Submodular Functions, Optimization, and Applications to Machine Learning

— Fall Quarter, Lecture 19 —

http://www.ee.washington.edu/people/faculty/bilmes/classes/ee563_spring_2018/

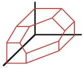
Prof. Jeff Bilmes

University of Washington, Seattle
Department of Electrical Engineering
http://melodi.ee.washington.edu/~bilmes

Dec 7th, 2020

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

$= f(A_r) + 2f(C) + f(B_r)$ $\quad = f(A_r) + f(C) + f(B_r)$ $\quad = f(A \cap B)$



Class Road Map - EE563

- L1(9/30): Motivation, Applications, Definitions, Properties
- L2(10/5): Sums concave(modular), uses (diversity/costs, feature selection), information theory
- L3(10/7): Monge, More Definitions, Graph and Combinatorial Examples,
- L4(10/12): Graph & Combinatorial Examples, Matrix Rank, Properties, Other Defs, Independence
- L5(10/14): Properties, Defs of Submodularity, Independence
- L6(10/19): Matroids, Matroid Examples, Matroid Rank,
- L7(10/21): Matroid Rank, More on Partition Matroid, Laminar Matroids, System of Distinct Reps, Transversals
- L8(10/26): Transversal Matroid, Matroid and representation, Dual Matroid
- L9(10/28): Other Matroid Properties, Combinatorial Geometries, Matroid and Greedy, Polyhedra, Matroid Polytopes
- L10(11/2): Matroid Polytopes, Matroids $\rightarrow$ Polymatroids
- L11(11/4): Matroids $\rightarrow$ Polymatroids, Polymatroids
- L12(11/9): Polymatroids, Polymatroids and Greedy
- L–(11/11): Veterans Day, Holiday
- L13(11/16): Polymatroids and Greedy, Possible Polytopes, Extreme Points, Cardinality Constrained Maximization
- L14(11/18): Cardinality Constrained Maximization, Curvature
- L15(11/23): Curvature, Submodular Max w. Other Constraints, Start Cont. Extensions
- L16(11/25): Submodular Max w. Other Constraints, Cont. Extensions, Lovász extension
- L17(11/30): Choquet Integration, Non-linear Measure/Aggregation, Definitions/Properties, Examples.
- L18(12/2): Multilinear Extension, Submodular Max/polyhedral, Most Violated Ineq., Matroids Closure/Sat
- L19(12/7): Fund. Circuit/Dep, SFM, L.E. primal, Start SFM via Min-Norm Point
- L20(12/9):
- L21(12/14): final meeting (presentations) maximization.

Last day of instruction, Fri. Dec 11th. Finals Week: Dec 12-18, 2020

Rest of class

- Homework 4 posted, due Thursday Dec 17th, 2020, 11:55pm.
- Final project paper proposal, was due Sunday Dec 6th, 11:59pm.
- Final project 4-page paper and presentation slides, due Sunday Dec 13th, 11:59pm.
- Final project presentation, Monday Dec 14th, starting at 10:30am.
- Final project: Read and present a recent (past 5 years) paper on submodular/supermodular optimization. Paper should have both a theoretical and practical component. What is due: (1) 4-page paper summary, and (2) 10 minute presentation about the paper, will be giving presentations on Monday 12/14/2020. You must choose your paper before the 14th (this will be HW5), and you must turn in your slides and 4-page paper (this will be HW6).
- Recall, grades will be based on a combination of a final project (40%) and the four homeworks (60%).

Most violated inequality problem in matroid polytope case

- Consider

$$P_r^+ = \{x \in \mathbb{R}^E : x \geq 0, x(A) \leq r_M(A), \forall A \subseteq E\} \tag{19.19}$$

- Suppose we have any $x \in \mathbb{R}_+^E$ such that $x \notin P_r^+$.
- Hence, there must be a set of $\mathcal{W} \subseteq 2^V$, each member of which corresponds to a violated inequality, i.e., equations of the form $x(A) > r_M(A)$ for $A \in \mathcal{W}$.
- The most violated inequality when x is considered w.r.t. P_r^+ corresponds to the set A that maximizes $x(A) - r_M(A)$, i.e., the most violated inequality is valuated as:

$$\max \{x(A) - r_M(A) : A \in \mathcal{W}\} = \max \{x(A) - r_M(A) : A \subseteq E\} \tag{19.20}$$

- Since x is modular and $x(E \setminus A) = x(E) - x(A)$, we can express this via a min as in;:

$$\min \{r_M(A) + x(E \setminus A) : A \subseteq E\} \tag{19.21}$$

Most violated inequality/polymatroid membership/SFM

- The most violated inequality when x is considered w.r.t. P_f^+ corresponds to the set A that maximizes $x(A) - f(A)$, i.e., the most violated inequality is valuated as:

$$\max \{x(A) - f(A) : A \in \mathcal{W}\} = \max \{x(A) - f(A) : A \subseteq E\} \tag{19.19}$$

- Since x is modular and $x(E \setminus A) = x(E) - x(A)$, we can express this via a min as in;:

$$\min \{f(A) + x(E \setminus A) : A \subseteq E\} \tag{19.20}$$

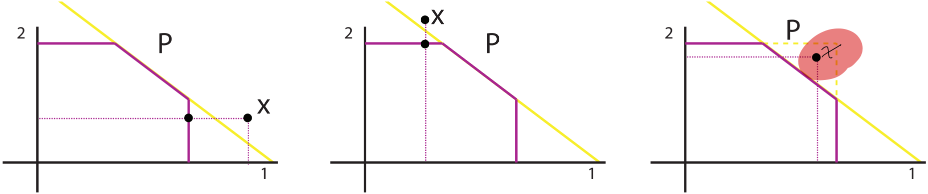
- More importantly, $\min \{f(A) + x(E \setminus A) : A \subseteq E\}$ is a form of submodular function minimization, namely $\min \{f(A) - x(A) : A \subseteq E\}$ for a submodular f and $x \in \mathbb{R}_+^E$, consisting of a difference of polymatroid and modular function (so $f - x$ is no longer necessarily monotone, nor positive).
- We will ultimately answer how general this form of SFM is.

Most violated inequality/polymatroid membership/SFM

- Consider

$$P_f^+ = \{x \in \mathbb{R}^E : x \geq 0, x(A) \leq f(A), \forall A \subseteq E\} \tag{19.19}$$

- Suppose we have any $x \in \mathbb{R}_+^E$ such that $x \notin P_f^+$, most violated inequality is based on set A that solves $\min\{f(A) - x(A) : A \subseteq E\}$ or $\min\{f(A) + x(E \setminus A) : A \subseteq E\}$
- Hence, there must be a set of $\mathcal{W} \subseteq 2^V$, each member of which corresponds to a violated inequality, i.e., equations of the form $x(A) > r_M(A)$ for $A \in \mathcal{W}$.



$\mathcal{W} = \{\{1\}, \{1,2\}\}$ $\mathcal{W} = \{\{2\}, \{1,2\}\}$ $\mathcal{W} = \{\{1,2\}\}$

Fundamental circuits in matroids

Lemma 19.2.9

Let $I \in \mathcal{I}(M)$, and $e \in E$, then $I \cup \{e\}$ contains at most one circuit in M.

Proof.

- Suppose, to the contrary, that there are two distinct circuits C_1, C_2 such that $(C_1 \cup C_2) \subseteq I \cup \{e\}$.
- Then $e \in C_1 \cap C_2$, and by (C2), there is a circuit C_3 of M s.t. $C_3 \subseteq (C_1 \cup C_2) \setminus \{e\} \subseteq I$
- This contradicts the independence of I.

$\square$

In general, let $C(I, e)$ be the unique circuit associated with $I \cup \{e\}$ (commonly called the fundamental circuit in M w.r.t. I and e).

Matroids: The Fundamental Circuit

- Define $C(I, e)$ be the unique circuit associated with $I \cup \{e\}$ (the fundamental circuit in M w.r.t. I and e, if it exists).
- If $e \in \text{span}(I) \setminus I$, then $C(I, e)$ is well defined ($I + e$ creates one circuit).
- If $e \in I$, then $I + e = I$ doesn't create a circuit. In such cases, $C(I, e)$ is not really defined.
- In such cases, we define $C(I, e) = \{e\}$, and we will soon see why.
- If $e \notin \text{span}(I)$ (i.e., when $I + e$ is independent), then we set $C(I, e) = \emptyset$.

The sat function = Polymatroid Closure

- In a matroid, closure (span) of a set A are all items that A spans (eq. that depend on A).
- We wish to generalize closure to polymatroids.
- Consider $x \in P_f$ for polymatroid function f.
- Again, recall, tight sets are closed under union and intersection, and therefore form a distributive lattice.
- That is, we saw in Lecture 11 that for any $A, B \in \mathcal{D}(x)$, we have that $A \cup B \in \mathcal{D}(x)$ and $A \cap B \in \mathcal{D}(x)$, which can constitute a join and meet.
- Recall, for a given $x \in P_f$, we have defined this tight family as

$$\mathcal{D}(x) = \{A : A \subseteq E, x(A) = f(A)\} \tag{19.20}$$

and

$$\text{sat}(x) \stackrel{\text{def}}{=} \bigcup \{A : A \in \mathcal{D}(x)\} \tag{19.21}$$

Minimizers of a Submodular Function form a lattice

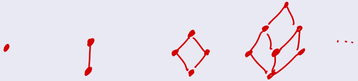
Theorem 19.2.10

For arbitrary submodular f, the minimizers are closed under union and intersection. That is, let $\mathcal{M} = \operatorname{argmin}_{X \subseteq E} f(X)$ be the set of minimizers of f. Let $A, B \in \mathcal{M}$. Then $A \cup B \in \mathcal{M}$ and $A \cap B \in \mathcal{M}$.

Proof.

Since A and B are minimizers, we have $f(A) = f(B) \le f(A \cap B)$ and $f(A) = f(B) \le f(A \cup B)$.

By submodularity, we have



$$f(A) + f(B) \ge f(A \cup B) + f(A \cap B) \tag{19.22}$$

Hence, we must have $f(A) = f(B) = f(A \cup B) = f(A \cap B)$. □

Thus, the minimizers of a submodular function form a lattice, and there is a maximal and a minimal minimizer of every submodular function.

The sat function = Polymatroid Closure

- Matroid closure is generalized by the unique maximal element in $\mathcal{D}(x)$, also called the polymatroid closure or sat (saturation function).
- For some $x \in P_f$, we have defined:

$$\text{cl}(x) \stackrel{\text{def}}{=} \text{sat}(x) \stackrel{\text{def}}{=} \bigcup \{A : A \in \mathcal{D}(x)\} \tag{19.22}$$

$$= \bigcup \{A : A \subseteq E, x(A) = f(A)\} \tag{19.23}$$

$$= \{e : e \in E, \forall \alpha > 0, x + \alpha \mathbf{1}_e \notin P_f\} \tag{19.24}$$

- Hence, $\text{sat}(x)$ is the maximal (zero-valued) minimizer of the submodular function $f_x(A) \triangleq f(A) - x(A)$.
- Eq. (**??**) says that sat consists of elements of E for point x that are P_f saturated (any additional positive movement, in that dimension, leaves P_f). We'll revisit this in a few slides.
- First, we see how sat generalizes matroid closure.

The sat function = Polymatroid Closure

Lemma 19.2.10 (Matroid $\text{sat} : \mathbb{R}_+^E \to 2^E$ is the same as closure.)

For $I \in \mathcal{I}$, *we have* $\text{sat}(\mathbf{1}_I) = \text{span}(I)$ (19.26)

Proof.

- For $\mathbf{1}_I(I) = |I| = r(I)$, so $I \in \mathcal{D}(\mathbf{1}_I)$ and $I \subseteq \text{sat}(\mathbf{1}_I)$. Also, $I \subseteq \text{span}(I)$.
- Consider some $b \in \text{span}(I) \setminus I$.
- Then $I \cup \{b\} \in \mathcal{D}(\mathbf{1}_I)$ since $\mathbf{1}_I(I \cup \{b\}) = |I| = r(I \cup \{b\}) = r(I)$.
- Thus, $b \in \text{sat}(\mathbf{1}_I)$.
- Therefore, $\text{sat}(\mathbf{1}_I) \supseteq \text{span}(I)$.

...

Saturation Capacity

- The max is achieved when

$$\alpha = \hat{c}(x;e) \stackrel{\text{def}}{=} \min\{f(A) - x(A), \forall A \supseteq \{e\}\} \tag{19.43}$$

- $\hat{c}(x;e)$ is known as the saturation capacity associated with $x \in P_f$ and e.
- Thus we have for $x \in P_f$,

$$\hat{c}(x;e) \stackrel{\text{def}}{=} \min\{f(A) - x(A), \forall A \ni e\} \tag{19.44}$$

$$= \max\{\alpha : \alpha \in \mathbb{R}, x + \alpha \mathbf{1}_e \in P_f\} \tag{19.45}$$

- We immediately see that for $e \in E \setminus \text{sat}(x)$, we have that $\hat{c}(x;e) > 0$.
- Also, we have that: $e \in \text{sat}(x) \Leftrightarrow \hat{c}(x;e) = 0$.
- Note that any α with $0 \leq \alpha \leq \hat{c}(x;e)$ we have $x + \alpha \mathbf{1}_e \in P_f$.
- We also see that computing $\hat{c}(x;e)$ is a form of submodular function minimization.

Dependence Function

more tools

- Tight sets can be restricted to contain a particular element.

Dependence Function

- Tight sets can be restricted to contain a particular element.
- Given $x \in P_f$, and $e \in \text{sat}(x)$, define

$$\mathcal{D}(x, e) = \{A : e \in A \subseteq E, x(A) = f(A)\} \tag{19.1}$$

$$= \mathcal{D}(x) \cap \{A : A \subseteq E, e \in A\} \tag{19.2}$$

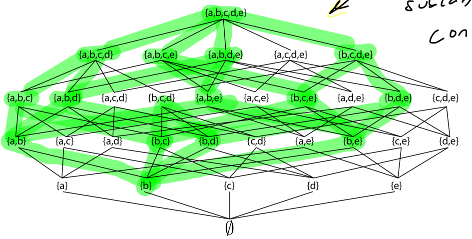
Dependence Function

- Tight sets can be restricted to contain a particular element.
- Given $x \in P_f$, and $e \in \text{sat}(x)$, define

$$\mathcal{D}(x, e) = \{A : e \in A \subseteq E, x(A) = f(A)\} \tag{19.1}$$
$$= \mathcal{D}(x) \cap \{A : A \subseteq E, e \in A\} \tag{19.2}$$

- Thus, $\mathcal{D}(x, e) \subseteq \mathcal{D}(x)$, and $\mathcal{D}(x, e)$ is a sublattice of $\mathcal{D}(x)$.

green is sublattice that contains b.



