Announcements

- Welcome to the class!
- Weekly Office Hours: Wednesdays, 2:40-3:40pm, 10 minutes after class ends on Wednesdays.
- Class web page is at our web page (http://j.ee.washington.edu/~bilmes/classes/ee596a_fall_2012/)
This course will serve as an introduction to submodular functions including methods for their optimization, and how they have been (and can be) applied in many application domains.
Rough Outline

- Introduction to submodular functions, including definitions, real-world and contrived examples of submodular functions, properties, operations that preserve submodularity, submodular variants and special submodular functions, and computational properties.

- Background on submodular functions, including a brief overview of the theory of matroids and lattices.

- Polyhedral properties of submodular functions

- The Lovász extension of submodular functions. The Choquet integral.

- Submodular maximization algorithms under simple constraints, submodular cover problems, greedy algorithms, approximation guarantees $1 - \frac{1}{e}$
Rough Outline (cont. II)

- Submodular minimization algorithms, a history of submodular minimization, including both numerical and combinatorial algorithms, computational properties of these algorithms, and descriptions of both known results and currently open problems in this area.

- Submodular flow problems, the principle partition of a submodular function and its variants.

- Constrained optimization problems with submodular functions, including maximization and minimization problems with various constraints. An overview of recent problems addressed in the community.

- Applications of submodularity in computer vision, constraint satisfaction, game theory, information theory, norms, natural language processing, graphical models, and machine learning.
Useful Books

- Fujishige, “Submodular Functions and Optimization”, 2005
- Narayanan, “Submodular Functions and Electrical Networks”, 1997
- Schrijver, “Combinatorial Optimization”, 2003
- Additional readings that will be announced here.
Facts about the class

- Prerequisites: ideally knowledge in probability, statistics, convex optimization, and combinatorial optimization these will be reviewed as necessary. The course is open to students in all UW departments. Any questions, please contact me.

- Text: We will be drawing from the book by Satoru Fujishige entitled "Submodular Functions and Optimization" 2nd Edition, 2005, but we will also be reading research papers that will be posted here on this web page, especially for some of the application areas.

- Grades and Assignments: Grades will be based on a combination of a final project (35%), homeworks (45%), and the take home midterm exam (20%). There will be between 3-6 homeworks during the quarter.

- Final project: The final project will consist of a 4-page paper (conference style) and a final project presentation. The project must involve using/dealing mathematically with submodularity in some way or another.
Facts about the class

- Homework/midterm must be submitted electronically using our dropbox (https://catalyst.uw.edu/collectit/dropbox/bilmes/23873). PDF submissions only please. Photos of neatly hand written solutions, combined into one PDF, are fine.

- Lecture slides - are being prepared as we speak. I will try to have them up on the web page the night before each class. I will not only draw from the book but other sources which will be listed at the end of each set of slides.

- Slides from previous version of this class are at http://ssl.ee.washington.edu/~bilmes/ee595a_spring_2011/.
Almost all equations will have numbers.
Almost all equations will have numbers.

Equations will be numbered with lecture number, and number within lecture in the form \((\ell.j)\) where \(\ell\) is the lecture number and \(j\) is the \(j^{th}\) equation in lecture \(\ell\). For example,

\[
f(A) = f(V \setminus A) \tag{1.2}
\]
Other logistics

- Almost all equations will have numbers.
- Equations will be numbered with lecture number, and number within lecture in the form \((\ell.j)\) where \(\ell\) is the lecture number and \(j\) is the \(j^{th}\) equation in lecture \(\ell\). For example,

\[ f(A) = f(V \setminus A) \quad (1.2) \]

- Theorems, Lemmas, postulates, etc. will be numbered with \((\ell.s.j)\) where \(\ell\) is the lecture number, \(s\) is the section number, and \(j\) is the order within that section.
Other logistics

- Almost all equations will have numbers.
- Equations will be numbered with lecture number, and number within lecture in the form \((\ell.j)\) where \(\ell\) is the lecture number and \(j\) is the \(j^{th}\) equation in lecture \(\ell\). For example,

\[
f(A) = f(V \setminus A) \tag{1.2}
\]

- Theorems, Lemmas, postulates, etc. will be numbered with \((\ell.s.j)\) where \(\ell\) is the lecture number, \(s\) is the section number, and \(j\) is the order within that section.

**Theorem 1.1.1 (foo’s theorem)**

*foo*
Other logistics

- Almost all equations will have numbers.
- Equations will be numbered with lecture number, and number within lecture in the form \((\ell.j)\) where \(\ell\) is the lecture number and \(j\) is the \(j^{th}\) equation in lecture \(\ell\). For example,

\[
f(A) = f(V \setminus A) \tag{1.2}\]

- Theorems, Lemmas, postulates, etc. will be numbered with \((\ell.s.j)\) where \(\ell\) is the lecture number, \(s\) is the section number, and \(j\) is the order within that section.

**Theorem 1.1.1 (foo’s theorem)**

**foo**

- Exception to these rules is in the review sections, where theorems, equation, etc. (even if repeated) will have new reference numbers.
Outstanding Reading

- Read chapter 1 from the book.

