CS599: Convex and Combinatorial Optimization
Fall 2013
Lecture 24: Introduction to Submodular Functions

Instructor: Shaddin Dughmi

Announcements

Introduction

- We saw how matroids form a class of feasible sets over which optimization of modular objectives is tractable
- If matroids are discrete analogues of convex sets, then submodular functions are discrete analogues of convex/concave functions
  - Submodular functions behave like convex functions sometimes (minimization) and concave other times (maximization)
- Today we will introduce submodular functions, go through some examples, and mention some of their properties

Outline

Set Functions

- A set function takes as input a set, and outputs a real number
  - Inputs are subsets of some ground set X
  - $f : 2^X \to \mathbb{R}$
- We will focus on set functions where X is finite, and denote $n = |X|$

Set Functions

- A set function takes as input a set, and outputs a real number
  - Inputs are subsets of some ground set X
  - $f : 2^X \to \mathbb{R}$
- We will focus on set functions where X is finite, and denote $n = |X|$
- Equivalently: map points in the hypercube $\{0, 1\}^n$ to the real numbers
  - Can be plotted as 2^n points in $n + 1$ dimensional space

Set Functions

- We have already seen modular set functions
  - Associate a weight w_i with each $i \in X$, and set $f(S) = \sum_{i \in S} w_i$
  - Discrete analogue of linear functions

Set Functions

- We have already seen modular set functions
  - Associate a weight w_i with each $i \in X$, and set $f(S) = \sum_{i \in S} w_i$
  - Discrete analogue of linear functions
  - Direct definition of modularity: $f(A) + f(B) = f(A \cap B) + f(A \cup B)$

Set Functions

- We have already seen modular set functions
  - Associate a weight w_i with each $i \in X$, and set $f(S) = \sum_{i \in S} w_i$
  - Discrete analogue of linear functions
  - Direct definition of modularity: $f(A) + f(B) = f(A \cap B) + f(A \cup B)$
- Supmodular/supermodular functions are weak analogues to convex/concave functions (in no particular order!)

Set Functions

- We have already seen modular set functions
  - Associate a weight w_i with each $i \in X$, and set $f(S) = \sum_{i \in S} w_i$
  - Discrete analogue of linear functions
  - Direct definition of modularity: $f(A) + f(B) = f(A \cap B) + f(A \cup B)$
- Supmodular/supermodular functions are weak analogues to convex/concave functions (in no particular order!)
- Other possibly useful properties a set function may have:
  - Monotone increasing or decreasing
  - Nonnegative: $f(A) \geq 0$ for all $S \subseteq X$
  - Normalized: $f(\emptyset) = 0$.

Submodular Functions

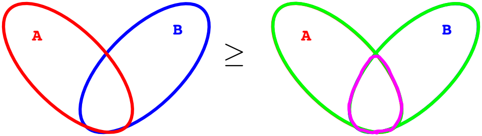
Definition 1

A set function $f : 2^X \to \mathbb{R}$ is submodular if and only if

$$f(A) + f(B) \geq f(A \cap B) + f(A \cup B)$$

for all $A, B \subseteq X$.

- "Uncrossing" two sets reduces their total function value



Submodular Functions

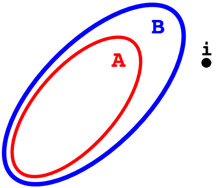
Definition 2

A set function $f : 2^X \to \mathbb{R}$ is submodular if and only if

$$f(B \cup \{i\}) - f(B) \leq f(A \cup \{i\}) - f(A))$$

for all $A \subseteq B \subseteq X$.

- The marginal value of an additional element exhibits "diminishing marginal returns"
- Should remind of concavity



Supermodular Functions

Definition 0

A set function $f : 2^X \to \mathbb{R}$ is supermodular if and only if $-f$ is submodular.

Supermodular Functions

Definition 0

A set function $f : 2^X \to \mathbb{R}$ is supermodular if and only if $-f$ is submodular.