Dependence Function

- Tight sets can be restricted to contain a particular element.
- Given $x \in P_f$, and $e \in \text{sat}(x)$, define

$$\mathcal{D}(x,e) = \{A : e \in A \subseteq E, x(A) = f(A)\} \tag{19.1}$$
$$= \mathcal{D}(x) \cap \{A : A \subseteq E, e \in A\} \tag{19.2}$$

- Thus, $\mathcal{D}(x,e) \subseteq \mathcal{D}(x)$, and $\mathcal{D}(x,e)$ is a sublattice of $\mathcal{D}(x)$.
- Therefore, we can define a unique minimal element of $\mathcal{D}(x,e)$ denoted as follows:

$$\text{dep}(x,e) = \begin{cases} \bigcap \{A : e \in A \subseteq E, x(A) = f(A)\} & \text{if } e \in \text{sat}(x) \\ \emptyset & \text{else} \end{cases} \tag{19.3}$$

Dependence Function

- Tight sets can be restricted to contain a particular element.
- Given $x \in P_f$, and $e \in \text{sat}(x)$, define

$$\mathcal{D}(x,e) = \{A : e \in A \subseteq E, x(A) = f(A)\} \quad (19.1)$$
$$= \mathcal{D}(x) \cap \{A : A \subseteq E, e \in A\} \quad (19.2)$$

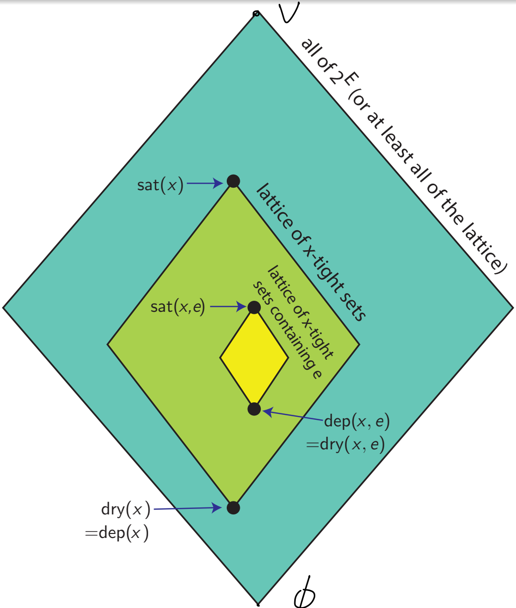
- Thus, $\mathcal{D}(x,e) \subseteq \mathcal{D}(x)$, and $\mathcal{D}(x,e)$ is a sublattice of $\mathcal{D}(x)$.
- Therefore, we can define a unique minimal element of $\mathcal{D}(x,e)$ denoted as follows:

$$\text{dep}(x,e) = \begin{cases} \bigcap \{A : e \in A \subseteq E, x(A) = f(A)\} & \text{if } e \in \text{sat}(x) \\ \emptyset & \text{else} \end{cases} \quad (19.3)$$

- I.e., $\text{dep}(x,e)$ is the minimal element in $\mathcal{D}(x)$ that contains e (the minimal x-tight set containing e). The necessary elements for e-containing tightness.

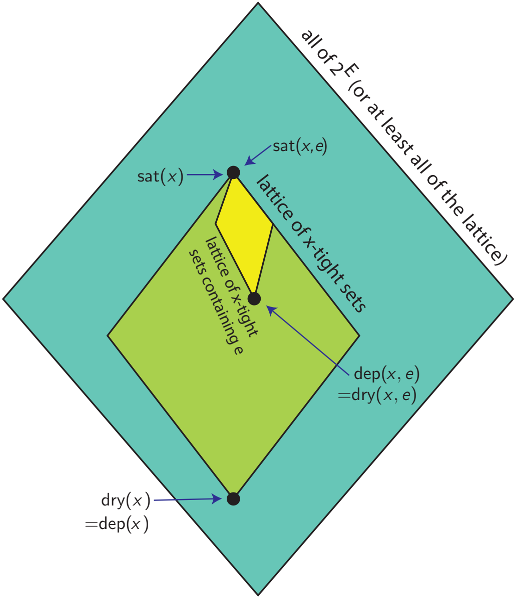
dep and sat in a lattice

- Given some $x \in P_f$,
- The picture on the right summarizes the relationships between the lattices and sublattices.
- Note, $\text{dep}(x, e) \supseteq \text{dep}(x) = \bigcap \{A : x(A) = f(A)\}$.



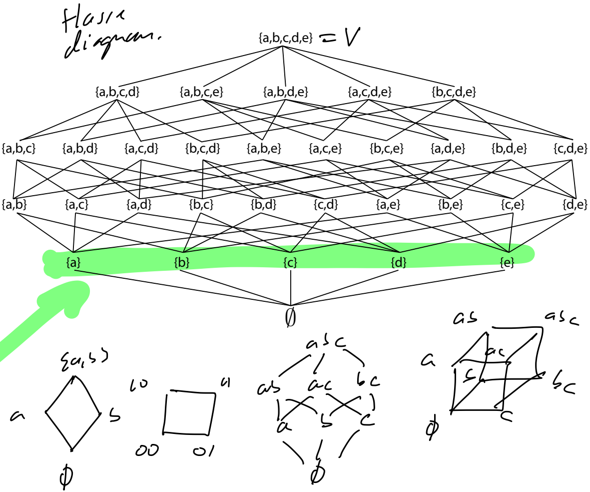
dep and sat in a lattice

- Given some $x \in P_f$,
- The picture on the right summarizes the relationships between the lattices and sublattices.
- Note, $\text{dep}(x, e) \supseteq \text{dep}(x) = \bigcap \{A : x(A) = f(A)\}$.
- In fact, $\text{sat}(x, e) = \text{sat}(x)$. Why?



dep and sat in a lattice

- Given some $x \in P_f$,
- The picture on the right summarizes the relationships between the lattices and sublattices.
- Note, $\text{dep}(x, e) \supseteq \text{dep}(x) = \bigcap \{A : x(A) = f(A)\}$.
- In fact, $\text{sat}(x, e) = \text{sat}(x)$. Why?
- Example lattice on 5 elements.



dep and sat in a lattice

- Given $x \in P_f$, recall distributive lattice of tight sets $\mathcal{D}(x) = \{A : x(A) = f(A)\}$

dep and sat in a lattice

- Given $x \in P_f$, recall distributive lattice of tight sets $\mathcal{D}(x) = \{A : x(A) = f(A)\}$
- We had that $\text{sat}(x) = \bigcup \{A : A \in \mathcal{D}(x)\}$ is the "1" element of this lattice.

dep and sat in a lattice

- Given $x \in P_f$, recall distributive lattice of tight sets $\mathcal{D}(x) = \{A : x(A) = f(A)\}$
- We had that $\text{sat}(x) = \bigcup \{A : A \in \mathcal{D}(x)\}$ is the "1" element of this lattice.
- Consider the "0" element of $\mathcal{D}(x)$, i.e., $\text{dry}(x) \stackrel{\text{def}}{=} \bigcap \{A : A \in \mathcal{D}(x)\}$

dep and sat in a lattice

- Given $x \in P_f$, recall distributive lattice of tight sets $\mathcal{D}(x) = \{A : x(A) = f(A)\}$
- We had that $\mathrm{sat}(x) = \bigcup \{A : A \in \mathcal{D}(x)\}$ is the "1" element of this lattice.
- Consider the "0" element of $\mathcal{D}(x)$, i.e., $\mathrm{dry}(x) \stackrel{\text{def}}{=} \bigcap \{A : A \in \mathcal{D}(x)\}$
- We can see $\mathrm{dry}(x)$ as the elements that are necessary for tightness.

dep and sat in a lattice

- Given $x \in P_f$, recall distributive lattice of tight sets $\mathcal{D}(x) = \{A : x(A) = f(A)\}$
- We had that $\text{sat}(x) = \bigcup \{A : A \in \mathcal{D}(x)\}$ is the "1" element of this lattice.
- Consider the "0" element of $\mathcal{D}(x)$, i.e., $\text{dry}(x) \stackrel{\text{def}}{=} \bigcap \{A : A \in \mathcal{D}(x)\}$
- We can see $\text{dry}(x)$ as the elements that are necessary for tightness.
- That is, we can equivalently define $\text{dry}(x)$ as

$x \in P_f$

$$\text{dry}(x) = \{e' : x(A) < f(A), \forall A \not\ni e'\} \tag{19.4}$$

dep and sat in a lattice

- Given $x \in P_f$, recall distributive lattice of tight sets $\mathcal{D}(x) = \{A : x(A) = f(A)\}$
- We had that $\text{sat}(x) = \bigcup \{A : A \in \mathcal{D}(x)\}$ is the "1" element of this lattice.
- Consider the "0" element of $\mathcal{D}(x)$, i.e., $\text{dry}(x) \stackrel{\text{def}}{=} \bigcap \{A : A \in \mathcal{D}(x)\}$
- We can see $\text{dry}(x)$ as the elements that are necessary for tightness.
- That is, we can equivalently define $\text{dry}(x)$ as

$$\text{dry}(x) = \left\{e' : x(A) < f(A), \forall A \not\ni e'\right\} \tag{19.4}$$

- This can be read as, for any $e' \in \text{dry}(x)$, any set that does not contain e' is not tight for x (any set A that is missing any element of $\text{dry}(x)$ is not tight).

dep and sat in a lattice

- Given $x \in P_f$, recall distributive lattice of tight sets $\mathcal{D}(x) = \{A : x(A) = f(A)\}$
- We had that $\text{sat}(x) = \bigcup \{A : A \in \mathcal{D}(x)\}$ is the "1" element of this lattice.
- Consider the "0" element of $\mathcal{D}(x)$, i.e., $\text{dry}(x) \stackrel{\text{def}}{=} \bigcap \{A : A \in \mathcal{D}(x)\}$
- We can see $\text{dry}(x)$ as the elements that are necessary for tightness.
- That is, we can equivalently define $\text{dry}(x)$ as

$$\text{dry}(x) = \left\{e' : x(A) < f(A), \forall A \not\ni e'\right\} \tag{19.4}$$

- This can be read as, for any $e' \in \text{dry}(x)$, any set that does not contain e' is not tight for x (any set A that is missing any element of $\text{dry}(x)$ is not tight).
- Perhaps, then, a better name for dry is $\text{ntight}(x)$, for the necessary for tightness (but we'll actually use neither name).

dep and sat in a lattice

- Given $x \in P_f$, recall distributive lattice of tight sets $\mathcal{D}(x) = \{A : x(A) = f(A)\}$
- We had that $\text{sat}(x) = \bigcup \{A : A \in \mathcal{D}(x)\}$ is the "1" element of this lattice.
- Consider the "0" element of $\mathcal{D}(x)$, i.e., $\text{dry}(x) \stackrel{\text{def}}{=} \bigcap \{A : A \in \mathcal{D}(x)\}$
- We can see $\text{dry}(x)$ as the elements that are necessary for tightness.
- That is, we can equivalently define $\text{dry}(x)$ as

$$\text{dry}(x) = \left\{e' : x(A) < f(A), \forall A \not\ni e'\right\} \tag{19.4}$$

- This can be read as, for any $e' \in \text{dry}(x)$, any set that does not contain e' is not tight for x (any set A that is missing any element of $\text{dry}(x)$ is not tight).
- Perhaps, then, a better name for dry is ntight(x), for the necessary for tightness (but we'll actually use neither name).
- Note that dry need not be the empty set. Exercise: give example.

e-containing dep and sat

- Now, given $x \in P_f$, and $e \in \text{sat}(x)$, recall distributive sub-lattice of e-containing tight sets $\mathcal{D}(x,e) = \{A : e \in A, x(A) = f(A)\}$
- We can define the "1" element of this sub-lattice as $\text{sat}(x,e) \stackrel{\text{def}}{=} \bigcup \{A : A \in \mathcal{D}(x,e)\}$.
- Analogously, we can define the "0" element of this sub-lattice as $\text{dry}(x,e) \stackrel{\text{def}}{=} \bigcap \{A : A \in \mathcal{D}(x,e)\}$ = ntight(x,e)
- We can see $\text{dry}(x,e)$ as the elements that are necessary for e-containing tightness, with $e \in \text{sat}(x)$.
- That is, we can view $\text{dry}(x,e)$ as

$$\text{dep}(x,e) = \text{dry}(x,e) = \{e' : x(A) < f(A), \forall A \not\ni e', e \in A\} \tag{19.5}$$

- This can be read as, for any $e' \in \text{dry}(x,e)$, any e-containing set that does not contain e' is not tight for x. Could call it ntight(x,e), necessary elements for e-containing tightness.
- But actually, $\text{dry}(x,e) = \text{dep}(x,e)$, so we have derived another expression for $\text{dep}(x,e)$ in Eq. (19.5).

Dependence Function and Fundamental Matroid Circuit

- Now, let $(E, \mathcal{I}) = (E, r)$ be a matroid, and let $I \in \mathcal{I}$ giving $\mathbf{1}_I \in P_r$. We have $\text{sat}(\mathbf{1}_I) = \text{span}(I) = \text{closure}(I)$.

Dependence Function and Fundamental Matroid Circuit

- Now, let $(E, \mathcal{I}) = (E, r)$ be a matroid, and let $I \in \mathcal{I}$ giving $\mathbf{1}_I \in P_r$. We have $\text{sat}(\mathbf{1}_I) = \text{span}(I) = \text{closure}(I)$.
- Suppose $e \in \text{sat}(\mathbf{1}_I) \setminus I$, then consider an $A \ni e$ with $|I \cap A| = r(A)$.

Dependence Function and Fundamental Matroid Circuit

- Now, let $(E, \mathcal{I}) = (E, r)$ be a matroid, and let $I \in \mathcal{I}$ giving $\mathbf{1}_I \in P_r$. We have $\text{sat}(\mathbf{1}_I) = \text{span}(I) = \text{closure}(I)$.
- Suppose $e \in \text{sat}(\mathbf{1}_I) \setminus I$, then consider an $A \ni e$ with $|I \cap A| = r(A)$.
- Then $I \cap A$ serves as a base for A (i.e., $I \cap A$ spans A) and any such A contains a circuit (i.e., we can add $e \in A \setminus I$ to $I \cap A$ w/o increasing rank).

Dependence Function and Fundamental Matroid Circuit

- Now, let $(E, \mathcal{I}) = (E, r)$ be a matroid, and let $I \in \mathcal{I}$ giving $\mathbf{1}_I \in P_r$. We have $\text{sat}(\mathbf{1}_I) = \text{span}(I) = \text{closure}(I)$.
- Suppose $e \in \text{sat}(\mathbf{1}_I) \setminus I$, then consider an $A \ni e$ with $|I \cap A| = r(A)$.
- Then $I \cap A$ serves as a base for A (i.e., $I \cap A$ spans A) and any such A contains a circuit (i.e., we can add $e \in A \setminus I$ to $I \cap A$ w/o increasing rank).
- Given $e \in \text{sat}(\mathbf{1}_I) \setminus I$, and consider $\text{dep}(\mathbf{1}_I, e)$, with

$$\text{dep}(\mathbf{1}_I, e) = \bigcap \{A : e \in A \subseteq E, \mathbf{1}_I(A) = r(A)\} \tag{19.6}$$

$$= \bigcap \{A : e \in A \subseteq E, |I \cap A| = r(A)\} \tag{19.7}$$

$$= \bigcap \{A : e \in A \subseteq E, r(A) - |I \cap A| = 0\} \tag{19.8}$$

Dependence Function and Fundamental Matroid Circuit

- Now, let $(E, \mathcal{I}) = (E, r)$ be a matroid, and let $I \in \mathcal{I}$ giving $\mathbf{1}_I \in P_r$. We have $\text{sat}(\mathbf{1}_I) = \text{span}(I) = \text{closure}(I)$.
- Suppose $e \in \text{sat}(\mathbf{1}_I) \setminus I$, then consider an $A \ni e$ with $|I \cap A| = r(A)$.
- Then $I \cap A$ serves as a base for A (i.e., $I \cap A$ spans A) and any such A contains a circuit (i.e., we can add $e \in A \setminus I$ to $I \cap A$ w/o increasing rank).
- Given $e \in \text{sat}(\mathbf{1}_I) \setminus I$, and consider $\text{dep}(\mathbf{1}_I, e)$, with

$$\text{dep}(\mathbf{1}_I, e) = \bigcap \{A : e \in A \subseteq E, \mathbf{1}_I(A) = r(A)\} \tag{19.6}$$

$$= \bigcap \{A : e \in A \subseteq E, |I \cap A| = r(A)\} \tag{19.7}$$

$$= \bigcap \{A : e \in A \subseteq E, r(A) - |I \cap A| = 0\} \tag{19.8}$$

- By SFM lattice, $\exists$ a unique minimal $A \ni e$ with $|I \cap A| = r(A)$.