Fujishige
Announcements, Assignments, and Reminders

- Please do use our discussion board (https://catalyst.uw.edu/gopost/board/bilmes/29948/) for all questions, comments, so that all will benefit from them being answered.
This is where each day we will be reviewing previous lecture material.
Discrete Optimization Problems

- Suppose we are given a finite set of objects $V$ of size $n = |V|$.
Discrete Optimization Problems

- Suppose we are given a finite set of objects $V$ of size $n = |V|$.
- We consider subsets of $V$. There are $2^n$ such subsets (denoted $2^V$).
Suppose we are given a finite set of objects $V$ of size $n = |V|$.

We consider subsets of $V$. There are $2^n$ such subsets (denoted $2^V$).

We have a function $f : 2^V \rightarrow \mathbb{R}$ that judges the quality (or value, or cost, or etc.) of each subset.
Suppose we are given a finite set of objects $V$ of size $n = |V|$.

We consider subsets of $V$. There are $2^n$ such subsets (denoted $2^V$).

We have a function $f : 2^V \rightarrow \mathbb{R}$ that judges the quality (or value, or cost, or etc.) of each subset.

We may be interested only in a subset of the set of possible subsets, namely $S \subseteq 2^V$. E.g., $S = \{S \subseteq V : |S| \leq k\}$. The set of sets $S$ might or might not itself be a function of $f$ (e.g., $S = \{S \subseteq V : f(S) \leq \alpha\}$.)
Discrete Optimization Problems

- Suppose we are given a finite set of objects $V$ of size $n = |V|$.
- We consider subsets of $V$. There are $2^n$ such subsets (denoted $2^V$).
- We have a function $f : 2^V \to \mathbb{R}$ that judges the quality (or value, or cost, or etc.) of each subset.
- We may be interested only in a subset of the set of possible subsets, namely $S \subseteq 2^V$. E.g., $S = \{ S \subseteq V : |S| \leq k \}$. The set of sets $S$ might or might not itself be a function of $f$ (e.g., $S = \{ S \subseteq V : f(S) \leq \alpha \}$).
- The general discrete optimization problem we consider here is:

$$\begin{align*}
\text{maximize} & \quad f(S) \\
\text{subject to} & \quad S \in S
\end{align*} \tag{1.3}$$
Discrete Optimization Problems

- Suppose we are given a finite set of objects $V$ of size $n = |V|$.
- We consider subsets of $V$. There are $2^n$ such subsets (denoted $2^V$).
- We have a function $f : 2^V \to \mathbb{R}$ that judges the quality (or value, or cost, or etc.) of each subset.

We may be interested only in a subset of the set of possible subsets, namely $S \subseteq 2^V$. E.g., $S = \{ S \subseteq V : |S| \leq k \}$. The set of sets $S$ might or might not itself be a function of $f$ (e.g., $S = \{ S \subseteq V : f(S) \leq \alpha \}$).

The general discrete optimization problem we consider here is:

$$\maximize_{S \subseteq 2^V} f(S) \quad \text{subject to} \quad S \in S \quad (1.3)$$

- Ignoring how complex and general this problem can be for the moment, let's consider some possible applications.
We are given a finite set $V$ of $n$ elements and a set of subsets $\mathcal{V} = \{V_1, V_2, \ldots, V_m\}$ of $m$ subsets of $V$, so that $V_i \subseteq V$. 
Set Cover and Maximum Coverage

We are given a finite set $V$ of $n$ elements and a set of subsets $\mathcal{V} = \{V_1, V_2, \ldots, V_m\}$ of $m$ subsets of $V$, so that $V_i \subseteq V$.

Our goal in Set Cover is to choose the smallest subset $A \subseteq [m] \triangleq \{1, \ldots, m\}$ such that $\bigcup_{a \in A} V_a = V$. 

The goal in Maximum Coverage is, given an integer $k \leq m$, select $k$ subsets, say $\{a_1, a_2, \ldots, a_k\}$ with $a_i \in [m]$ such that $|\bigcup_{i=1}^k V_{a_i}|$ is maximized.

Both Set Cover and Maximum Coverage are well known to be NP-hard, but have a fast greedy approximation algorithm.
We are given a finite set $V$ of $n$ elements and a set of subsets $\mathcal{V} = \{V_1, V_2, \ldots, V_m\}$ of $m$ subsets of $V$, so that $V_i \subseteq V$.

Our goal in \textsc{set cover} is to choose the smallest subset $A \subseteq [m] \triangleq \{1, \ldots, m\}$ such that $\bigcup_{a \in A} V_a = V$.

The goal in \textsc{maximum coverage} is, given an integer $k \leq m$, select $k$ subsets, say $\{a_1, a_2, \ldots, a_k\}$ with $a_i \in [m]$ such that $|\bigcup_{i=1}^{k} V_{a_i}|$ is maximized.
**Set Cover and Maximum Coverage**

- We are given a finite set $V$ of $n$ elements and a set of subsets $\mathcal{V} = \{V_1, V_2, \ldots, V_m\}$ of $m$ subsets of $V$, so that $V_i \subseteq V$.

- Our goal in **Set Cover** is to choose the smallest subset $A \subseteq [m] \triangleq \{1, \ldots, m\}$ such that $\bigcup_{a \in A} V_a = V$.

- The goal in **Maximum Coverage** is, given an integer $k \leq m$, select $k$ subsets, say $\{a_1, a_2, \ldots, a_k\}$ with $a_i \in [m]$ such that $|\bigcup_{i=1}^{k} V_{a_i}|$ is maximized.

