Theorem 6.

Corollary 9 *A vector $x \in B_f$ is optimum for* $\max\{x^-(S) \mid x \in B_f\}$ *iff* $\alpha(x, v, u) = 0$ *for all $u \in N$ and $v \in P$ where $N = \{u \in S \mid x(u) < 0\}$ and $P = \{v \in S \mid x(v) > 0\}$.*

We remark that $\max\{x^-(S) \mid x \in B_f\}$ is not a linear optimization problem. The function $x^-(S)$ is a concave function (why?) and in particular the optimum solution need not be a vertex (in other words an extreme base) of B_f. See [1] for an illustrative example in two dimensions.

2.3 Towards an Algorithm

From Corollary 9 one imagines an algorithm that starts with arbitrary $x \in B_f$ (we can pick some arbitrary linear order L on S and set $x = b_L$) and improve $x^-(S)$ by finding a pair u, v with $u \in N$ and $v \in P$ with non-negative exchange capacity and improving as suggested in the proof of Theorem 6. However, this depends on our ability to compute $\alpha(x, v, u)$ and one sees from the definition that this is another submodular function minimization problem!

A second technical difficulty, as we mentioned earlier, is that the set of optimum solutions to $\max\{x^-(S) \mid x \in B_f\}$ may not contain an extreme base.

The general approach to overcoming these problems follows the work of Cunningham. Given $x \in B_f$ we express x as a convex combination of extreme bases (vertices of B_f); in fact, using Lemma 4, it is convenient to use linear orders as the implicit representation for an extreme base. Then we write $x = \sum_{L \in \Lambda} \lambda_L b_L$ where Λ is a collection of linear orders. By Caratheodary's theorem, Λ can be chosen such that $|\Lambda| \leq |S|$ since the dimension of B_f is $|S| - 1$. Although computing the exchange capacities with respect to an arbitrary $x \in B_f$ is difficult, if x is an extreme base b_L for a linear order L, then we see below that several natural exchanges can be efficiently computed. The goal would then to obtain exchanges for $x \in B_f$ by using exchanges for the linear orders in the convex combination for x given by Λ. Different algorithms take different approaches for this. See [1, 5], in particular [5] for detailed descriptions including intuition.

References

[1] L. Fleischer. Recent Progress in Submodular Function Minimization. OPTIMA: Mathematical Programming Society Newsletter , September 2000, no.64, 1-11. Available online at `http://www.mathprog.org/Optima-Issues/optima64.pdf`.

[2] S. Iwata and J. Orlin. A Simple Combinatorial Algorithm for Submodular Function Minimization. *Proc. of ACM-SIAM SODA*, 2009.

[3] L. Lovász. Submodular functions and convexity. Mathematical programming: the state of the art, Bonn, 235–257, 1982.

[4] A. Schrijver. *Combinatorial Optimization.* Springer-Verlag Berlin Heidelberg, 2003.

[5] Alexander Toshev. Submodular Function Minimization. Manuscript, January 2010. `http://www.seas.upenn.edu/~toshev/Site/About_Me_files/wpii-2.pdf`.