Dependence Function and Fundamental Matroid Circuit

- Now, let $(E,\mathcal{I}) = (E,r)$ be a matroid, and let $I \in \mathcal{I}$ giving $\mathbf{1}_I \in P_r$. We have $\text{sat}(\mathbf{1}_I) = \text{span}(I) = \text{closure}(I)$.
- Suppose $e \in \text{sat}(\mathbf{1}_I) \setminus I$, then consider an $A \ni e$ with $|I \cap A| = r(A)$.
- Then $I \cap A$ serves as a base for A (i.e., $I \cap A$ spans A) and any such A contains a circuit (i.e., we can add $e \in A \setminus I$ to $I \cap A$ w/o increasing rank).
- Given $e \in \text{sat}(\mathbf{1}_I) \setminus I$, and consider $\text{dep}(\mathbf{1}_I, e)$, with

$$\text{dep}(\mathbf{1}_I, e) = \bigcap \{A : e \in A \subseteq E, \mathbf{1}_I(A) = r(A)\} \tag{19.6}$$

$$= \bigcap \{A : e \in A \subseteq E, |I \cap A| = r(A)\} \tag{19.7}$$

$$= \bigcap \{A : e \in A \subseteq E, r(A) - |I \cap A| = 0\} \tag{19.8}$$

- By SFM lattice, $\exists$ a unique minimal $A \ni e$ with $|I \cap A| = r(A)$.
- Thus, $\text{dep}(\mathbf{1}_I, e)$ must be a circuit since if it included more than a circuit, it would not be minimal in this sense.

Dependence Function and Fundamental Matroid Circuit

- Therefore, when $e \in \text{sat}(\mathbf{1}_I) \setminus I$, then $\text{dep}(\mathbf{1}_I, e) = C(I, e)$ where $C(I, e)$ is the unique circuit contained in $I + e$ in a matroid (the fundamental circuit of e and I that we encountered before).

Dependence Function and Fundamental Matroid Circuit

- Therefore, when $e \in \text{sat}(\mathbf{1}_I) \setminus I$, then $\text{dep}(\mathbf{1}_I, e) = C(I, e)$ where $C(I, e)$ is the unique circuit contained in $I + e$ in a matroid (the fundamental circuit of e and I that we encountered before).
- Now, if $e \in \text{sat}(\mathbf{1}_I) \cap I$ with $I \in \mathcal{I}$, we said that $C(I, e)$ was undefined (since no circuit is created in this case) and so we defined it as $C(I, e) = \{e\}$

Dependence Function and Fundamental Matroid Circuit

- Therefore, when $e \in \text{sat}(\mathbf{1}_I) \setminus I$, then $\text{dep}(\mathbf{1}_I, e) = C(I, e)$ where $C(I, e)$ is the unique circuit contained in $I + e$ in a matroid (the fundamental circuit of e and I that we encountered before).
- Now, if $e \in \text{sat}(\mathbf{1}_I) \cap I$ with $I \in \mathcal{I}$, we said that $C(I, e)$ was undefined (since no circuit is created in this case) and so we defined it as $C(I, e) = \{e\}$
- This explains why: for such an e, we have $\text{dep}(\mathbf{1}_I, e) = \{e\}$ since all such sets $A \ni e$ with $|I \cap A| = r(A)$ contain e, but in this case no cycle is created, i.e., $|I \cap A| \geq |I \cap \{e\}| = r(e) = 1$.

Dependence Function and Fundamental Matroid Circuit

- Therefore, when $e \in \text{sat}(\mathbf{1}_I) \setminus I$, then $\text{dep}(\mathbf{1}_I, e) = C(I, e)$ where $C(I, e)$ is the unique circuit contained in $I + e$ in a matroid (the fundamental circuit of e and I that we encountered before).
- Now, if $e \in \text{sat}(\mathbf{1}_I) \cap I$ with $I \in \mathcal{I}$, we said that $C(I, e)$ was undefined (since no circuit is created in this case) and so we defined it as $C(I, e) = \{e\}$
- This explains why: for such an e, we have $\text{dep}(\mathbf{1}_I, e) = \{e\}$ since all such sets $A \ni e$ with $|I \cap A| = r(A)$ contain e, but in this case no cycle is created, i.e., $|I \cap A| \geq |I \cap \{e\}| = r(e) = 1$. Assuming e is not a loop.
- We are thus free to take subsets of I as A, all of which must contain e, but all of which have rank equal to size, and min size is 1.

Dependence Function and Fundamental Matroid Circuit

- Therefore, when $e \in \text{sat}(\mathbf{1}_I) \setminus I$, then $\text{dep}(\mathbf{1}_I, e) = C(I, e)$ where $C(I, e)$ is the unique circuit contained in $I + e$ in a matroid (the fundamental circuit of e and I that we encountered before).
- Now, if $e \in \text{sat}(\mathbf{1}_I) \cap I$ with $I \in \mathcal{I}$, we said that $C(I, e)$ was undefined (since no circuit is created in this case) and so we defined it as $C(I, e) = \{e\}$
- This explains why: for such an e, we have $\text{dep}(\mathbf{1}_I, e) = \{e\}$ since all such sets $A \ni e$ with $|I \cap A| = r(A)$ contain e, but in this case no cycle is created, i.e., $|I \cap A| \geq |I \cap \{e\}| = r(e) = 1$.
- We are thus free to take subsets of I as A, all of which must contain e, but all of which have rank equal to size, and min size is 1.
- Also note: in general for $x \in P_f$ and $e \in \text{sat}(x)$, we have $\text{dep}(x, e)$ is tight by definition (i.e., $x(\text{dep}(x, e)) = f(\text{dep}(x, e))$), the minimum e-constaining x-tight set.

Summary of sat, and dep

- For $x \in P_f$, $\text{sat}(x)$ (span, closure) is the maximal saturated (x-tight) set w.r.t. x. I.e., $\text{sat}(x) = \{e : e \in E, \forall \alpha > 0, x + \alpha \mathbf{1}_e \notin P_f\}$. That is,

$$\text{cl}(x) \stackrel{\text{def}}{=} \text{sat}(x) \triangleq \bigcup \{A : A \in \mathcal{D}(x)\} \tag{19.9}$$

$$= \bigcup \{A : A \subseteq E, x(A) = f(A)\} \tag{19.10}$$

$$= \{e : e \in E, \forall \alpha > 0, x + \alpha \mathbf{1}_e \notin P_f\} \tag{19.11}$$

- For $e \in \text{sat}(x)$, we have $\text{dep}(x, e) \subseteq \text{sat}(x)$ (fundamental circuit) is the minimal (common) saturated (x-tight) set w.r.t. x containing e. I.e.,

$$\text{dep}(x, e) = \begin{cases} \bigcap \{A : e \in A \subseteq E, x(A) = f(A)\} & \text{if } e \in \text{sat}(x) \\ \emptyset & \text{else} \end{cases}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } x + \alpha(\mathbf{1}_e - \mathbf{1}_{e'}) \in P_f\} \tag{19.12}$$

Note, for $x \in P_f$, if $x + \alpha(\mathbf{1}_e - \mathbf{1}_{e'}) \in P_f$, then $x + \alpha'(\mathbf{1}_e - \mathbf{1}_{e'}) \in P_f$ for any $0 \leq \alpha' < \alpha$.

Dependence Function and exchange

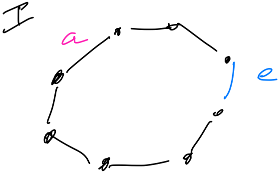
- For $e \in \text{span}(I) \setminus I$, we have that $I + e \notin \mathcal{I}$. This is a set addition restriction property.

Dependence Function and exchange

- For $e \in \operatorname{span}(I) \setminus I$, we have that $I + e \notin \mathcal{I}$. This is a set addition restriction property.
- Analogously, for $e \in \operatorname{sat}(x)$, any $x + \alpha \mathbf{1}_e \notin P_f$ for $\alpha > 0$. This is a vector increase restriction property.

Dependence Function and exchange

- For $e \in \text{span}(I) \setminus I$, we have that $I + e \notin \mathcal{I}$. This is a set addition restriction property.
- Analogously, for $e \in \text{sat}(x)$, any $x + \alpha \mathbf{1}_e \notin P_f$ for $\alpha > 0$. This is a vector increase restriction property.
- Recall, we have $C(I,e) \setminus e' \in \mathcal{I}$ for $e' \in C(I,e)$. I.e., $C(I,e)$ consists of elements that when removed recover independence.

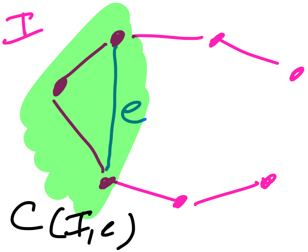


Dependence Function and exchange

- For $e \in \text{span}(I) \setminus I$, we have that $I + e \notin \mathcal{I}$. This is a set addition restriction property.
- Analogously, for $e \in \text{sat}(x)$, any $x + \alpha \mathbf{1}_e \notin P_f$ for $\alpha > 0$. This is a vector increase restriction property.
- Recall, we have $C(I,e) \setminus e' \in \mathcal{I}$ for $e' \in C(I,e)$. I.e., $C(I,e)$ consists of elements that when removed recover independence.
- In other words, given an $e \in \text{span}(I) \setminus I$, we have that

$$C(I,e) = \{a \in E : I + e - a \in \mathcal{I}\} \tag{19.13}$$

$\subseteq I + e$



Dependence Function and exchange

- For $e \in \operatorname{span}(I) \setminus I$, we have that $I + e \notin \mathcal{I}$. This is a set addition restriction property.
- Analogously, for $e \in \operatorname{sat}(x)$, any $x + \alpha \mathbf{1}_e \notin P_f$ for $\alpha > 0$. This is a vector increase restriction property.
- Recall, we have $C(I, e) \setminus e' \in \mathcal{I}$ for $e' \in C(I, e)$. I.e., $C(I, e)$ consists of elements that when removed recover independence.
- In other words, given an $e \in \operatorname{span}(I) \setminus I$, we have that

$$C(I, e) = \{a \in E : I + e - a \in \mathcal{I}\} \tag{19.13}$$

- I.e., an addition of e to I stays within $\mathcal{I}$ only if we simultaneously remove one of the elements of $C(I, e)$.

Dependence Function and exchange

- For $e \in \text{span}(I) \setminus I$, we have that $I + e \notin \mathcal{I}$. This is a set addition restriction property.
- Analogously, for $e \in \text{sat}(x)$, any $x + \alpha \mathbf{1}_e \notin P_f$ for $\alpha > 0$. This is a vector increase restriction property.
- Recall, we have $C(I, e) \setminus e' \in \mathcal{I}$ for $e' \in C(I, e)$. I.e., $C(I, e)$ consists of elements that when removed recover independence.
- In other words, given an $e \in \text{span}(I) \setminus I$, we have that

$$C(I, e) = \{a \in E : I + e - a \in \mathcal{I}\} \tag{19.13}$$

- I.e., an addition of e to I stays within $\mathcal{I}$ only if we simultaneously remove one of the elements of $C(I, e)$.
- But, analogous to the circuit case, is there an exchange property for $\text{dep}(x, e)$ in the form of vector movement restriction?

Dependence Function and exchange

- For $e \in \text{span}(I) \setminus I$, we have that $I + e \notin \mathcal{I}$. This is a set addition restriction property.
- Analogously, for $e \in \text{sat}(x)$, any $x + \alpha \mathbf{1}_e \notin P_f$ for $\alpha > 0$. This is a vector increase restriction property.
- Recall, we have $C(I, e) \setminus e' \in \mathcal{I}$ for $e' \in C(I, e)$. I.e., $C(I, e)$ consists of elements that when removed recover independence.
- In other words, given an $e \in \text{span}(I) \setminus I$, we have that

$$C(I, e) = \{a \in E : I + e - a \in \mathcal{I}\} \tag{19.13}$$

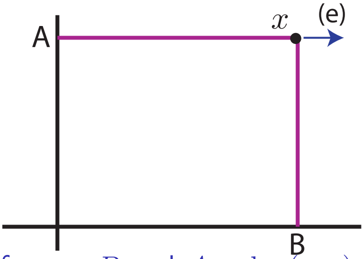
- I.e., an addition of e to I stays within $\mathcal{I}$ only if we simultaneously remove one of the elements of $C(I, e)$.
- But, analogous to the circuit case, is there an exchange property for $\text{dep}(x, e)$ in the form of vector movement restriction?
- We might expect the vector $\text{dep}(x, e)$ property to take the form: a positive move in the e-direction stays within P_f^+ only if we simultaneously take a negative move in one of the $\text{dep}(x, e)$ directions.

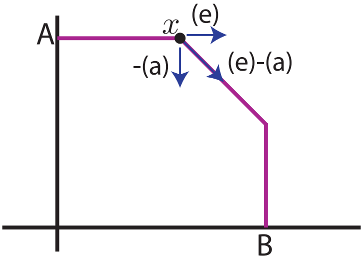
Dependence Function and exchange in 2D

- $\text{dep}(x, e)$ is set of neg. directions we must move if we want to move in pos. e direction, starting at x and staying within P_f.

Dependence Function and exchange in 2D

- $\mathrm{dep}(x, e)$ is set of neg. directions we must move if we want to move in pos. e direction, starting at x and staying within P_f.
- Viewable in 2D, we have for $A, B \subseteq E$, $A \cap B = \emptyset$:





Left: $e \in B$ and $A \cap \mathrm{dep}(x, e) = \emptyset$, and we can't move further in (e) direction, and moving in any negative $a \in A$ direction doesn't change that. **No dependence** between (e) and any element in A.

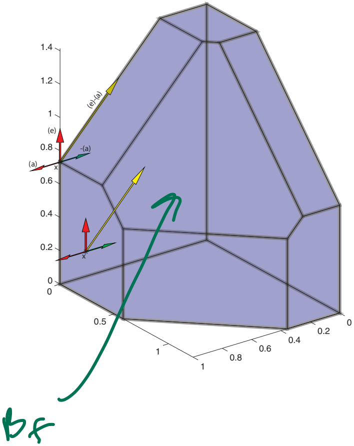
Right: $A \subseteq \mathrm{dep}(x, e)$. We can't move further in the (e) direction, but we can move further in (e) direction by moving in some negative $a \in A$ direction. **Dependence** between (e) and elements in A.

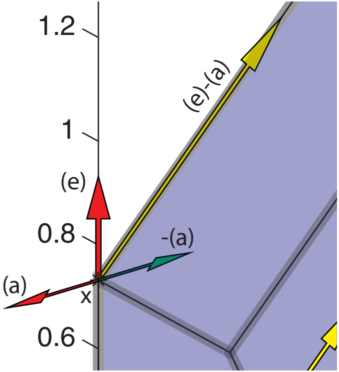
Dependence Function and exchange in 3D

- We can move neither in the (e) nor the (a) direction, but we can move in the (e) direction if we simultaneously move in the -(a) direction.