- Both **Set Cover** and **Maximum Coverage** are well known to be NP-hard, but have a fast greedy approximation algorithm.
**Definition 1.3.1**

vertex cover A vertex cover (an “vertex-based cover of edges”) in graph $G = (V, E)$ is a set $S \subseteq V(G)$ of vertices such that every edge in $G$ is incident to at least one vertex in $S$.

- Let $I(S)$ be the number of edges incident to vertex set $S$. Then we wish to find the smallest set $S \subseteq V$ subject to $I(S) = |E|$.  

**Definition 1.3.2**

edge cover A edge cover (an “edge-based cover of vertices”) in graph $G = (V, E)$ is a set $F \subseteq E(G)$ of edges such that every vertex in $G$ is incident to at least one edge in $F$.

- Let $|V|(F)$ be the number of vertices incident to edge set $F$. Then we wish to find the smallest set $F \subseteq E$ subject to $|V|(F) = |V|$.  

Prof. Jeff Bilmes  
EE596A/Fall 2012/Submodularity – Lecture 1 - September 26th, 2012  
page 18
Sensor Placement

- Given an environment, there is a set $V$ of candidate locations for placement of a sensor (e.g., temperature, gas, audio, video, bacteria or other environmental contaminant, etc.).
Sensor Placement

- Given an environment, there is a set $V$ of candidate locations for placement of a sensor (e.g., temperature, gas, audio, video, bacteria or other environmental contaminant, etc.).

- We have a function $f(S)$ that measures the “coverage” of any given set $S$ of sensor placement decisions. Then $f(V)$ is maximum possible coverage.
Sensor Placement

- Given an environment, there is a set $V$ of candidate locations for placement of a sensor (e.g., temperature, gas, audio, video, bacteria or other environmental contaminant, etc.).

- We have a function $f(S)$ that measures the “coverage” of any given set $S$ of sensor placement decisions. Then $f(V)$ is maximum possible coverage.

- One possible goal: choose smallest set $S$ such that $f(S) = f(V)$. 

---

Prof. Jeff Bilmes
EE596A/Fall 2012/Submodularity – Lecture 1 - September 26th, 2012 

---

page 19
Sensor Placement

- Given an environment, there is a set $V$ of candidate locations for placement of a sensor (e.g., temperature, gas, audio, video, bacteria or other environmental contaminant, etc.).

- We have a function $f(S)$ that measures the “coverage” of any given set $S$ of sensor placement decisions. Then $f(V)$ is maximum possible coverage.

- One possible goal: choose smallest set $S$ such that $f(S) = f(V)$.

- Another possible goal: choose size at most $k$ set $S$ such that $f(S)$ is maximized.
Sensor Placement

- Given an environment, there is a set $V$ of candidate locations for placement of a sensor (e.g., temperature, gas, audio, video, bacteria or other environmental contaminant, etc.).

- We have a function $f(S)$ that measures the “coverage” of any given set $S$ of sensor placement decisions. Then $f(V)$ is maximum possible coverage.

- One possible goal: choose smallest set $S$ such that $f(S) = f(V)$.

- Another possible goal: choose size at most $k$ set $S$ such that $f(S)$ is maximized.

- Environment could be a floor of a building, water network, monitored ecological preservation.
Sensor Placement in Buildings

An example of a room layout. Should possible to determine temperature at all points in the room. Sensors cannot sense beyond wall (thick black line) boundaries.
Example sensor placement using small range cheap sensors (located at red dots).
Example sensor placement using large range expensive sensors (located at red dots).
Example sensor placement using mixed range sensors (located at red dots).
Graph Cut Problems

- **Minimum cut**: Given a graph $G = (V, E)$, find a set of vertices $S \subseteq V$ that minimize the cut (set of edges) between $S$ and $V \setminus S$. 
Graph Cut Problems

- **Minimum cut**: Given a graph $G = (V, E)$, find a set of vertices $S \subseteq V$ that minimize the cut (set of edges) between $S$ and $V \setminus S$.

- **Maximum cut**: Given a graph $G = (V, E)$, find a set of vertices $S \subseteq V$ that minimize the cut (set of edges) between $S$ and $V \setminus S$. 

Weighted versions.
Many examples of this, we will see more later.
Graph Cut Problems

- **Minimum cut**: Given a graph $G = (V, E)$, find a set of vertices $S \subseteq V$ that minimize the cut (set of edges) between $S$ and $V \setminus S$.
- **Maximum cut**: Given a graph $G = (V, E)$, find a set of vertices $S \subseteq V$ that minimize the cut (set of edges) between $S$ and $V \setminus S$.
- **Weighted versions**.
Graph Cut Problems

- **Minimum cut**: Given a graph $G = (V, E)$, find a set of vertices $S \subseteq V$ that minimize the cut (set of edges) between $S$ and $V \setminus S$.
- **Maximum cut**: Given a graph $G = (V, E)$, find a set of vertices $S \subseteq V$ that minimize the cut (set of edges) between $S$ and $V \setminus S$.
- Weighted versions.
- Many examples of this, we will see more later.
Facility/Plant Location (uncapacitated)

- Core problem in operations research and strong original motivation for submodular functions.
- Goal: as efficiently as possible, place “facilities” (factories) at certain locations to satisfy sites (at all locations) having various demands.
Facility/Plant Location (uncapacitated)