Definition 1

A set function $f : 2^X \to \mathbb{R}$ is supermodular if and only if

$$f(A) + f(B) \leq f(A \cap B) + f(A \cup B)$$

for all $A, B \subseteq X$.

Supermodular Functions

Definition 0

A set function $f : 2^X \to \mathbb{R}$ is supermodular if and only if $-f$ is submodular.

Definition 1

A set function $f : 2^X \to \mathbb{R}$ is supermodular if and only if

$$f(A) + f(B) \leq f(A \cap B) + f(A \cup B)$$

for all $A, B \subseteq X$.

Definition 2

A set function $f : 2^X \to \mathbb{R}$ is supermodular if and only if

$$f(B \cup \{i\}) - f(B) \geq f(A \cup \{i\}) - f(A))$$

for all $A \subseteq B \subseteq X$.

Examples

Many common examples are monotone, normalized, and submodular. We mention some.

Coverage Functions

X is the left hand side of a graph, and $f(S)$ is the total number of neighbors of S.

- Can think of $i \in X$ as a set, and $f(S)$ as the total “coverage” of S.

Examples

Many common examples are monotone, normalized, and submodular. We mention some.

Coverage Functions

X is the left hand side of a graph, and $f(S)$ is the total number of neighbors of S.

- Can think of $i \in X$ as a set, and $f(S)$ as the total "coverage" of S.

Probability

X is a set of probability events, and $f(S)$ is the probability at least one of them occurs.

Examples

Social Influence

- X is the family of nodes in a social network
- A meme, idea, or product is adopted at a set of nodes S
- The idea propagates through the network through some random diffusion process
  - Many different models
- $f(S)$ is the expected number of nodes in the network which end up adopting the idea.

Examples

Social Influence

- X is the family of nodes in a social network
- A meme, idea, or product is adopted at a set of nodes S
- The idea propagates through the network through some random diffusion process
  - Many different models
- $f(S)$ is the expected number of nodes in the network which end up adopting the idea.

Utility Functions

When X is a set of goods, $f(S)$ can represent the utility of an agent for a bundle of these goods. Utilities which exhibit diminishing marginal returns are natural in many settings.

Examples

Entropy

X is a set of random events, and $f(S)$ is the entropy of events S.

Examples

Entropy

X is a set of random events, and $f(S)$ is the entropy of events S.

Matroid Rank

The rank function of a matroid is monotone, submodular, and normalized.

Examples

Entropy

X is a set of random events, and $f(S)$ is the entropy of events S.

Matroid Rank

The rank function of a matroid is monotone, submodular, and normalized.

Clustering Quality

X is the set of nodes in a graph G, and $f(S) = E(S)$ is the internal connectedness of cluster S.

- Supermodular

Examples

There are fewer examples of non-monotone submodular/supermodular functions, which are nontheless fundamental.

Graph Cuts

X is the set of nodes in a graph G, and $f(S)$ is the number of edges crossing the cut $(S, X \setminus S)$.

- Submodular
- Non-monotone.

Examples

There are fewer examples of non-monotone submodular/supermodular functions, which are nontheless fundamental.

Graph Cuts

X is the set of nodes in a graph G, and $f(S)$ is the number of edges crossing the cut $(S, X \setminus S)$.

- Submodular
- Non-monotone.

Graph Density

X is the set of nodes in a graph G, and $f(S) = \frac{E(S)}{|S|}$ where $E(S)$ is the number of edges with both endpoints in S.

- Non-monotone
- Neither submodular nor supermodular

Examples

There are fewer examples of non-monotone submodular/supermodular functions, which are nontheless fundamental.

Graph Cuts

X is the set of nodes in a graph G, and $f(S)$ is the number of edges crossing the cut $(S, X \setminus S)$.

- Submodular
- Non-monotone.