Dependence Function and exchange in 3D

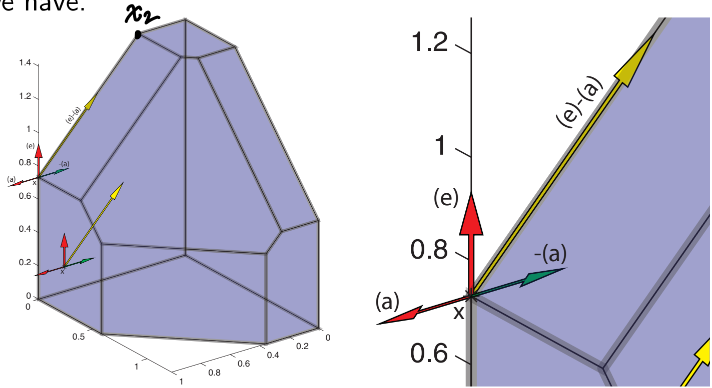
- We can move neither in the (e) nor the (a) direction, but we can move in the (e) direction if we simultaneously move in the -(a) direction.
- In 3D, we have:





Dependence Function and exchange in 3D

- We can move neither in the (e) nor the (a) direction, but we can move in the (e) direction if we simultaneously move in the -(a) direction.
- In 3D, we have:

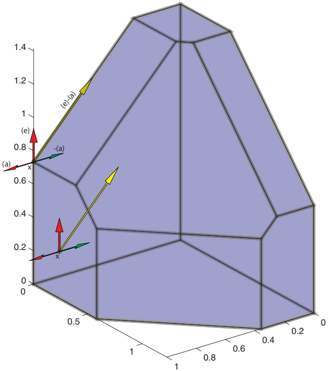


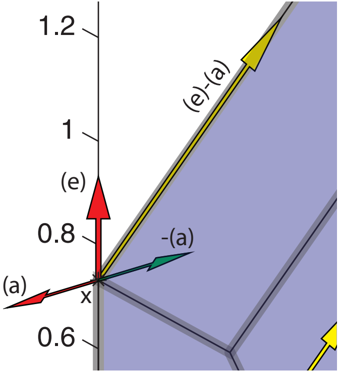
- I.e., for $e \in \text{sat}(x)$, $a \in \text{sat}(x)$, $a \in \text{dep}(x, e)$, $e \notin \text{dep}(x, a)$,

we do have $e \in \text{dep}(x_2, a)$
$a \notin \text{dep}(x_2, e)$

Dependence Function and exchange in 3D

- We can move neither in the (e) nor the (a) direction, but we can move in the (e) direction if we simultaneously move in the -(a) direction.
- In 3D, we have:



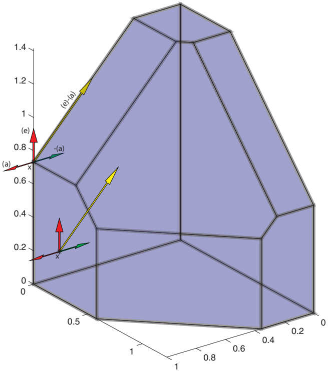


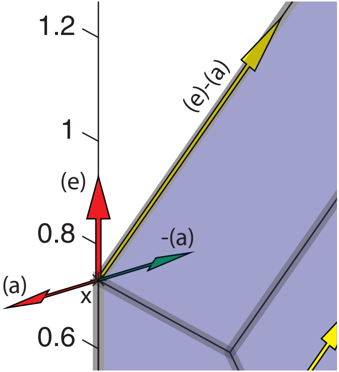
- I.e., for $e \in \text{sat}(x)$, $a \in \text{sat}(x)$, $a \in \text{dep}(x, e)$, $e \notin \text{dep}(x, a)$, and

$$\text{dep}(x, e) = \{a : a \in E, \exists \alpha > 0 : x + \alpha(\mathbf{1}_e - \mathbf{1}_a) \in P_f\} \tag{19.14}$$

Dependence Function and exchange in 3D

- We can move neither in the (e) nor the (a) direction, but we can move in the (e) direction if we simultaneously move in the -(a) direction.
- In 3D, we have:





- I.e., for $e \in \text{sat}(x)$, $a \in \text{sat}(x)$, $a \in \text{dep}(x,e)$, $e \notin \text{dep}(x,a)$, and

$$\text{dep}(x,e) = \{a : a \in E, \exists \alpha > 0 : x + \alpha(\mathbf{1}_e - \mathbf{1}_a) \in P_f\} \tag{19.14}$$

- We next show this formally . . .

dep and exchange derived

- The derivation for $\text{dep}(x, e)$, $x \in P_f$, involves turning a strict inequality into a non-strict one with a strict explicit slack variable α:

$$\text{dep}(x, e) = \text{ntight}(x, e) = \tag{19.15}$$

dep and exchange derived

- The derivation for $\text{dep}(x, e)$, $x \in P_f$, involves turning a strict inequality into a non-strict one with a strict explicit slack variable α:

$$\text{dep}(x, e) = \text{ntight}(x, e) = \tag{19.15}$$

$$= \{e' : x(A) < f(A), \forall A \not\ni e', e \in A\} \tag{19.16}$$

dep and exchange derived

- The derivation for $\text{dep}(x, e)$, $x \in P_f$, involves turning a strict inequality into a non-strict one with a strict explicit slack variable α:

$$\text{dep}(x, e) = \text{ntight}(x, e) = \tag{19.15}$$

$$= \{e' : x(A) < f(A), \forall A \not\ni e', e \in A\} \tag{19.16}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.17}$$

dep and exchange derived

- The derivation for $\text{dep}(x, e)$, $x \in P_f$, involves turning a strict inequality into a non-strict one with a strict explicit slack variable α:

$$\text{dep}(x, e) = \text{ntight}(x, e) = \tag{19.15}$$

$$= \{e' : x(A) < f(A), \forall A \not\ni e', e \in A\} \tag{19.16}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.17}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \underbrace{\alpha \mathbf{1}_e(A)}_{=\alpha} \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.18}$$

dep and exchange derived

- The derivation for $\text{dep}(x,e)$, $x \in P_f$, involves turning a strict inequality into a non-strict one with a strict explicit slack variable α:

$$\text{dep}(x,e) = \text{ntight}(x,e) = \tag{19.15}$$

$$= \{e' : x(A) < f(A), \forall A \not\ni e', e \in A\} \tag{19.16}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.17}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha \mathbf{1}_e(A) \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.18}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.19}$$

$$1_{e'}(A) = 0$$

dep and exchange derived

- The derivation for $\text{dep}(x, e)$, $x \in P_f$, involves turning a strict inequality into a non-strict one with a strict explicit slack variable α:

$$\text{dep}(x, e) = \text{ntight}(x, e) = \tag{19.15}$$

$$= \{e' : x(A) < f(A), \forall A \not\ni e', e \in A\} \tag{19.16}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.17}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha \mathbf{1}_e(A) \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.18}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.19}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } x(A) + \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) \leq f(A), \forall A \not\ni e', e \in A\} \tag{19.20}$$

dep and exchange derived

- The derivation for $\text{dep}(x, e)$, $x \in P_f$, involves turning a strict inequality into a non-strict one with a strict explicit slack variable α:

$$\text{dep}(x, e) = \text{ntight}(x, e) = \tag{19.15}$$

$$= \{e' : x(A) < f(A), \forall A \not\ni e', e \in A\} \tag{19.16}$$

$$= \{e' : \exists\alpha > 0, \text{ s.t. } \alpha \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.17}$$

$$= \{e' : \exists\alpha > 0, \text{ s.t. } \alpha \mathbf{1}_e(A) \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.18}$$

$$= \{e' : \exists\alpha > 0, \text{ s.t. } \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.19}$$

$$= \{e' : \exists\alpha > 0, \text{ s.t. } x(A) + \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) \leq f(A), \forall A \not\ni e', e \in A\} \tag{19.20}$$

- Now, $1_e(A) - \mathbf{1}_{e'}(A) = 0$ if either $\{e, e'\} \subseteq A$, or $\{e, e'\} \cap A = \emptyset$.

dep and exchange derived

- The derivation for $\text{dep}(x, e)$, $x \in P_f$, involves turning a strict inequality into a non-strict one with a strict explicit slack variable α:

$$\text{dep}(x, e) = \mathsf{ntight}(x, e) = \tag{19.15}$$

$$= \{e' : x(A) < f(A), \forall A \not\ni e', e \in A\} \tag{19.16}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.17}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha \mathbf{1}_e(A) \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.18}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) \leq f(A) - x(A), \forall A \not\ni e', e \in A\} \tag{19.19}$$

$$= \{e' : \exists \alpha > 0, \text{ s.t. } x(A) + \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) \leq f(A), \forall A \not\ni e', e \in A\} \tag{19.20}$$

- Now, $1_e(A) - \mathbf{1}_{e'}(A) = 0$ if either $\{e, e'\} \subseteq A$, or $\{e, e'\} \cap A = \emptyset$.
- Also, if $e' \in A$ but $e \notin A$, then $x(A) + \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) = x(A) - \alpha \leq f(A)$ since $x \in P_f$ and $\alpha > 0$.

dep and exchange derived

- thus, we get the same in the above if we remove the constraint $A \not\ni e', e \in A$, that is we get

$$\mathrm{dep}(x, e) = \{e' : \exists \alpha > 0, \text{ s.t. } x(A) + \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) \leq f(A), \forall A\} \tag{19.21}$$

dep and exchange derived

- thus, we get the same in the above if we remove the constraint $A \not\ni e', e \in A$, that is we get

$$\mathrm{dep}(x, e) = \{e' : \exists \alpha > 0, \text{ s.t. } x(A) + \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) \le f(A), \forall A\} \tag{19.21}$$

- This is then identical to

$$\mathrm{dep}(x, e) = \{e' : \exists \alpha > 0, \text{ s.t. } x + \alpha(\mathbf{1}_e - \mathbf{1}_{e'}) \in P_f\} \tag{19.22}$$

dep and exchange derived

- thus, we get the same in the above if we remove the constraint $A \not\ni e', e \in A$, that is we get

$$\text{dep}(x,e) = \left\{e' : \exists \alpha > 0, \text{ s.t. } x(A) + \alpha(\mathbf{1}_e(A) - \mathbf{1}_{e'}(A)) \leq f(A), \forall A\right\} \tag{19.21}$$

- This is then identical to

$$\text{dep}(x,e) = \left\{e' : \exists \alpha > 0, \text{ s.t. } x + \alpha(\mathbf{1}_e - \mathbf{1}_{e'}) \in P_f\right\} \tag{19.22}$$

- Compare with original, the minimal element of $\mathcal{D}(x,e)$, with $e \in \text{sat}(x)$:

$$\text{dep}(x,e) = \begin{cases} \bigcap \{A : e \in A \subseteq E, x(A) = f(A)\} & \text{if } e \in \text{sat}(x) \\ \emptyset & \text{else} \end{cases} \tag{19.23}$$

Submodular Function Minimization (SFM)

- We now have the tools to discuss unconstrained SFM.

Submodular Function Minimization (SFM)

- We now have the tools to discuss unconstrained SFM.
- We saw that SFM can be used to solve most violated inequality problems for a given $x \in P_f$ and, in general, SFM can solve the question "Is $x \in P_f$" by seeing if x violates any inequality (if the most violated one is negative, solution to SFM, then $x \in P_f$). That is, given $x \in \mathbb{R}^V$, compute either:

$$\min_{A \subseteq V}(f(A) - x(A)), \text{ or } \min_{A \subseteq V}(f(A) + x(V \setminus A)). \tag{19.24}$$

Submodular Function Minimization (SFM)

- We now have the tools to discuss unconstrained SFM.
- We saw that SFM can be used to solve most violated inequality problems for a given $x \in P_f$ and, in general, SFM can solve the question "Is $x \in P_f$" by seeing if x violates any inequality (if the most violated one is negative, solution to SFM, then $x \in P_f$). That is, given $x \in \mathbb{R}^V$, compute either:

$$\min_{A \subseteq V} (f(A) - x(A)), \text{ or } \min_{A \subseteq V} (f(A) + x(V \setminus A)). \tag{19.24}$$

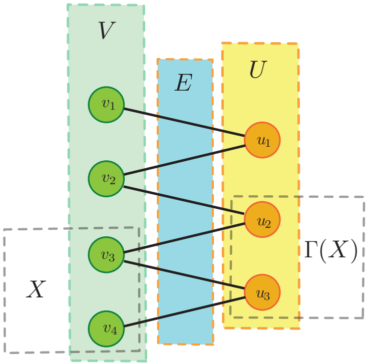
- Unconstrained SFM, $\min_{A \subseteq V} f(A)$ solves many other problems as well in combinatorial optimization, machine learning, and other fields. It generally produces sets that are homogeneous in some way as measured by f.

SFM application in ML: Low complexity data subsets.

- Find large (or preferable) and low-complexity subsets of datasets *Lin & Bilmes, "An Application of the Submodular Principal Partition to Training Data Subset Selection", NeurIPS workshops 2011*

SFM application in ML: Low complexity data subsets.

- Find large (or preferable) and low-complexity subsets of datasets *Lin & Bilmes, "An Application of the Submodular Principal Partition to Training Data Subset Selection", NeurIPS workshops 2011*
- Given bipartite graph $G = (V, U, E)$, nodes V, U and edges E, where V is a set of data objects, U is a set of possible properties of each data object (e.g., objects in images, or words in documents). Neighbor function $\Gamma(X) \subseteq U$ are the objects in $X \subseteq V$ and $f(\Gamma(X))$ is submodular for submodular $f : 2^V \to \mathbb{R}_+$.



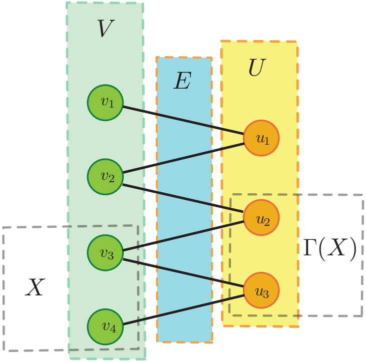
Exercise

Simple ex: $f(z) = |z|$

$f(\Gamma(x)) = |\Gamma(x)|$

SFM application in ML: Low complexity data subsets.

- Find large (or preferable) and low-complexity subsets of datasets *Lin & Bilmes, "An Application of the Submodular Principal Partition to Training Data Subset Selection", NeurIPS workshops 2011*
- Given bipartite graph $G = (V, U, E)$, nodes V, U and edges E, where V is a set of data objects, U is a set of possible properties of each data object (e.g., objects in images, or words in documents). Neighbor function $\Gamma(X) \subseteq U$ are the objects in $X \subseteq V$ and $f(\Gamma(X))$ is submodular for submodular $f : 2^V \to \mathbb{R}_+$.



- Given modular $w : 2^V \to \mathbb{R}_+$ scores for objects $v \in V$. Then $h(X) = w(V \setminus X) + f(\Gamma(X))$ is submodular, the minimization (SFM) of which produces are desiable ($w(X)$ big, large if $w(X) = |X|$) subset that is low complexity relative to $f(\Gamma(X))$. $\lambda \cdot f(\Gamma(x)) - w(x) = h'(x)$

Ellipsoid algorithm, and polynomial time SFM

- For a long time, it was not known if general purpose submodular function minimization was possible in polynomial time.

Ellipsoid algorithm, and polynomial time SFM

- For a long time, it was not known if general purpose submodular function minimization was possible in polynomial time.
- This was answered in the early 1980s via the help of Edmonds's greedy algorithm from 1970.