- Core problem in operations research and strong original motivation for submodular functions.
- Goal: as efficiently as possible, place “facilities” (factories) at certain locations to satisfy sites (at all locations) having various demands.
Facility/Plant Location (uncapacitated)

- Core problem in operations research and strong original motivation for submodular functions.
- Goal: as efficiently as possible, place “facilities” (factories) at certain locations to satisfy sites (at all locations) having various demands.
Facility/Plant Location (uncapacitated)

- Let $F = \{1, \ldots, f\}$ be a set of possible factory/plant locations for facilities to be built.
- $S = \{1, \ldots, s\}$ is a set of sites needing to be serviced (e.g., cities, clients).
- Let $c_{ij}$ be the “benefit” (e.g., $1/c_{ij}$ is the cost) of servicing site $i$ with facility location $j$.
- Let $m_j$ be the benefit (e.g., either $1/m_j$ is the cost or $-m_j$ is the cost) to build a plant at location $j$.
- Each site needs to be serviced by only one plant but no less than one.
- Define $f(A)$ as the “delivery benefit” plus “construction benefit” when the locations $A \subseteq F$ are to be constructed.
- We can define $f(A) = \sum_{j \in A} m_j + \sum_{i \in F} \max_{j \in A} c_{ij}$.
- Goal is to find a set $A$ that maximizes $f(A)$ (the benefit) placing a bound on the number of plants $A$ (e.g., $|A| \leq k$).
Information Gain and Feature Selection

- Task: pattern recognition based on (at most) features $X_V$ to predict random variable $Y$. True model is $p(Y, X_V)$, where $V$ is a finite set of feature indices.

Information gain is defined as:

$$f(A) = I(Y; X_A) = H(Y) - H(Y | X_A) = H(X_A) - H(X_A | Y)$$

(1.4)

So goal is to find a subset $A$ that has high information gain.

Applicable not only in pattern recognition, but in the sensor coverage problem as well, where $Y$ is whatever question we wish to ask about the room.
Information Gain and Feature Selection

- Task: pattern recognition based on (at most) features $X_V$ to predict random variable $Y$. True model is $p(Y, X_V)$, where $V$ is a finite set of feature indices.

- Given subset of features $A \subseteq V$, prediction based on $p(y|x_A)$.
Information Gain and Feature Selection

- Task: pattern recognition based on (at most) features $X_V$ to predict random variable $Y$. True model is $p(Y, X_V)$, where $V$ is a finite set of feature indices.
- Given subset of features $A \subseteq V$, prediction based on $p(y|x_A)$.
- Goal: choose the smallest set of features that retains accuracy.
Information Gain and Feature Selection

- Task: pattern recognition based on (at most) features $X_V$ to predict random variable $Y$. True model is $p(Y, X_V)$, where $V$ is a finite set of feature indices.
- Given subset of features $A \subseteq V$, prediction based on $p(y|x_A)$.
- Goal: choose the smallest set of features that retains accuracy.
- Information gain is defined as:

$$f(A) = I(Y; X_A) = H(Y) - H(Y|X_A) = H(X_A) - H(X_A|Y)$$ (1.4)
Task: pattern recognition based on (at most) features $X_V$ to predict random variable $Y$. True model is $p(Y, X_V)$, where $V$ is a finite set of feature indices.

Given subset of features $A \subseteq V$, prediction based on $p(y|x_A)$.

Goal: choose the smallest set of features that retains accuracy.

Information gain is defined as:

$$f(A) = I(Y; X_A) = H(Y) - H(Y|X_A) = H(X_A) - H(X_A|Y)$$

So goal is to find a subset $A$ that has high information gain.
Information Gain and Feature Selection

- Task: pattern recognition based on (at most) features $X_V$ to predict random variable $Y$. True model is $p(Y, X_V)$, where $V$ is a finite set of feature indices.

- Given subset of features $A \subseteq V$, prediction based on $p(y|x_A)$.

- Goal: choose the smallest set of features that retains accuracy.

- Information gain is defined as:

  $$f(A) = I(Y; X_A) = H(Y) - H(Y|X_A) = H(X_A) - H(X_A|Y)$$

  (1.4)

- So goal is to find a subset $A$ that has high information gain.

- Applicable not only in pattern recognition, but in the sensor coverage problem as well, where $Y$ is whatever question we wish to ask about the room.
Monge Matrices

- $m \times n$ matrices $C = [c_{ij}]_{ij}$ are called Monge matrices if they satisfy the Monge property, namely:

$$c_{ij} + c_{rs} \leq c_{is} + c_{rj}$$

(1.5)

for all $1 \leq i < r \leq m$ and $1 \leq j < s \leq n$. 

Monge Matrices

- $m \times n$ matrices $C = [c_{ij}]_{ij}$ are called Monge matrices if they satisfy the Monge property, namely:

$$c_{ij} + c_{rs} \leq c_{is} + c_{rj} \quad (1.5)$$

for all $1 \leq i < r \leq m$ and $1 \leq j < s \leq n$.