Graph Density

X is the set of nodes in a graph G, and $f(S) = \frac{E(S)}{|S|}$ where $E(S)$ is the number of edges with both endpoints in S.

- Non-monotone
- Neither submodular nor supermodular
- However, maximizing it reduces to maximizing supermodular function $E(S) - \alpha|S|$ for various $\alpha > 0$ (binary search)

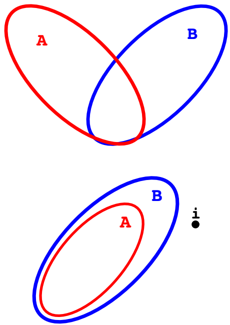
Equivalence of Both Definitions

Definition 1

$$f(A) + f(B) \geq f(A \cap B) + f(A \cup B)$$

Definition 2

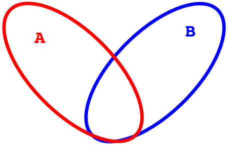
$$f(B \cup \{i\}) - f(B) \leq f(A \cup \{i\}) - f(A))$$



Equivalence of Both Definitions

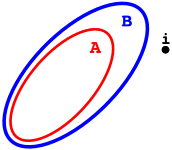
Definition 1

$$f(A) + f(B) \geq f(A \cap B) + f(A \cup B)$$



Definition 2

$$f(B \cup \{i\}) - f(B) \leq f(A \cup \{i\}) - f(A))$$



Definition 1 $\Rightarrow$ Definition 2

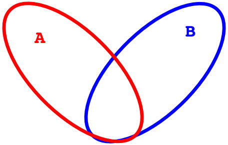
- To prove (2), let $A' = A \bigcup \{i\}$ and $B' = B$ and apply (1)

$$\begin{aligned} f(A \cup \{i\}) + f(B) &= f(A') + f(B') \\ &\geq f(A' \cap B') + f(A' \cup B') \\ &= f(A) + f(B \cup \{i\}) \end{aligned}$$

Equivalence of Both Definitions

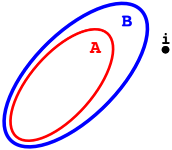
Definition 1

$$f(A)+f(B)\geq f(A\cap B)+f(A\cup B)$$



Definition 2

$$f(B\cup\{i\})-f(B)\leq f(A\cup\{i\})-f(A))$$



Definition 2 $\Rightarrow$ Definition 1

- To prove (1), start with $A = B$ and repeatedly elements to one but not the other
- At each step, (2) implies that the LHS of inequality (1) increases more than the RHS

Operations Preserving Submodularity

- Nonnegative-weighted combinations (a.k.a. conic combinations): If $f_1, \ldots, f_k$ are submodular, and $w_1, \ldots, w_k \geq 0$, then $g(S) = \sum_i w_i f_i(S)$ is also submodular
  - Special case: adding or subtracting a modular function

Operations Preserving Submodularity

- Nonnegative-weighted combinations (a.k.a. conic combinations): If $f_1, \ldots, f_k$ are submodular, and $w_1, \ldots, w_k \geq 0$, then $g(S) = \sum_i w_i f_i(S)$ is also submodular
  - Special case: adding or subtracting a modular function
- Restriction: If f is a submodular function on X, and $T \subseteq X$, then $g(S) = f(S \cap T)$ is submodular

Operations Preserving Submodularity

- Nonnegative-weighted combinations (a.k.a. conic combinations): If $f_1, \ldots, f_k$ are submodular, and $w_1, \ldots, w_k \geq 0$, then $g(S) = \sum_i w_i f_i(S)$ is also submodular
  - Special case: adding or subtracting a modular function
- Restriction: If f is a submodular function on X, and $T \subseteq X$, then $g(S) = f(S \cap T)$ is submodular
- Contraction (a.k.a conditioning): If f is a submodular function on X, and $T \subseteq X$, then $f_T(S) = f(S \cup T)$ is submodular