Ellipsoid algorithm, and polynomial time SFM

- For a long time, it was not known if general purpose submodular function minimization was possible in polynomial time.
- This was answered in the early 1980s via the help of Edmonds's greedy algorithm from 1970. Let $C \subseteq \mathbb{R}^V$ be a non-empty convex compact set.

Ellipsoid algorithm, and polynomial time SFM

- For a long time, it was not known if general purpose submodular function minimization was possible in polynomial time.
- This was answered in the early 1980s via the help of Edmonds's greedy algorithm from 1970. Let $C \subseteq \mathbb{R}^V$ be a non-empty convex compact set.

Definition 19.4.1 ((strong) optimization problem)

Given $c \in \mathbb{R}^V$, find a vector $x \in C$ that maximizes $c^\intercal x$ on C. I.e., solve

$$\max_{x \in C} c^\intercal x \tag{19.25}$$

Ellipsoid algorithm, and polynomial time SFM

- For a long time, it was not known if general purpose submodular function minimization was possible in polynomial time.
- This was answered in the early 1980s via the help of Edmonds's greedy algorithm from 1970. Let $C \subseteq \mathbb{R}^V$ be a non-empty convex compact set.

Definition 19.4.1 ((strong) optimization problem)

Given $c \in \mathbb{R}^V$, find a vector $x \in C$ that maximizes $c^\intercal x$ on C. I.e., solve

$$\max_{x \in C} c^\intercal x \tag{19.25}$$

Definition 19.4.2 ((strong) separation problem)

Given a vector $y \in \mathbb{R}^V$, decide if $y \in C$, and if not, find a hyperplane that separates y from C. I.e., find vector $c \in \mathbb{R}^V$ such that:

$$c^\intercal y > \max_{x \in C} c^\intercal x \tag{19.26}$$

Ellipsoid algorithm, and polynomial time SFM

- We have the following important theorem:

Ellipsoid algorithm, and polynomial time SFM

- We have the following important theorem:

Theorem 19.4.3 (Grötschel, Lovász, and Schrijver, 1981)

Let $\mathcal{C}$ be set of convex sets. Then there is a polynomial-time algorithm to solve the separation problem for the members of $\mathcal{C}$ iff there is a polynomial-time algorithm to solve the optimization problem for the members of $\mathcal{C}$.

Ellipsoid algorithm, and polynomial time SFM

- We have the following important theorem:

Theorem 19.4.3 (Grötschel, Lovász, and Schrijver, 1981)

Let $\mathcal{C}$ be set of convex sets. Then there is a polynomial-time algorithm to solve the separation problem for the members of $\mathcal{C}$ iff there is a polynomial-time algorithm to solve the optimization problem for the members of $\mathcal{C}$.

- We saw already that the Edmonds greedy algorithm solves the strong optimization problem for polymatroidal polytopes, e.g., $\max_{x \in B_f} c^\intercal x$.

Ellipsoid algorithm, and polynomial time SFM

- We have the following important theorem:

Theorem 19.4.3 (Grötschel, Lovász, and Schrijver, 1981)

Let $\mathcal{C}$ be set of convex sets. Then there is a polynomial-time algorithm to solve the separation problem for the members of $\mathcal{C}$ iff there is a polynomial-time algorithm to solve the optimization problem for the members of $\mathcal{C}$.

- We saw already that the Edmonds greedy algorithm solves the strong optimization problem for polymatroidal polytopes, e.g., $\max_{x \in B_f} c^\intercal x$.
- The ellipsoid algorithm first bounds a polytope P with an ellipsoid, and then creates a sequence of elipsoids of exponentially decreasing volume which are used to address a P membership problem.

Ellipsoid algorithm, and polynomial time SFM

- We have the following important theorem:

Theorem 19.4.3 (Grötschel, Lovász, and Schrijver, 1981)

Let $\mathcal{C}$ be set of convex sets. Then there is a polynomial-time algorithm to solve the separation problem for the members of $\mathcal{C}$ iff there is a polynomial-time algorithm to solve the optimization problem for the members of $\mathcal{C}$.

- We saw already that the Edmonds greedy algorithm solves the strong optimization problem for polymatroidal polytopes, e.g., $\max_{x \in B_f} c^\intercal x$.
- The ellipsoid algorithm first bounds a polytope P with an ellipsoid, and then creates a sequence of elipsoids of exponentially decreasing volume which are used to address a P membership problem.
- This is sufficient to show that we can solve SFM in polynomial time! See the book: Grötschel, Lovász, and Schrijver, "Geometric Algorithms and Combinatorial Optimization" for details.

Ellipsoid algorithm, and polynomial time SFM

- We have the following important theorem:

Theorem 19.4.3 (Grötschel, Lovász, and Schrijver, 1981)

Let $\mathcal{C}$ be set of convex sets. Then there is a polynomial-time algorithm to solve the separation problem for the members of $\mathcal{C}$ iff there is a polynomial-time algorithm to solve the optimization problem for the members of $\mathcal{C}$.

- We saw already that the Edmonds greedy algorithm solves the strong optimization problem for polymatroidal polytopes, e.g., $\max_{x \in B_f} c^\intercal x$.
- The ellipsoid algorithm first bounds a polytope P with an ellipsoid, and then creates a sequence of elipsoids of exponentially decreasing volume which are used to address a P membership problem.
- This is sufficient to show that we can solve SFM in polynomial time! See the book: Grötschel, Lovász, and Schrijver, "Geometric Algorithms and Combinatorial Optimization" for details.
- Unfortunately, it does not lead to a practical algorithm.

Lovász extension, convex minimization, and SFM

- SFM is also related to the convexity of the Lovász extension, the ease of minimizing convex functions.

Lovász extension, convex minimization, and SFM

- SFM is also related to the convexity of the Lovász extension, the ease of minimizing convex functions.
- We can recover f from $\breve{f}$ via $f(A) = \breve{f}(\mathbf{1}_A)$. We can also minimize $\breve{f}$ since it is convex.

Lovász extension, convex minimization, and SFM

- SFM is also related to the convexity of the Lovász extension, the ease of minimizing convex functions.
- We can recover f from $\breve{f}$ via $f(A) = \breve{f}(\mathbf{1}_A)$. We can also minimize $\breve{f}$ since it is convex.
- We will now show that we can get discrete solutions to the minimization of f from the continuous solution to the minimization of $\breve{f}$.

Review from lecture 17

- The next slide comes from lecture 17.

One slide review of convex closure/L.E./CI

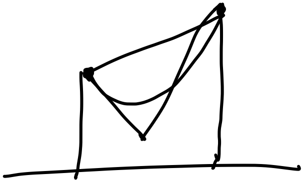
- convex closure $\breve{f}(x) = \min_{p \in \triangle^n(x)} E_{S \sim p}[f(S)]$, where where $\triangle^n(x) = \left\{ p \in \mathbb{R}^{2^n} : \sum_{S \subseteq V} p_S = 1,\ p_S \geq 0 \forall S \subseteq V,\ \&\ \sum_{S \subseteq V} p_S \mathbf{1}_S = x \right\}$
- "Edmonds" extension $\breve{f}(w) = \max(wx : x \in B_f)$, greedy.
- Lovász extension $f_{\mathsf{LE}}(w) = \sum_{i=1}^m \lambda_i f(E_i)$, with λ_i such that $w = \sum_{i=1}^m \lambda_i \mathbf{1}_{E_i}$
- $f_{\sigma^*}(w) = \max_{\sigma \in \Pi_{[m]}} w^\intercal c^\sigma$, $\Pi_{[m]}$ set of $m!$ permutations of $[m]$, $\sigma \in \Pi_{[m]}$ a permutation, c^σ vector with $c_i^\sigma = f(E_{\sigma_i}) - f(E_{\sigma_{i-1}})$, $E_{\sigma_i} = \{e_{\sigma_1}, e_{\sigma_2}, \ldots, e_{\sigma_i}\}$.
- Choquet integral $C_f(w) = \sum_{i=1}^m (w_{e_i} - w_{e_{i+1}}) f(E_i)$
- $\mathfrak{f}(w) = \int_{-\infty}^{+\infty} \hat{f}(\alpha) d\alpha$, $\hat{f}(\alpha) = \begin{cases} f(\{w \geq \alpha\}) & \text{if } \alpha \geq 0 \\ f(\{w \geq \alpha\}) - f(E) & \text{if } \alpha < 0 \end{cases}$
- All the same when f is submodular. We'll use $\breve{f}(w)$ for the Lovász extension.

Minimizing $\breve{f}$ vs. minimizing f

In fact, we have:

Theorem 19.5.1

Let f be submodular and $\breve{f}$ be its Lovász extension. Then $\min\{f(A)|A \subseteq E\} = \min_{w\in\{0,1\}^E} \breve{f}(w) = \min_{w\in[0,1]^E} \breve{f}(w)$.



Minimizing $\breve{f}$ vs. minimizing f

In fact, we have:

Theorem 19.5.1

Let f be submodular and $\breve{f}$ be its Lovász extension. Then $\min \{f(A) | A \subseteq E\} = \min_{w \in \{0,1\}^E} \breve{f}(w) = \min_{w \in [0,1]^E} \breve{f}(w)$.

Proof.

- First, since $\breve{f}(\mathbf{1}_A) = f(A), \forall A \subseteq V$, we clearly have $\min \{f(A) | A \subseteq V\} = \min_{w \in \{0,1\}^E} \breve{f}(w) \geq \min_{w \in [0,1]^E} \breve{f}(w)$.

...

Minimizing $\breve{f}$ vs. minimizing f

In fact, we have:

Theorem 19.5.1

Let f be submodular and $\breve{f}$ be its Lovász extension. Then $\min\{f(A)|A \subseteq E\} = \min_{w\in\{0,1\}^E} \breve{f}(w) = \min_{w\in[0,1]^E} \breve{f}(w)$.

Proof.

- First, since $\breve{f}(\mathbf{1}_A) = f(A), \forall A \subseteq V$, we clearly have $\min\{f(A)|A \subseteq V\} = \min_{w\in\{0,1\}^E} \breve{f}(w) \geq \min_{w\in[0,1]^E} \breve{f}(w)$.
- Next, consider any $w \in [0,1]^E$, sort elements $E = \{e_1, \ldots, e_m\}$ as $w(e_1) \geq w(e_2) \geq \cdots \geq w(e_m)$, define $E_i = \{e_1, \ldots, e_i\}$, and define $\lambda_m = w(e_m)$ and $\lambda_i = w(e_i) - w(e_{i+1})$ for $i \in \{1, \ldots, m-1\}$.

...

Minimizing $\breve{f}$ vs. minimizing f

In fact, we have:

Theorem 19.5.1

Let f be submodular and $\breve{f}$ be its Lovász extension. Then $\min\{f(A)|A \subseteq E\} = \min_{w \in \{0,1\}^E} \breve{f}(w) = \min_{w \in [0,1]^E} \breve{f}(w)$.

Proof.

- First, since $\breve{f}(\mathbf{1}_A) = f(A), \forall A \subseteq V$, we clearly have $\min\{f(A)|A \subseteq V\} = \min_{w \in \{0,1\}^E} \breve{f}(w) \geq \min_{w \in [0,1]^E} \breve{f}(w)$.
- Next, consider any $w \in [0,1]^E$, sort elements $E = \{e_1, \dots, e_m\}$ as $w(e_1) \geq w(e_2) \geq \dots \geq w(e_m)$, define $E_i = \{e_1, \dots, e_i\}$, and define $\lambda_m = w(e_m)$ and $\lambda_i = w(e_i) - w(e_{i+1})$ for $i \in \{1, \dots, m-1\}$.
- Then, as we have seen, $w = \sum_i \lambda_i \mathbf{1}_{E_i}$ and $\lambda_i \geq 0$.

...

Minimizing $\breve{f}$ vs. minimizing f

In fact, we have:

Theorem 19.5.1

Let f be submodular and $\breve{f}$ be its Lovász extension. Then $\min\{f(A)|A \subseteq E\} = \min_{w \in \{0,1\}^E} \breve{f}(w) = \min_{w \in [0,1]^E} \breve{f}(w)$.

Proof.

- First, since $\breve{f}(\mathbf{1}_A) = f(A), \forall A \subseteq V$, we clearly have $\min\{f(A)|A \subseteq V\} = \min_{w \in \{0,1\}^E} \breve{f}(w) \geq \min_{w \in [0,1]^E} \breve{f}(w)$.
- Next, consider any $w \in [0,1]^E$, sort elements $E = \{e_1, \dots, e_m\}$ as $w(e_1) \geq w(e_2) \geq \dots \geq w(e_m)$, define $E_i = \{e_1, \dots, e_i\}$, and define $\lambda_m = w(e_m)$ and $\lambda_i = w(e_i) - w(e_{i+1})$ for $i \in \{1, \dots, m-1\}$.
- Then, as we have seen, $w = \sum_i \lambda_i \mathbf{1}_{E_i}$ and $\lambda_i \geq 0$.
- Also, $\sum_i \lambda_i = w(e_1) \leq 1$.

...

Minimizing $\breve{f}$ vs. minimizing f

. . . cont. proof of Thm. 19.5.1.

- Note that since $f(\emptyset) = 0$, $\min\{f(A) | A \subseteq E\} \leq 0$.

$\square$

Minimizing $\breve{f}$ vs. minimizing f

. . . cont. proof of Thm. 19.5.1.

- Note that since $f(\emptyset) = 0$, $\min\{f(A) | A \subseteq E\} \leq 0$.
- Then we have for all $w \in [0,1]^E$,

$$\breve{f}(w) = \int_0^1 f(\{w \geq \alpha\}) d\alpha = \sum_{i=1}^m \lambda_i f(E_i) \tag{19.27}$$

$$\geq \sum_{i=1}^m \lambda_i \min_{A \subseteq E} f(A) \tag{19.28}$$

$$\geq \min_{A \subseteq E} f(A) \tag{19.29}$$

$\sum_{i=1}^{m} \lambda_i \leq 1$

□

Minimizing $\breve{f}$ vs. minimizing f

. . . cont. proof of Thm. 19.5.1.

- Note that since $f(\emptyset) = 0$, $\min \{f(A) | A \subseteq E\} \leq 0$.
- Then we have for all $w \in [0,1]^E$,

$$\breve{f}(w) = \int_0^1 f(\{w \geq \alpha\}) d\alpha = \sum_{i=1}^m \lambda_i f(E_i) \tag{19.27}$$

$$\geq \sum_{i=1}^m \lambda_i \min_{A \subseteq E} f(A) \tag{19.28}$$

$$\geq \min_{A \subseteq E} f(A) \tag{19.29}$$

- Thus, $\min \{f(A) | A \subseteq E\} = \min_{w \in [0,1]^E} \breve{f}(w)$.

□

Other minimizers based on min of $\breve{f}$

- Let $w^* \in \text{argmin}\left\{\breve{f}(w) | w \in [0,1]^E\right\}$ and let $A^* \in \underbrace{\text{argmin}\left\{f(A) | A \subseteq V\right\}}_{\text{a lattice.}}$.

Other minimizers based on min of $\breve{f}$

- Let $w^* \in \operatorname{argmin}\left\{\breve{f}(w) | w \in [0,1]^E\right\}$ and let $A^* \in \operatorname{argmin}\{f(A) | A \subseteq V\}$.
- Previous theorem states that $\breve{f}(w^*) = f(A^*)$.