- Consider four elements of the matrix:
Monge Matrices

- Can generate a Monge matrix from a convex polygon - delete two segments, separately number vertices on each chain, distances $c_{ij}$ satisfy Monge property.
Can generate a Monge matrix from a convex polygon - delete two segments, separately number vertices on each chain, distances $c_{ij}$ satisfy Monge property.
Monge Matrices

- Can generate a Monge matrix from a convex polygon - delete two segments, separately number vertices on each chain, distances $c_{ij}$ satisfy Monge property.
A model of Influence in Social Networks

- Given a graph $G = (V, E)$, each $v \in V$ corresponds to a person, to each $v$ we have an activation function $f_v : 2^V \rightarrow [0, 1]$ dependent only on its neighbors. I.e., $f_v(A) = f_v(A \cap \Gamma(v))$. 
A model of Influence in Social Networks

- Given a graph $G = (V, E)$, each $v \in V$ corresponds to a person, to each $v$ we have an activation function $f_v : 2^V \rightarrow [0, 1]$ dependent only on its neighbors. I.e., $f_v(A) = f_v(A \cap \Gamma(v))$.

- Goal - Viral Marketing: find a small subset $S \subseteq V$ of individuals to directly influence, and thus indirectly influence the greatest number of possible other individuals (via the social network $G$).
A model of Influence in Social Networks

- Given a graph $G = (V, E)$, each $v \in V$ corresponds to a person, to each $v$ we have an activation function $f_v : 2^V \rightarrow [0, 1]$ dependent only on its neighbors. I.e., $f_v(A) = f_v(A \cap \Gamma(v))$.

- Goal - Viral Marketing: find a small subset $S \subseteq V$ of individuals to directly influence, and thus indirectly influence the greatest number of possible other individuals (via the social network $G$).

- We define a function $f : 2^V \rightarrow \mathbb{Z}^+$ that models the ultimate influence of an initial set $S$ of nodes based on the following iterative process: At each step, a given set of nodes $S$ are activated, and we activate new nodes $v \in V \setminus S$ if $f_v(S) \geq U[0, 1]$ (where $U[0, 1]$ is a uniform random number between 0 and 1).
Let $V$ be a group of individuals. How valuable to you is a given friend $v \in V$?

It depends on how many friends you have.

Given a group of friends $S \subseteq V$, can you valuate them with a function $f(S)$ and how?

Let $f(S)$ be the value of the set of friends $S$. Is submodular or supermodular a good model?
The value of a friend

• Let $V$ be a group of individuals. How valuable to you is a given friend $v \in V$?

• It depends on how many friends you have.

• Given a group of friends $S \subseteq V$, can you valuate them with a function $f(S)$ an how?

• Let $f(S)$ be the value of the set of friends $S$. Is submodular or supermodular a good model?
The value of a friend

- Let $V$ be a group of individuals. How valuable to you is a given friend $v \in V$?
- It depends on how many friends you have.
- Given a group of friends $S \subseteq V$, can you valuate them with a function $f(S)$ an how?
- Let $f(S)$ be the value of the set of friends $S$. Is submodular or supermodular a good model?
Information Cascades

- How to model the flow of information from its source to the point it reaches users — here, we use information in its common sense (like news events).
Information Cascades

- How to model the flow of information from its source to the point it reaches users — here, we use information in its common sense (like news events).
- How to find the most influential sources, the ones that often set off cascades, which are large waves of information flow?
Information Cascades

- How to model the flow of information from its source to the point it reaches users — here, we use information in its common sense (like news events).
- How to find the most influential sources, the ones that often set off cascades, which are large waves of information flow?
- There can be one seed source or many.
Diversity Functions

Given a set $V$ of items, how do we choose a subset $S \subseteq V$ that is as diverse as possible, with perhaps constraints on $S$ such as its size.
Diversity Functions

- Given a set $V$ of items, how do we choose a subset $S \subseteq V$ that is as diverse as possible, with perhaps constraints on $S$ such as its size.
- How do we choose the smallest set $S$ that maintains a given quality of diversity?
Diversity Functions

- Given a set $V$ of items, how do we choose a subset $S \subseteq V$ that is as diverse as possible, with perhaps constraints on $S$ such as its size.
- How do we choose the smallest set $S$ that maintains a given quality of diversity?
- Goal of diversity: ensure proper representation in chosen set that, say otherwise in a random sample, would lead to poor representation of normally underrepresented groups.
Given a graphical model $G = (V, E)$ and a family of probability distributions $p \in \mathcal{F}(G, \mathcal{M})$ that factor w.r.t. that distribution.
Graphical Model Tree Distributions

- Given a graphical model $G = (V, E)$ and a family of probability distributions $p \in \mathcal{F}(G, \mathcal{M})$ that factor w.r.t. that distribution.
- Find the closest distribution $p_t$ to $p$ subject to $p_t$ factoring w.r.t. some tree $T = (V, F)$. 
Graphical Model Tree Distributions

- Given a graphical model $G = (V, E)$ and a family of probability distributions $p \in \mathcal{F}(G, \mathcal{M})$ that factor w.r.t. that distribution.
- Find the closest distribution $p_t$ to $p$ subject to $p_t$ factoring w.r.t. some tree $T = (V, F)$.
- I.e., optimization problem

$$\begin{align*}
\text{minimize} & \quad D(p \| p_t) \\
\text{subject to} & \quad p_t \text{ factors w.r.t. the tree } T \text{ (i.e., } p_t \in \mathcal{F}(T, \mathcal{M})).
\end{align*}$$

(1.6)
Given a graphical model $G = (V, E)$ and a family of probability distributions $p \in F(G, M)$ that factor w.r.t. that distribution.

Find the closest distribution $p_t$ to $p$ subject to $p_t$ factoring w.r.t. some tree $T = (V, F)$.