Operations Preserving Submodularity

- Nonnegative-weighted combinations (a.k.a. conic combinations): If $f_1, \ldots, f_k$ are submodular, and $w_1, \ldots, w_k \geq 0$, then $g(S) = \sum_i w_i f_i(S)$ is also submodular
  - Special case: adding or subtracting a modular function
- Restriction: If f is a submodular function on X, and $T \subseteq X$, then $g(S) = f(S \cap T)$ is submodular
- Contraction (a.k.a conditioning): If f is a submodular function on X, and $T \subseteq X$, then $f_T(S) = f(S \cup T)$ is submodular
- Reflection: If f is a submodular function on X, then $\overline{f}(S) = f(X \setminus S)$ is also submodular

Operations Preserving Submodularity

- Nonnegative-weighted combinations (a.k.a. conic combinations): If $f_1, \ldots, f_k$ are submodular, and $w_1, \ldots, w_k \geq 0$, then $g(S) = \sum_i w_i f_i(S)$ is also submodular
  - Special case: adding or subtracting a modular function
- Restriction: If f is a submodular function on X, and $T \subseteq X$, then $g(S) = f(S \cap T)$ is submodular
- Contraction (a.k.a conditioning): If f is a submodular function on X, and $T \subseteq X$, then $f_T(S) = f(S \cup T)$ is submodular
- Reflection: If f is a submodular function on X, then $\overline{f}(S) = f(X \setminus S)$ is also submodular
- Others: Dilworth trucation, convolution with modular functions, ...

Operations Preserving Submodularity

- Nonnegative-weighted combinations (a.k.a. conic combinations): If $f_1, \ldots, f_k$ are submodular, and $w_1, \ldots, w_k \geq 0$, then $g(S) = \sum_i w_i f_i(S)$ is also submodular
  - Special case: adding or subtracting a modular function
- Restriction: If f is a submodular function on X, and $T \subseteq X$, then $g(S) = f(S \cap T)$ is submodular
- Contraction (a.k.a conditioning): If f is a submodular function on X, and $T \subseteq X$, then $f_T(S) = f(S \cup T)$ is submodular
- Reflection: If f is a submodular function on X, then $\overline{f}(S) = f(X \setminus S)$ is also submodular
- Others: Dilworth trucation, convolution with modular functions, ...

Note

The minimum or maximum of two submodular functions is not necessarily submodular

Optimizing Submodular Functions

- As our examples suggest, optimization problems involving submodular functions are very common
- These can be classified on two axes: constrained/unconstrained and maximization/minimization

	Maximization	Minimization
Unconstrained	NP-hard $\frac{1}{2}$ approximation	Polynomial time via convex opt
Constrained	Usually NP-hard $1 - 1/e$ (mono, matroid) $O(1)$ ("nice" constriants)	Usually NP-hard to apx. Few easy special cases

Optimizing Submodular Functions

- As our examples suggest, optimization problems involving submodular functions are very common
- These can be classified on two axes: constrained/unconstrained and maximization/minimization

	Maximization	Minimization
Unconstrained	NP-hard $\frac{1}{2}$ approximation	Polynomial time via convex opt
Constrained	Usually NP-hard $1 - 1/e$ (mono, matroid) $O(1)$ (“nice” constriants)	Usually NP-hard to apx. Few easy special cases

Optimizing Submodular Functions

- As our examples suggest, optimization problems involving submodular functions are very common
- These can be classified on two axes: constrained/unconstrained and maximization/minimization

	Maximization	Minimization
Unconstrained	NP-hard $\frac{1}{2}$ approximation	Polynomial time via convex opt
Constrained	Usually NP-hard $1 - 1/e$ (mono, matroid) $O(1)$ ("nice" constriants)	Usually NP-hard to apx. Few easy special cases

Representation

In order to generalize all our examples, algorithmic results are often posed in the value oracle model. Namely, we only assume we have access to a subroutine evaluating $f(S)$.