Other minimizers based on min of $\breve{f}$

- Let $w^* \in \operatorname{argmin}\left\{\breve{f}(w) | w \in [0,1]^E\right\}$ and let $A^* \in \operatorname{argmin}\{f(A) | A \subseteq V\}$.
- Previous theorem states that $\breve{f}(w^*) = f(A^*)$.
- Let λ_i^* be the Lovász extension weights and E_i^* be the chain of sets associated with optimal w^*. From previous theorem, we have

$$\breve{f}(w^*) = \sum_i \lambda_i^* f(E_i^*) = f(A^*) = \min\{f(A) | A \subseteq E\} \tag{19.30}$$

Other minimizers based on min of $\breve{f}$

- Let $w^* \in \operatorname{argmin}\left\{\breve{f}(w) | w \in [0,1]^E\right\}$ and let $A^* \in \operatorname{argmin}\{f(A) | A \subseteq V\}$.
- Previous theorem states that $\breve{f}(w^*) = f(A^*)$.
- Let λ_i^* be the Lovász extension weights and E_i^* be the chain of sets associated with optimal w^*. From previous theorem, we have

$$\breve{f}(w^*) = \sum_i \lambda_i^* f(E_i^*) = f(A^*) = \min\{f(A) | A \subseteq E\} \tag{19.30}$$

and that $f(A^*) \leq f(E_i^*), \forall i,$

Other minimizers based on min of $\breve{f}$

- Let $w^* \in \operatorname{argmin}\left\{\breve{f}(w) | w \in [0,1]^E\right\}$ and let $A^* \in \operatorname{argmin}\{f(A) | A \subseteq V\}$.
- Previous theorem states that $\breve{f}(w^*) = f(A^*)$.
- Let λ_i^* be the Lovász extension weights and E_i^* be the chain of sets associated with optimal w^*. From previous theorem, we have

$$\breve{f}(w^*) = \sum_i \lambda_i^* f(E_i^*) = f(A^*) = \min\{f(A) | A \subseteq E\} \tag{19.30}$$

and that $f(A^*) \leq f(E_i^*), \forall i$, and that $f(A^*) \leq 0$,

Other minimizers based on min of $\breve{f}$

- Let $w^* \in \operatorname{argmin}\left\{\breve{f}(w) | w \in [0,1]^E\right\}$ and let $A^* \in \operatorname{argmin}\{f(A) | A \subseteq V\}$.
- Previous theorem states that $\breve{f}(w^*) = f(A^*)$.
- Let λ_i^* be the Lovász extension weights and E_i^* be the chain of sets associated with optimal w^*. From previous theorem, we have

$$\breve{f}(w^*) = \sum_i \lambda_i^* f(E_i^*) = f(A^*) = \min\{f(A) | A \subseteq E\} \qquad (19.30)$$

and that $f(A^*) \leq f(E_i^*), \forall i$, and that $f(A^*) \leq 0$, and $\sum_i \lambda_i \leq 1$.

Other minimizers based on min of $\breve{f}$

- Let $w^* \in \operatorname{argmin}\left\{\breve{f}(w) | w \in [0,1]^E\right\}$ and let $A^* \in \operatorname{argmin}\{f(A) | A \subseteq V\}$.
- Previous theorem states that $\breve{f}(w^*) = f(A^*)$.
- Let λ_i^* be the Lovász extension weights and E_i^* be the chain of sets associated with optimal w^*. From previous theorem, we have

$$\breve{f}(w^*) = \sum_i \lambda_i^* f(E_i^*) = f(A^*) = \min\{f(A) | A \subseteq E\} \tag{19.30}$$

and that $f(A^*) \leq f(E_i^*), \forall i$, and that $f(A^*) \leq 0$, and $\sum_i \lambda_i \leq 1$.
- Thus, since $w^* \in [0,1]^E$, each $0 \leq \lambda_i^* \leq 1$, we have for all i such that $\lambda_i^* > 0$,

$$f(E_i^*) = f(A^*) \tag{19.31}$$

meaning such E_i^* are also minimizers of f, and $\sum_i \lambda_i = 1$.

Other minimizers based on min of $\breve{f}$

- Let $w^* \in \operatorname{argmin}\left\{\breve{f}(w) | w \in [0,1]^E\right\}$ and let $A^* \in \operatorname{argmin}\{f(A) | A \subseteq V\}$.
- Previous theorem states that $\breve{f}(w^*) = f(A^*)$.
- Let λ_i^* be the Lovász extension weights and E_i^* be the chain of sets associated with optimal w^*. From previous theorem, we have

$$\breve{f}(w^*) = \sum_i \lambda_i^* f(E_i^*) = f(A^*) = \min\{f(A) | A \subseteq E\} \tag{19.30}$$

  and that $f(A^*) \leq f(E_i^*), \forall i$, and that $f(A^*) \leq 0$, and $\sum_i \lambda_i \leq 1$.
- Thus, since $w^* \in [0,1]^E$, each $0 \leq \lambda_i^* \leq 1$, we have for all i such that $\lambda_i^* > 0$,

$$f(E_i^*) = f(A^*) \tag{19.31}$$

  meaning such E_i^* are also minimizers of f, and $\sum_i \lambda_i = 1$.
- Note that the negative of $f(A^*)$ is crucial here (see next slides).

Other minimizers based on min of $\breve{f}$

- Let $w^* \in \operatorname{argmin}\left\{\breve{f}(w) | w \in [0,1]^E\right\}$ and let $A^* \in \operatorname{argmin}\{f(A) | A \subseteq V\}$.
- Previous theorem states that $\breve{f}(w^*) = f(A^*)$.
- Let λ_i^* be the Lovász extension weights and E_i^* be the chain of sets associated with optimal w^*. From previous theorem, we have

$$\breve{f}(w^*) = \sum_i \lambda_i^* f(E_i^*) = f(A^*) = \min\{f(A) | A \subseteq E\} \tag{19.30}$$

  and that $f(A^*) \leq f(E_i^*), \forall i$, and that $f(A^*) \leq 0$, and $\sum_i \lambda_i \leq 1$.

show next slide

- Thus, since $w^* \in [0,1]^E$, each $0 \leq \lambda_i^* \leq 1$, we have for all i such that $\lambda_i^* > 0$,

$$f(E_i^*) = f(A^*) \tag{19.31}$$

  meaning such E_i^* are also minimizers of f, and $\sum_i \lambda_i = 1$.
- Note that the negative of $f(A^*)$ is crucial here (see next slides).
- By the L.E. properties, $w^* = \sum_i \lambda_i^* \mathbf{1}_{E_i}$, we have that w^* is in the convex hull of incidence vectors of minimizers of f.

A bit more on level sets being minimizers

- f is normalized $f(\emptyset) = 0$, so minimizer is ≤ 0.
- We know that $f(E_i^*) \geq f(A^*)$ for all i, and $f(A^*) = \sum_i \lambda_i f(E_i^*)$.
- If $f(A^*) = 0$, then we must have $f(E_i^*) = 0$ for any i such that $\lambda_i > 0$. Otherwise, assume $f(A^*) < 0$.
- Suppose there exists an i such that $f(E_i^*) > f(A^*)$.
- Then we have

$$f(A^*) = \sum_i \lambda_i f(E_i^*) > \sum_i \lambda_i f(A^*) = f(A^*) \sum_i \lambda_i \tag{19.32}$$

  and since $f(A^*) < 0$, this means that $\sum_i \lambda_i > 1$ which is a contradiction.
- Hence, must have $f(E_i^*) = f(A^*)$ for all i. with $\lambda_i > 0$
- Hence, $\sum_i \lambda_i = 1$ since $f(A^*) = \sum_i \lambda_i f(A^*)$.

Yet another way to see Equation 19.31

- We know $f(A^*) \leq 0$. Consider two cases in Equation 19.31.
- Case 1: $f(A^*) = 0$. Then for any i with $\lambda_i > 0$ we must have $f(E_i) = 0$ as well for all i since $f(A^*) \leq f(E_i)$.
- Case 2 is where $f(A^*) < 0$. In this second case, we have

$$0 > f(A^*) = \sum_i \lambda_i f(E_i) \geq \sum_i \lambda_i f(A^*) \tag{19.33}$$

$$\overset{(a)}{\geq} \sum_i \lambda_i f(A^*) + (1 - \bar{\lambda}) f(A^*) = f(A^*) \tag{19.34}$$

where $\bar{\lambda} = \sum_i \lambda_i$ and $(1 - \bar{\lambda}) \geq 0$ and where (a) follows since $f(A^*) < 0$.

- Hence, all inequalities must be equalities, which means that we must have that $\bar{\lambda} = 1$.

θ-rounding the L.E. minimum

We can also view the above as a form of rounding a continuous convex relaxation to the problem.

Definition 19.5.2 (θ-rounding)

Given vector $\hat{x} \in [0,1]^E$, choose $\theta \in (0,1)$ and define a set corresponding to elements above θ, i.e.,

$$\hat{X}_\theta = \{i : \hat{x}(i) \geq \theta\} \triangleq \{\hat{x} \geq \theta\} \tag{19.35}$$

Lemma 19.5.3 (Fujishige-2005)

Given a continuous minimizer $x^* \in \operatorname{argmin}_{x \in [0,1]^n} \breve{f}(x)$, the discrete minimizers are exactly the maximal chain of sets $\emptyset \subseteq X_{\theta_1} \subset \ldots X_{\theta_k}$ obtained by θ-rounding x^*, for $\theta_j \in (0,1)$.

- gradient of L.E. is easy to get. (wednesday)

Min-Norm Point: Definition

- Consider the optimization:

$$\text{minimize} \quad \|x\|_2^2 \tag{19.36a}$$

$$\text{subject to} \quad x \in B_f \tag{19.36b}$$

where B_f is the base polytope of submodular f, and $\|x\|_2^2 = \sum_{e \in E} x(e)^2$ is the squared 2-norm. Let x^* be the optimal solution.

Min-Norm Point: Definition

- Consider the optimization:

$$\text{minimize} \quad \|x\|_2^2 \tag{19.36a}$$

$$\text{subject to} \quad x \in B_f \tag{19.36b}$$

where B_f is the base polytope of submodular f, and $\|x\|_2^2 = \sum_{e \in E} x(e)^2$ is the squared 2-norm. Let x^* be the optimal solution.

- Note, x^* is the unique optimal solution since we have a strictly convex objective over a set of convex constraints.

Min-Norm Point: Definition

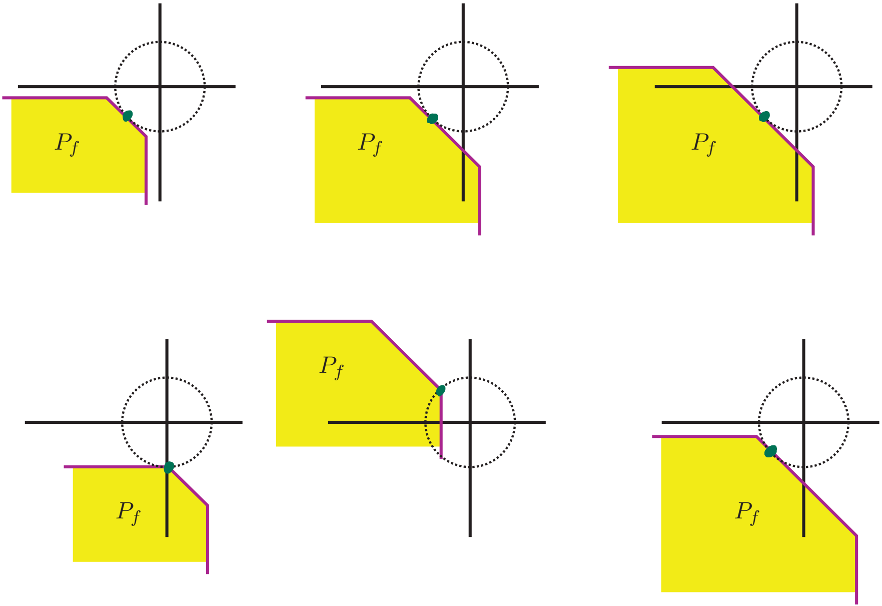
- Consider the optimization:

$$\text{minimize} \quad \|x\|_2^2 \tag{19.36a}$$

$$\text{subject to} \quad x \in B_f \tag{19.36b}$$

  where B_f is the base polytope of submodular f, and $\|x\|_2^2 = \sum_{e \in E} x(e)^2$ is the squared 2-norm. Let x^* be the optimal solution.
- Note, x^* is the unique optimal solution since we have a strictly convex objective over a set of convex constraints.
- x^* is called the minimum norm point of the base polytope.

Min-Norm Point: Examples



Ex: 3D base B_f: permutahedron

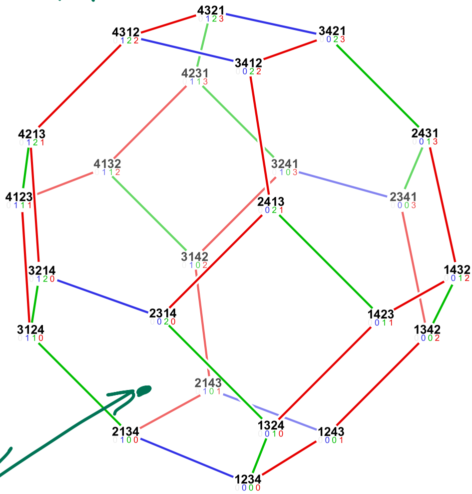
- Consider submodular function $f : 2^V \to \mathbb{R}$ with $n = |V| = 4$, and for $X \subseteq V$, concave g,

$$f(X) = g(|X|) = \sum_{i=1}^{|X|}(n - i + 1)$$

$$= |X|\left(n - \frac{|X|-1}{2}\right)$$

- Then B_f is a 3D polytope, and in this particular case gives us a permutahedron with 24 distinct extreme points, on the right (from wikipedia).

B_f



Min-Norm Point and Submodular Function Minimization

- Given optimal solution x^* to $[\min \|x\|_2^2$ s.t. $x \in B_f]$, and consider:

$$y^* = x^* \wedge 0 = (\min(x^*(e), 0) | e \in E), \tag{19.37}$$

$$\underbrace{A_- = \{e : x^*(e) < 0\}}_{\text{negative elements of min-norm point}}, \qquad \underbrace{A_0 = \{e : x^*(e) \leq 0\}}_{\text{non-positive elements of min-norm point}}. \tag{19.38}$$

Min-Norm Point and Submodular Function Minimization

- Given optimal solution x^* to $[\min \|x\|_2^2$ s.t. $x \in B_f]$, and consider:

$$y^* = x^* \wedge 0 = (\min(x^*(e), 0) | e \in E), \tag{19.37}$$

$$A_- = \{e : x^*(e) < 0\}, \qquad A_0 = \{e : x^*(e) \leq 0\}. \tag{19.38}$$

- Thus, we immediately have that:

$$A_- \subseteq A_0 \tag{19.39}$$

and that

$$x^*(A_-) = x^*(A_0) = y^*(A_-) = y^*(A_0). \tag{19.40}$$

Min-Norm Point and Submodular Function Minimization

- Given optimal solution x^* to $[\min \|x\|_2^2$ s.t. $x \in B_f]$, and consider:

$$y^* = x^* \wedge 0 = (\min(x^*(e), 0) | e \in E), \tag{19.37}$$

$$A_- = \{e : x^*(e) < 0\}, \qquad A_0 = \{e : x^*(e) \leq 0\}. \tag{19.38}$$

- Thus, we immediately have that:

$$A_- \subseteq A_0 \tag{19.39}$$

and that

$$x^*(A_-) = x^*(A_0) = y^*(A_-) = y^*(A_0). \tag{19.40}$$

- These quantities will solve the SFM problem: we will see that $f(A_-) = f(A_0) = \min_{A \subseteq V} f(A)$ and that A_- is the unique minimal minimizer and A_0 is the unique maximal minimizer.