I.e., optimization problem

$$
\text{minimize} \quad D(p \| p_t) \\
\text{subject to} \quad p_t \text{ factors w.r.t. the tree } T \text{ (i.e., } p_t \in F(T, M)) \quad (1.6)
$$

Discrete problem: Choose the right set of edges from $E$ that make up a tree (i.e., find a spanning sub-tree of best quality).
The figure below represents the sentences of a document.
Extractive Document Summarization

- We extract sentences (green) as a summary of the full document.
Extractive Document Summarization

- We extract sentences (green) as a summary of the full document.
We extract sentences (green) as a summary of the full document.

The summary on the left is a subset of the summary on the right.
Extractive Document Summarization

- We extract sentences (green) as a summary of the full document.

- The summary on the left is a subset of the summary on the right.
- Consider adding a new (blue) sentence to each of the two summaries.
We extract sentences (green) as a summary of the full document. The summary on the left is a subset of the summary on the right.

Consider adding a new (blue) sentence to each of the two summaries. The marginal (incremental) benefit of adding the new (blue) sentence to the smaller (left) summary is no more than the marginal benefit of adding the new sentence to the larger (right) summary.

\[ \subset \]

\[ \cap \]
Extractive Document Summarization

- We extract sentences (green) as a summary of the full document

\[ \begin{align*}
\text{The summary on the left is a subset of the summary on the right.} \\
\text{Consider adding a new (blue) sentence to each of the two summaries.} \\
\text{The marginal (incremental) benefit of adding the new (blue) sentence to the smaller (left) summary is no more than the marginal benefit of adding the new sentence to the larger (right) summary.} \\
\text{diminishing returns} \leftrightarrow \text{submodularity}
\end{align*} \]
Web search and information retrieval

- A web search is a form of summarization based on query.
- Goal of a web search engine is to produce a ranked list of web pages that, conditioned on the text query entered, summarizes the most important links on the web.
- Information retrieval (the science of automatically acquiring information), book and music recommendation systems —
- Overall goal: user should quickly find information that is informative, concise, accurate, relevant (to the user’s needs), and comprehensive.
Active Learning and Semi-Supervised Learning

- Given training data \( \mathcal{D}_V = \{(x_i, y_i)\}_{i \in V} \) of \((x, y)\) pairs where \(x\) is a query (data item) and \(y\) is an answer (label), goal is to learn a good mapping \(y = h(x)\).
Active Learning and Semi-Supervised Learning

- Given training data \( \mathcal{D}_V = \{(x_i, y_i)\}_{i \in V} \) of \((x, y)\) pairs where \(x\) is a query (data item) and \(y\) is an answer (label), goal is to learn a good mapping \(y = h(x)\).

- Often, getting \(y\) is time-consuming, expensive, and error prone (manual transcription, Amazon Turk, etc.)
Active Learning and Semi-Supervised Learning

- Given training data $\mathcal{D}_V = \{(x_i, y_i)\}_{i \in V}$ of $(x, y)$ pairs where $x$ is a query (data item) and $y$ is an answer (label), goal is to learn a good mapping $y = h(x)$.

- Often, getting $y$ is time-consuming, expensive, and error prone (manual transcription, Amazon Turk, etc.)

- Batch active learning: choose a subset $S \subset V$ so that only the labels $\{y_i\}_{i \in S}$ should be acquired.
Active Learning and Semi-Supervised Learning

- Given training data $D_V = \{(x_i, y_i)\}_{i \in V}$ of $(x, y)$ pairs where $x$ is a query (data item) and $y$ is an answer (label), goal is to learn a good mapping $y = h(x)$.
- Often, getting $y$ is time-consuming, expensive, and error prone (manual transcription, Amazon Turk, etc.)
- Batch active learning: choose a subset $S \subset V$ so that only the labels $\{y_i\}_{i \in S}$ should be acquired.
- Adaptive active learning: choose a policy whereby we choose an $i_1 \in V$, get the label $y_{i_1}$, choose another $i_2 \in V$, get label $y_{i_2}$, where each chose can be based on previously acquired labels.
Active Learning and Semi-Supervised Learning

- Given training data $\mathcal{D}_V = \{(x_i, y_i)\}_{i \in V}$ of $(x, y)$ pairs where $x$ is a query (data item) and $y$ is an answer (label), goal is to learn a good mapping $y = h(x)$.

- Often, getting $y$ is time-consuming, expensive, and error prone (manual transcription, Amazon Turk, etc.)

- Batch active learning: choose a subset $S \subset V$ so that only the labels $\{y_i\}_{i \in S}$ should be acquired.

- Adaptive active learning: choose a policy whereby we choose an $i_1 \in V$, get the label $y_{i_1}$, choose another $i_2 \in V$, get label $y_{i_2}$, where each chose can be based on previously acquired labels.

- Semi-supervised (transductive) learning: Once we have $\{y_i\}_{i \in S}$, infer the remaining labels $\{y_i\}_{i \in V \setminus S}$.
Markets: Supply Side Economies of scale

- Economies of Scale: Many goods and services can be produced at a much lower per-unit cost only if they are produced in very large quantities.
Markets: Supply Side Economies of scale

- **Economies of Scale**: Many goods and services can be produced at a much lower per-unit cost only if they are produced in very large quantities.

- **The profit margin for producing a unit of goods improved as more of those goods are created.**
Markets: Supply Side Economies of scale

- Economies of Scale: Many goods and services can be produced at a much lower per-unit cost only if they are produced in very large quantities.
- The profit margin for producing a unit of goods improved as more of those goods are created.
- If you already make a good, making a similar good is easier than if you start from scratch (e.g., Apple making both iPod and iPhone).
Economies of Scale: Many goods and services can be produced at a much lower per-unit cost only if they are produced in very large quantities.

The profit margin for producing a unit of goods improved as more of those goods are created.

If you already make a good, making a similar good is easier than if you start from scratch (e.g., Apple making both iPod and iPhone).

An argument in favor of free trade is that it opens up larger markets to firms in (especially otherwise small markets), thereby enabling better economies of scale, and hence greater efficiency (lower costs and resources per unit of good produced).
Supply Side Economies of scale: Cost of manufacturing a set of items

- Let $V$ be a set of possible items that a company might possibly wish to manufacture, and let $f(S)$ for $S \subseteq V$ be the cost to that company to manufacture subset $S$. 

Ex: $V$ might be colors of paint in a paint manufacturer: green, red, blue, yellow, white, etc. Producing green when you are already producing yellow and blue is probably cheaper than if you were only producing some other colors. 

$f(\text{green}, \text{blue}, \text{yellow}) - f(\text{blue}, \text{yellow}) < f(\text{green}, \text{blue}) - f(\text{blue})$ (1.7)

So a submodular function would be a good model.
Supply Side Economies of scale: Cost of manufacturing a set of items

- Let $V$ be a set of possible items that a company might possibly wish to manufacture, and let $f(S)$ for $S \subseteq V$ be the cost to that company to manufacture subset $S$.
- Ex: $V$ might be colors of paint in a paint manufacturer: green, red, blue, yellow, white, etc.
Supply Side Economies of scale: Cost of manufacturing a set of items

- Let $V$ be a set of possible items that a company might possibly wish to manufacture, and let $f(S)$ for $S \subseteq V$ be the cost to that company to manufacture subset $S$.

- Ex: $V$ might be colors of paint in a paint manufacturer: green, red, blue, yellow, white, etc.

- Producing green when you are already producing yellow and blue is probably cheaper than if you were only producing some other colors.

\[
f(\text{green, blue, yellow}) - f(\text{blue, yellow}) \leq f(\text{green, blue}) - f(\text{blue})
\]  
(1.7)
Demand side Economies of Scale: Network Externalities

- consumers of a good derive positive value when size of the market increases.
Demand side Economies of Scale: Network Externalities

- consumers of a good derive positive value when size of the market increases.
- the value of a network to a user depends on the number of other users in that network.
Demand side Economies of Scale: Network Externalities

- consumers of a good derive positive value when size of the market increases.
- the value of a network to a user depends on the number of other users in that network.
- Hence, network externalities are a form of “demand” economies of scale
Demand side Economies of Scale: Network Externalities

- consumers of a good derive positive value when size of the market increases.
- the value of a network to a user depends on the number of other users in that network.
- Hence, network externalities are a form of “demand” economies of scale.
- “value” in this case can be seen as a “willingness-to-pay” for the service (WTP).
Demand side Economies of Scale: Network Externalities

- consumers of a good derive positive value when size of the market increases.
- the value of a network to a user depends on the number of other users in that network.
- Hence, network externalities are a form of “demand” economies of scale
- “value” in this case can be seen as a “willingness-to-pay” for the service (WTP)
- WTP tends to increase but then saturate (like a logistic function)
Demand side Economies of Scale: Network Externalities

- consumers of a good derive positive value when size of the market increases.
- the value of a network to a user depends on the number of other users in that network.
- Hence, network externalities are a form of “demand” economies of scale.
- “value” in this case can be seen as a “willingness-to-pay” for the service (WTP).
- WTP tends to increase but then saturate (like a logistic function).
- Given network externalities, a consumer in today’s market cares also about the future success of the product and competing products.
Demand side Economies of Scale: Network Externalities

- consumers of a good derive positive value when size of the market increases.
- the value of a network to a user depends on the number of other users in that network.
- Hence, network externalities are a form of “demand” economies of scale
- “value” in this case can be seen as a “willingness-to-pay” for the service (WTP)
- WTP tends to increase but then saturate (like a logistic function)
- Given network externalities, a consumer in today’s market cares also about the future success of the product and competing products.
- If the good is durable (or there is human capital investment), the total benefits derived from a good will depend on the number of consumers who adopt compatible products in the future.
Positive Network Externalities

- railroad - standard rail format and shared access
Positive Network Externalities

- railroad - standard rail format and shared access
- The telephone, who wants to talk by phone only to oneself?
Positive Network Externalities

- railroad - standard rail format and shared access
- The telephone, who wants to talk by phone only to oneself?
- the internet, more valuable per person the more people use it.
Positive Network Externalities

- railroad - standard rail format and shared access
- The telephone, who wants to talk by phone only to oneself?
- the internet, more valuable per person the more people use it.
- ebooks (the more people comment, the better it gets)
Positive Network Externalities

- railroad - standard rail format and shared access
- The telephone, who wants to talk by phone only to oneself?
- the internet, more valuable per person the more people use it.
- ebooks (the more people comment, the better it gets)
- social network sites: facebook more valuable with everyone online
Positive Network Externalities