Min-Norm Point and Submodular Function Minimization

- Given optimal solution x^* to $[\min \|x\|_2^2$ s.t. $x \in B_f]$, and consider:

$$y^* = x^* \wedge 0 = (\min(x^*(e), 0) | e \in E), \tag{19.37}$$

$$A_- = \{e : x^*(e) < 0\}, \qquad A_0 = \{e : x^*(e) \leq 0\}. \tag{19.38}$$

- Thus, we immediately have that:

$$A_- \subseteq A_0 \tag{19.39}$$

and that

$$x^*(A_-) = x^*(A_0) = y^*(A_-) = y^*(A_0). \tag{19.40}$$

- These quantities will solve the SFM problem: we will see that $f(A_-) = f(A_0) = \min_{A \subseteq V} f(A)$ and that A_- is the unique minimal minimizer and A_0 is the unique maximal minimizer.
- The proof is nice since it uses recently developed tools (e.g., dep, sat).

Min-Norm Point and Submodular Function Minimization

- Given optimal solution x^* to $[\min \|x\|_2^2 \text{ s.t. } x \in B_f]$, and consider:

$$y^* = x^* \wedge 0 = (\min(x^*(e), 0) | e \in E), \tag{19.37}$$

$$A_- = \{e : x^*(e) < 0\}, \qquad A_0 = \{e : x^*(e) \leq 0\}. \tag{19.38}$$

- Thus, we immediately have that:

$$A_- \subseteq A_0 \tag{19.39}$$

and that

$$x^*(A_-) = x^*(A_0) = y^*(A_-) = y^*(A_0). \tag{19.40}$$

- These quantities will solve the SFM problem: we will see that $f(A_-) = f(A_0) = \min_{A \subseteq V} f(A)$ and that A_- is the unique minimal minimizer and A_0 is the unique maximal minimizer.
- The proof is nice since it uses recently developed tools (e.g., dep, sat).
- We'll also show both the Fujishige-Wolfe algorithm and the Frank-Wolfe algorithm (which are quite different from each other) can find the min-norm point relatively efficiently.

Base Polytope B_f Existence

- Given polymatroid function f, the base polytope $B_f = \left\{x \in \mathbb{R}_+^E : x(A) \leq f(A)\ \forall A \subseteq E,\right.$ and $\left.x(E) = f(E)\right\}$ always exists.

Base Polytope B_f Existence

- Given polymatroid function f, the base polytope $B_f = \left\{x \in \mathbb{R}_+^E : x(A) \leq f(A)\ \forall A \subseteq E, \text{ and } x(E) = f(E)\right\}$ always exists.
- Consider any order of E and generate a vector x by this order (i.e., $x(e_1) = f(\{e_1\})$, $x(e_2) = f(\{e_1, e_2\}) - f(\{e_1\})$, and so on).

Base Polytope B_f Existence

- Given polymatroid function f, the base polytope $B_f = \left\{x \in \mathbb{R}_+^E : x(A) \leq f(A) \; \forall A \subseteq E, \text{ and } x(E) = f(E)\right\}$ always exists.
- Consider any order of E and generate a vector x by this order (i.e., $x(e_1) = f(\{e_1\})$, $x(e_2) = f(\{e_1, e_2\}) - f(\{e_1\})$, and so on).
- From past lectures, we now know that:

Base Polytope B_f Existence

- Given polymatroid function f, the base polytope $B_f = \left\{x \in \mathbb{R}_+^E : x(A) \leq f(A)\ \forall A \subseteq E, \text{ and } x(E) = f(E)\right\}$ always exists.
- Consider any order of E and generate a vector x by this order (i.e., $x(e_1) = f(\{e_1\})$, $x(e_2) = f(\{e_1, e_2\}) - f(\{e_1\})$, and so on).
- From past lectures, we now know that:

  (1) $x \in P_f$

Base Polytope B_f Existence

- Given polymatroid function f, the base polytope $B_f = \left\{x \in \mathbb{R}_+^E : x(A) \leq f(A)\ \forall A \subseteq E, \text{ and } x(E) = f(E)\right\}$ always exists.
- Consider any order of E and generate a vector x by this order (i.e., $x(e_1) = f(\{e_1\})$, $x(e_2) = f(\{e_1, e_2\}) - f(\{e_1\})$, and so on).
- From past lectures, we now know that:
  (1) $x \in P_f$
  (2) x is an extreme point in P_f

Base Polytope B_f Existence

- Given polymatroid function f, the base polytope $B_f = \{x \in \mathbb{R}_+^E : x(A) \leq f(A) \; \forall A \subseteq E, \text{ and } x(E) = f(E)\}$ always exists.
- Consider any order of E and generate a vector x by this order (i.e., $x(e_1) = f(\{e_1\})$, $x(e_2) = f(\{e_1, e_2\}) - f(\{e_1\})$, and so on).
- From past lectures, we now know that:
  1. $x \in P_f$
  2. x is an extreme point in P_f
  3. Since x is generated using an ordering of all of E, we have that $x(E) = f(E)$.

$x(E_i) = f(E_i) \quad \forall i$

$E_m = E$

Base Polytope B_f Existence

- Given polymatroid function f, the base polytope $B_f = \big\{x \in \mathbb{R}_+^E : x(A) \leq f(A)\ \forall A \subseteq E, \text{ and } x(E) = f(E)\big\}$ always exists.
- Consider any order of E and generate a vector x by this order (i.e., $x(e_1) = f(\{e_1\})$, $x(e_2) = f(\{e_1, e_2\}) - f(\{e_1\})$, and so on).
- From past lectures, we now know that:
  1. $x \in P_f$
  2. x is an extreme point in P_f
  3. Since x is generated using an ordering of all of E, we have that $x(E) = f(E)$.
- Thus $x \in B_f$, and B_f is never empty.

Base Polytope B_f Existence

- Given polymatroid function f, the base polytope $B_f = \left\{x \in \mathbb{R}_+^E : x(A) \leq f(A)\ \forall A \subseteq E, \text{ and } x(E) = f(E)\right\}$ always exists.
- Consider any order of E and generate a vector x by this order (i.e., $x(e_1) = f(\{e_1\})$, $x(e_2) = f(\{e_1, e_2\}) - f(\{e_1\})$, and so on).
- From past lectures, we now know that:
  (1) $x \in P_f$
  (2) x is an extreme point in P_f
  (3) Since x is generated using an ordering of all of E, we have that $x(E) = f(E)$.
- Thus $x \in B_f$, and B_f is never empty.
- Moreover, in this case, x is a vertex of B_f since it is extremal.

Base Polytope B_f Dominance

- Now, for any $A \subseteq E$, we can generate a particular point x in B_f

s.t. $x(A) = f(A)$.

Base Polytope B_f Dominance

- Now, for any $A \subseteq E$, we can generate a particular point in B_f
- That is, choose the ordering of $E = (e_1, e_2, \ldots, e_n)$ where $n = |E|$, and where $E_i = (e_1, e_2, \ldots, e_i)$, so that we have $E_k = A$ with $k = |A|$.

Base Polytope B_f Dominance

- Now, for any $A \subseteq E$, we can generate a particular point in B_f
- That is, choose the ordering of $E = (e_1, e_2, \dots, e_n)$ where $n = |E|$, and where $E_i = (e_1, e_2, \dots, e_i)$, so that we have $E_k = A$ with $k = |A|$.
- Note there are $k!(n-k)! < n!$ such orderings.

Base Polytope B_f Dominance

- Now, for any $A \subseteq E$, we can generate a particular point in B_f
- That is, choose the ordering of $E = (e_1, e_2, \ldots, e_n)$ where $n = |E|$, and where $E_i = (e_1, e_2, \ldots, e_i)$, so that we have $E_k = A$ with $k = |A|$.
- Note there are $k!(n-k)! < n!$ such orderings.
- Generate x via greedy using this order, $\forall i, x(e_i) = f(e_i|E_{i-1})$.

Base Polytope B_f Dominance

- Now, for any $A \subseteq E$, we can generate a particular point in B_f
- That is, choose the ordering of $E = (e_1, e_2, \dots, e_n)$ where $n = |E|$, and where $E_i = (e_1, e_2, \dots, e_i)$, so that we have $E_k = A$ with $k = |A|$.
- Note there are $k!(n-k)! < n!$ such orderings.
- Generate x via greedy using this order, $\forall i, x(e_i) = f(e_i | E_{i-1})$.
- We have generated a point (a vertex) x in B_f such that $x(A) = f(A)$.

Since $x(E_i) = f(E_i) \quad \forall i$

and $E_k = A$.

Base Polytope B_f Dominance

- Now, for any $A \subseteq E$, we can generate a particular point in B_f
- That is, choose the ordering of $E = (e_1, e_2, \ldots, e_n)$ where $n = |E|$, and where $E_i = (e_1, e_2, \ldots, e_i)$, so that we have $E_k = A$ with $k = |A|$.
- Note there are $k!(n-k)! < n!$ such orderings.
- Generate x via greedy using this order, $\forall i, x(e_i) = f(e_i | E_{i-1})$.
- We have generated a point (a vertex) x in B_f such that $x(A) = f(A)$.
- Thus, for any A, we have

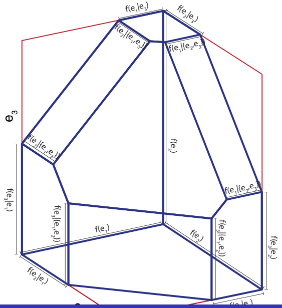
$$B_f \cap \left\{ x \in \mathbb{R}^E : x(A) = f(A) \right\} \neq \emptyset \tag{19.41}$$

Base Polytope B_f Dominance

- Now, for any $A \subseteq E$, we can generate a particular point in B_f
- That is, choose the ordering of $E = (e_1, e_2, \ldots, e_n)$ where $n = |E|$, and where $E_i = (e_1, e_2, \ldots, e_i)$, so that we have $E_k = A$ with $k = |A|$.
- Note there are $k!(n-k)! < n!$ such orderings.
- Generate x via greedy using this order, $\forall i, x(e_i) = f(e_i | E_{i-1})$.
- We have generated a point (a vertex) x in B_f such that $x(A) = f(A)$.
- Thus, for any A, we have

$$B_f \cap \{x \in \mathbb{R}^E : x(A) = f(A)\} \neq \emptyset \tag{19.41}$$

- In words, B_f intersects all "multi-axis congruent" hyperplanes within R^E of the form $\{x \in \mathbb{R}^E : x(A) = f(A)\}$ for all $A \subseteq E$.



B_f dominates P_f

- In fact, every $x \in P_f$ is dominated by $x \leq y \in B_f$.

Theorem 19.7.1

If $x \in P_f$ and T is tight for x (meaning $x(T) = f(T)$), then there exists $y \in B_f$ with $x \leq y$ and $y(e) = x(e)$ for $e \in T$.

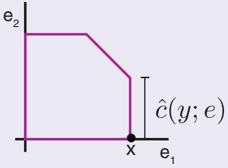
Proof.

- We construct the y algorithmically: initially set $y \leftarrow x$.
- $y \in P_f$, T is tight for y so $y(T) = f(T)$.
- Recall saturation capacity: for $y \in P_f$, $\hat{c}(y;e) = \min\{f(A) - y(A) | \forall A \ni e\} = \max\{\alpha : \alpha \in \mathbb{R}, y + \alpha \mathbf{1}_e \in P_f\}$
- Consider following algorithm:

$T' \leftarrow T$;

for $e \in E \setminus T$ **do**

$\quad y \leftarrow y + c(y;e)\mathbf{1}_e$; $T' \leftarrow T' \cup \{e\}$;



...

B_f dominates P_f

. . . proof of Thm. 19.7.1 cont.

- Each step maintains feasibility: consider one step adding e to T' — for $e \notin T'$, feasibility requires $y(T'+e) = y(T') + y(e) \leq f(T'+e)$, or $y(e) \leq f(T'+e) - y(T') = y(e) + f(T'+e) - y(T'+e)$.
- We set $y(e) \leftarrow y(e) + \hat{c}(y;e) \leq y(e) + f(T'+e) - y(T'+e)$. Hence, after each step, $y \in P_f$ and $\hat{c}(y;e) \geq 0$. (also, consider r.h. version of $\hat{c}(y;e)$).
- Also, only $y(e)$ for $e \notin T$ changed, final y has $y(e) = x(e)$ for $e \in T$.
- Let $S_e \ni e$ be a set that achieves $c(y;e) = f(S_e) - y(S_e)$.
- At iteration e, let $y'(e)$ (resp. $y(e)$) be new (resp. old) entry for e, then
$$\begin{aligned} y'(S_e) &= y(S_e \setminus \{e\}) + y'(e) \\ &= y(S_e \setminus \{e\}) + [y(e) + f(S_e) - y(S_e)] = f(S_e) \end{aligned} \tag{19.42}$$
So, S_e is tight for y'. It remains tight in further iterations since y doesn't decrease and it stays within P_f.
- Also, $E = T \cup \bigcup_{e \notin T} S_e$ is also tight, meaning the final y has $y \in B_f$. $\square$

Review from Lecture 12

The following slide repeats Theorem 13.4.2 from lecture 12 and is one of the most important theorems in submodular theory.

A polymatroid function's polyhedron is a polymatroid.

Theorem 19.7.1

Let f be a submodular function defined on subsets of E. For any $x \in \mathbb{R}^E$, we have:

$$\textit{rank}(x) = \max\left(y(E) : y \leq x, y \in P_f\right) = \min\left(x(A) + f(E \setminus A) : A \subseteq E\right) \tag{19.1}$$

Essentially the same theorem as Theorem ??, but note P_f rather than P_f^+. Taking $x = 0$ we get:

Corollary 19.7.2

Let f be a submodular function defined on subsets of E. We have:

$$\textit{rank}(0) = \max\left(y(E) : y \leq 0, y \in P_f\right) = \min\left(f(A) : A \subseteq E\right) \tag{19.2}$$

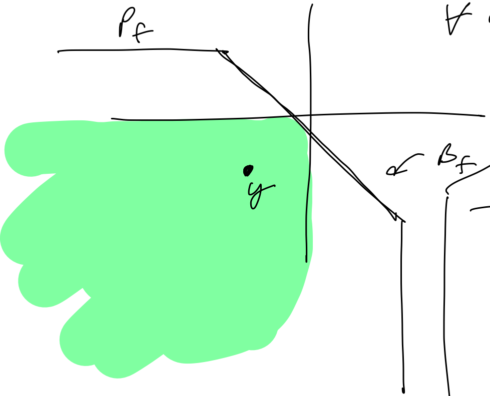
Modified max-min theorem

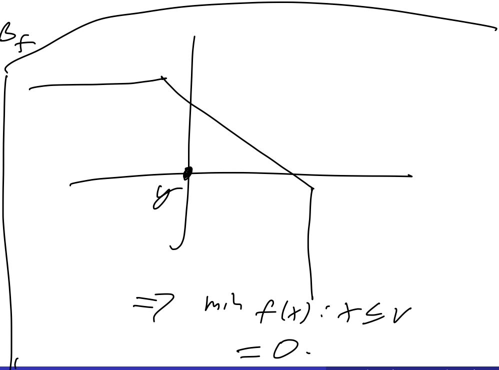
- Min-max theorem (Thm 13.4.2) restated for $x = 0$.

$$\max\{y(E) | y \in P_f, y \leq 0\} = \min\{f(X) | X \subseteq V\} \qquad (19.43)$$

$\forall\, y \in P_f,\ y \leq 0$

$y(E) \leq \min f(X) : X \subseteq V$





$\Rightarrow \min f(X) : X \subseteq V$
$= 0.$

Modified max-min theorem

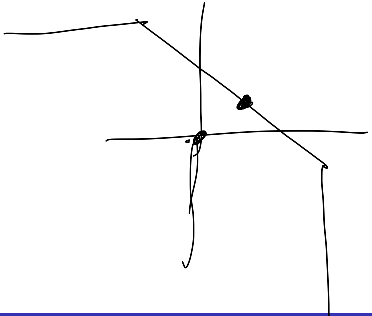
- Min-max theorem (Thm 13.4.2) restated for $x = 0$.

$$\max\{y(E) | y \in P_f, y \le 0\} = \min\{f(X) | X \subseteq V\} \tag{19.43}$$

Theorem 19.7.2 (Edmonds-1970)

$$\min\{f(X) | X \subseteq E\} = \max\left\{x^-(E) | x \in B_f\right\} \tag{19.44}$$

where $x^-(e) = \min\{x(e), 0\}$ *for* $e \in E$.



Modified max-min theorem

- Min-max theorem (Thm 13.4.2) restated for $x = 0$.

$$\max\{y(E) | y \in P_f, y \leq 0\} = \min\{f(X) | X \subseteq V\} \tag{19.43}$$

Theorem 19.7.2 (Edmonds-1970)

$$\min\{f(X) | X \subseteq E\} = \max\left\{x^-(E) | x \in B_f\right\} \tag{19.44}$$

where $x^-(e) = \min\{x(e), 0\}$ *for* $e \in E$.

Proof via the Lovász ext.

$$\min\{f(X) | X \subseteq E\} = \min_{w \in [0,1]^E} \breve{f}(w) = \min_{w \in [0,1]^E} \max_{x \in P_f} w^\intercal x \tag{19.45}$$

$$= \min_{w \in [0,1]^E} \max_{x \in B_f} w^\intercal x \tag{19.46}$$

$$= \max_{x \in B_f} \min_{w \in [0,1]^E} w^\intercal x \tag{19.47}$$

$$= \max_{x \in B_f} x^-(E) \tag{19.48}$$

$\square$

Alternate proof of modified max-min theorem

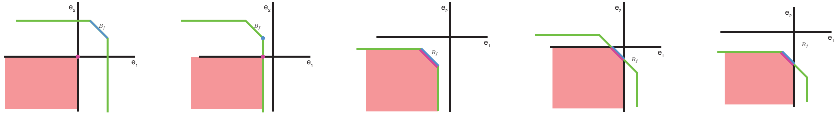
We start directly from Theorem 13.4.2.

$$\max\left(y(E) : y \leq 0, y \in P_f\right) = \min\left(f(A) : A \subseteq E\right) \tag{19.52}$$

Given $y \in \mathbb{R}^E$, define $y^- \in \mathbb{R}^E$ with $y^-(e) = \min\{y(e), 0\}$ for $e \in E$.

$$\begin{aligned}
\max\left(y(E) : y \leq 0, y \in P_f\right) &= \max\left(y^-(E) : y \leq 0, y \in P_f\right) && (19.53)\\
&= \max\left(y^-(E) : y \in P_f\right) && (19.54)\\
&= \max\left(y^-(E) : y \in B_f\right) && (19.55)
\end{aligned}$$

The first equality follows since $y \leq 0$. The second equality (together with the first) shown on following slide. The third equality follows since for any $x \in P_f$ there exists a $y \in B_f$ with $x \leq y$ (follows from Theorem 19.7.1).



Alternate proof of modified max-min theorem

Consider the following two problems for down-closed polyhedron P:

$$\max \sum_{e \in E} y(e) \quad (19.56a)$$
$$\text{s.t. } y \leq x \quad (19.56b)$$
$$y \in P \quad (19.56c)$$

$$\max \sum_{e \in E} \min(y(e), x(e)) \quad (19.57a)$$
$$\text{s.t. } y \in P \quad (19.57b)$$

- Solutions identical cost. Let y_1^* be l.h.s. OPT and y_2^* be r.h.s. OPT.
- Consider l.h.s. OPT y_1^* in r.h.s. evaluation and suppose it is worse (lower) than r.h.s. OPT:

$$\sum_{e \in E} \min(y_1^*(e), x(e)) < \sum_{e \in E} \min(y_2^*(e), x(e)) \quad (19.58)$$

Alternate proof of modified max-min theorem

Consider the following two problems for down-closed polyhedron P:

$$\max \sum_{e \in E} y(e) \tag{19.56a}$$
$$\text{s.t. } y \leq x \tag{19.56b}$$
$$y \in P \tag{19.56c}$$

$$\max \sum_{e \in E} \min(y(e), x(e)) \tag{19.57a}$$
$$\text{s.t. } y \in P \tag{19.57b}$$

- Solutions identical cost. Let y_1^* be l.h.s. OPT and y_2^* be r.h.s. OPT.
- Consider l.h.s. OPT y_1^* in r.h.s. evaluation and suppose it is worse (lower) than r.h.s. OPT:

$$\sum_{e \in E} \min(y_1^*(e), x(e)) < \sum_{e \in E} \min(y_2^*(e), x(e)) \tag{19.58}$$

- But the vector $\bar{y}_1^*$ with entries $\bar{y}_1^*(e) = \min(y_2^*(e), x(e))$ has $\bar{y}_1^*(e) \leq x(e)$ and $\bar{y}_1^* \in P$ since $y_2^* \in P$, $\bar{y}_1^* \leq y_2^*$, and P is down-closed.

Alternate proof of modified max-min theorem

Consider the following two problems for down-closed polyhedron P:

$$\max \sum_{e \in E} y(e) \tag{19.56a}$$
$$\text{s.t. } y \leq x \tag{19.56b}$$
$$y \in P \tag{19.56c}$$

$$\max \sum_{e \in E} \min(y(e), x(e)) \tag{19.57a}$$
$$\text{s.t. } y \in P \tag{19.57b}$$

- Solutions identical cost. Let y_1^* be l.h.s. OPT and y_2^* be r.h.s. OPT.
- Consider l.h.s. OPT y_1^* in r.h.s. evaluation and suppose it is worse (lower) than r.h.s. OPT:

$$\sum_{e \in E} \min(y_1^*(e), x(e)) < \sum_{e \in E} \min(y_2^*(e), x(e)) \tag{19.58}$$

- But the vector $\bar{y}_1^*$ with entries $\bar{y}_1^*(e) = \min(y_2^*(e), x(e))$ has $\bar{y}_1^*(e) \leq x(e)$ and $\bar{y}_1^* \in P$ since $y_2^* \in P$, $\bar{y}_1^* \leq y_2^*$, and P is down-closed.
- Thus, $\bar{y}_1^*$ is l.h.s. feasible but a better l.h.s. evaluation, a contradiction of the optimality of y_1^* for l.h.s.

Alternate proof of modified max-min theorem

Consider the following two problems for down-closed polyhedron P:

$$\max \sum_{e \in E} y(e) \tag{19.56a}$$
$$\text{s.t. } y \leq x \tag{19.56b}$$
$$y \in P \tag{19.56c}$$

$$\max \sum_{e \in E} \min(y(e), x(e)) \tag{19.57a}$$
$$\text{s.t. } y \in P \tag{19.57b}$$

- Solutions identical cost. Let y_1^* be l.h.s. OPT and y_2^* be r.h.s. OPT.
- Similarly, consider r.h.s. OPT y_2^* in l.h.s. evaluation and suppose it is worse (lower) than l.h.s. OPT

$$\sum_{e \in E} y_2^*(e) < \sum_{e \in E} y_1^*(e) \tag{19.58}$$

Alternate proof of modified max-min theorem

Consider the following two problems for down-closed polyhedron P:

$$\max \sum_{e \in E} y(e) \quad (19.56a)$$
$$\text{s.t. } y \leq x \quad (19.56b)$$
$$y \in P \quad (19.56c)$$

$$\max \sum_{e \in E} \min(y(e), x(e)) \quad (19.57a)$$
$$\text{s.t. } y \in P \quad (19.57b)$$

- Solutions identical cost. Let y_1^* be l.h.s. OPT and y_2^* be r.h.s. OPT.
- Similarly, consider r.h.s. OPT y_2^* in l.h.s. evaluation and suppose it is worse (lower) than l.h.s. OPT

$$\sum_{e \in E} y_2^*(e) < \sum_{e \in E} y_1^*(e) \quad (19.58)$$

- But the vector $\bar{y}_2^*$ with entries $\bar{y}_2^*(e) = y_1^*(e)$ has $\bar{y}_2^* \in P$ and since $\bar{y}_2^*(e) \leq x(e)$ for all e, we have

$$\sum_{e \in E} y_2^*(e) < \sum_{e \in E} y_1^*(e) = \sum_{e \in E} \bar{y}_2^*(e) = \sum_{e \in E} \min(\bar{y}_2^*(e), x(e)) \quad (19.59)$$

Alternate proof of modified max-min theorem

Consider the following two problems for down-closed polyhedron P:

$$\max \sum_{e \in E} y(e) \tag{19.56a}$$
$$\text{s.t. } y \leq x \tag{19.56b}$$
$$y \in P \tag{19.56c}$$

$$\max \sum_{e \in E} \min(y(e), x(e)) \tag{19.57a}$$
$$\text{s.t. } y \in P \tag{19.57b}$$

- Solutions identical cost. Let y_1^* be l.h.s. OPT and y_2^* be r.h.s. OPT.
- Similarly, consider r.h.s. OPT y_2^* in l.h.s. evaluation and suppose it is worse (lower) than l.h.s. OPT

$$\sum_{e \in E} y_2^*(e) < \sum_{e \in E} y_1^*(e) \tag{19.58}$$

- But the vector $\bar{y}_2^*$ with entries $\bar{y}_2^*(e) = y_1^*(e)$ has $\bar{y}_2^* \in P$ and since $\bar{y}_2^*(e) \leq x(e)$ for all e, we have

$$\sum_{e \in E} y_2^*(e) < \sum_{e \in E} y_1^*(e) = \sum_{e \in E} \bar{y}_2^*(e) = \sum_{e \in E} \min(\bar{y}_2^*(e), x(e)) \tag{19.59}$$

- Thus, we have r.h.s. feasible vector $\bar{y}_2^*$ strictly better than r.h.s. OPT contradicting the optimality of y_2^*.

Alternate proof of modified max-min theorem

Consider the following two problems for down-closed polyhedron P:

$$\max \sum_{e \in E} y(e) \qquad (19.56a)$$
$$\text{s.t. } y \le x \qquad (19.56b)$$
$$y \in P \qquad (19.56c)$$

$$\max \sum_{e \in E} \min(y(e), x(e)) \qquad (19.57a)$$
$$\text{s.t. } y \in P \qquad (19.57b)$$

- Solutions identical cost. Let y_1^* be l.h.s. OPT and y_2^* be r.h.s. OPT.
- Thus, l.h.s. and r.h.s. have identically valued solutions.

Alternate proof of modified max-min theorem

Consider the following two problems for down-closed polyhedron P:

$$\max \sum_{e \in E} y(e) \tag{19.56a}$$
$$\text{s.t. } y \leq x \tag{19.56b}$$
$$y \in P \tag{19.56c}$$

$$\max \sum_{e \in E} \min(y(e), x(e)) \tag{19.57a}$$
$$\text{s.t. } y \in P \tag{19.57b}$$

- Solutions identical cost. Let y_1^* be l.h.s. OPT and y_2^* be r.h.s. OPT.
- Thus, l.h.s. and r.h.s. have identically valued solutions.
- Hence, from previous slide, taking $x = 0$, $\max\left(y(E) : y \leq 0, y \in P_f\right) = \max\left(y^-(E) : y \in P_f\right) = \max\left(y^-(E) : y \in B_f\right)$

$\min \{w^\intercal x : x \in B_f\}$

- Recall that the greedy algorithm solves, for $w \in \mathbb{R}_+^E$

$$\max \{w^\intercal x | x \in P_f\} = \max \{w^\intercal x | x \in B_f\} \tag{19.58}$$

since for all $x \in P_f$, there exists $y \geq x$ with $y \in B_f$.

$\min \{w^\intercal x : x \in B_f\}$

- Recall that the greedy algorithm solves, for $w \in \mathbb{R}_+^E$

$$\max \{w^\intercal x | x \in P_f\} = \max \{w^\intercal x | x \in B_f\} \tag{19.58}$$

since for all $x \in P_f$, there exists $y \geq x$ with $y \in B_f$.

- For arbitrary $w \in \mathbb{R}^E$, we saw in Lecture 16 that the greedy algorithm will also solve:

$$\max \{w^\intercal x | x \in B_f\} \tag{19.59}$$

$\min\{w^\intercal x : x \in B_f\}$

- Recall that the greedy algorithm solves, for $w \in \mathbb{R}_+^E$

$$\max\{w^\intercal x | x \in P_f\} = \max\{w^\intercal x | x \in B_f\} \tag{19.58}$$

since for all $x \in P_f$, there exists $y \geq x$ with $y \in B_f$.

- For arbitrary $w \in \mathbb{R}^E$, we saw in Lecture 16 that the greedy algorithm will also solve:

$$\max\{w^\intercal x | x \in B_f\} \tag{19.59}$$

- Also, since $w \in \mathbb{R}^E$ is arbitrary, and since

$$\min\{w^\intercal x | x \in B_f\} = -\max\{-w^\intercal x | x \in B_f\} \tag{19.60}$$

the greedy algorithm using ordering $(e_1, e_2, \dots, e_m)$ such that

$$w(e_1) \leq w(e_2) \leq \dots \leq w(e_m) \tag{19.61}$$

will solve l.h.s. of Equation (19.60).

Greedy solves $\max\{w^\intercal x | x \in B_f\}$ for arbitrary $w \in \mathbb{R}^E$

Let $f(A)$ be arbitrary submodular function, and $f(A) = f'(A) - m(A)$ where f' is polymatroidal, and $w \in \mathbb{R}^E$.

$$
\begin{aligned}
\max\{w^\intercal x | x \in B_f\} &= \max\{w^\intercal x | x(A) \le f(A)\, \forall A, x(E) = f(E)\} \\
&= \max\left\{w^\intercal x | x(A) \le f'(A) - m(A)\, \forall A, x(E) = f'(E) - m(E)\right\} \\
&= \max\left\{w^\intercal x | x(A) + m(A) \le f'(A)\, \forall A, x(E) + m(E) = f'(E)\right\} \\
&= \max\Big\{w^\intercal x + w^\intercal m | \\
&\qquad x(A) + m(A) \le f'(A)\, \forall A, x(E) + m(E) = f'(E)\Big\} - w^\intercal m \\
&= \max\left\{w^\intercal y | y \in B_{f'}\right\} - w^\intercal m \\
&= w^\intercal y^* - w^\intercal m = w^\intercal (y^* - m)
\end{aligned}
$$

where $y = x + m$, so that $x^* = y^* - m$.
So y^* uses greedy algorithm with positive orthant $B_{f'}$. To show, we use Theorem **??** in Lecture 11, but we don't require $y \ge 0$, and don't stop when w goes negative to ensure $y^* \in B_{f'}$. Then when we subtract off m from y^*, we get solution to the original problem.