- railroad - standard rail format and shared access
- The telephone, who wants to talk by phone only to oneself?
- the internet, more valuable per person the more people use it.
- ebooks (the more people comment, the better it gets)
- social network sites: facebook more valuable with everyone online
- online education, Coursera, edX, etc. – with many people simultaneously taking a class, all gain due to richer peer discussions due to greater pool of well matched study groups, more simultaneous similar questions/problems that are asked, leading to more efficient learning.
Positive Network Externalities

- railroad - standard rail format and shared access
- The telephone, who wants to talk by phone only to oneself?
- the internet, more valuable per person the more people use it.
- ebooks (the more people comment, the better it gets)
- social network sites: facebook more valuable with everyone online
- online education, Coursera, edX, etc. – with many people simultaneously taking a class, all gain due to richer peer discussions due to greater pool of well matched study groups, more simultaneous similar questions/problems that are asked, leading to more efficient learning.
- Software, Microsoft office, smartphone apps: more people use it more people report bugs, help with problems, software gets better for every user.
Positive Network Externalities

- railroad - standard rail format and shared access
- The telephone, who wants to talk by phone only to oneself?
- the internet, more valuable per person the more people use it.
- ebooks (the more people comment, the better it gets)
- social network sites: facebook more valuable with everyone online
- online education, Coursera, edX, etc. – with many people simultaneously taking a class, all gain due to richer peer discussions due to greater pool of well matched study groups, more simultaneous similar questions/problems that are asked, leading to more efficient learning.
- Software, Microsoft office, smartphone apps: more people use it more people report bugs, help with problems, software gets better for every user.
- wikipedia
Positive Network Externalities

- railroad - standard rail format and shared access
- The telephone, who wants to talk by phone only to oneself?
- the internet, more valuable per person the more people use it.
- ebooks (the more people comment, the better it gets)
- social network sites: facebook more valuable with everyone online
- online education, Coursera, edX, etc. – with many people simultaneously taking a class, all gain due to richer peer discussions due to greater pool of well matched study groups, more simultaneous similar questions/problems that are asked, leading to more efficient learning.
- Software, Microsoft office, smartphone apps: more people use it more people report bugs, help with problems, software gets better for every user.
- wikipedia
- any widely used standard (job training now is useful in the future)
Positive Network Externalities

- railroad - standard rail format and shared access
- The telephone, who wants to talk by phone only to oneself?
- the internet, more valuable per person the more people use it.
- ebooks (the more people comment, the better it gets)
- social network sites: facebook more valuable with everyone online
- online education, Coursera, edX, etc. – with many people simultaneously taking a class, all gain due to richer peer discussions due to greater pool of well matched study groups, more simultaneous similar questions/problems that are asked, leading to more efficient learning.
- Software, Microsoft office, smartphone apps: more people use it more people report bugs, help with problems, software gets better for every user.
- wikipedia
- any widely used standard (job training now is useful in the future)
- Concepts like the “tipping point”, and “winner take all” markets.
Other Network Externalities

No Network Externalities

- food/drink - (should be) independent of how many others are eating the type of food.
Other Network Externalities

No Network Externalities

- food/drink - (should be) independent of how many others are eating the type of food.
- Music - your enjoyment should be independent of others enjoyment.
Other Network Externalities

No Network Externalities

- food/drink - (should be) independent of how many others are eating the type of food.
- Music - your enjoyment should be independent of others enjoyment.
Other Network Externalities

No Network Externalities

- food/drink - (should be) independent of how many others are eating the type of food.
- Music - your enjoyment should be independent of others enjoyment.

Negative Network Externalities

- clothing
Other Network Externalities

No Network Externalities
- food/drink - (should be) independent of how many others are eating the type of food.
- Music - your enjoyment should be independent of others enjoyment.

Negative Network Externalities
- clothing
- costumes
Optimization Problem Involving Network Externalities

- Let $V$ be a set of users.
Optimization Problem Involving Network Externalities

- Let $V$ be a set of users.
- Let $v_i(S)$ be the value that user $i$ has for a good if $S \subseteq V$ already own the good — e.g. $v_i(S) = \omega_i + f_i(\sum_{j \in S} w_{ij})$ where $\omega_i$ is inherent value, and $f_i$ might be a concave function, and $w_{ij}$ is now important $j \in S$ is to $i$ (e.g., a network).
Optimization Problem Involving Network Externalities

- Let \( V \) be a set of users.
- Let \( v_i(S) \) be the value that user \( i \) has for a good if \( S \subseteq V \) already own the good — e.g. \( v_i(S) = \omega_i + f_i(\sum_{j \in S} w_{ij}) \) where \( \omega_i \) is inherent value, and \( f_i \) might be a concave function, and \( w_{ij} \) is now important \( j \in S \) is to \( i \) (e.g., a network).
- Given price \( p \) for good, user \( i \) buys good if \( v_i(S) \geq p \).
Optimization Problem Involving Network Externalities

- Let $V$ be a set of users.
- Let $v_i(S)$ be the value that user $i$ has for a good if $S \subseteq V$ already own the good — e.g. $v_i(S) = \omega_i + f_i\left(\sum_{j \in S} w_{ij}\right)$ where $\omega_i$ is inherent value, and $f_i$ might be a concave function, and $w_{ij}$ is now important $j \in S$ is to $i$ (e.g., a network).
- Given price $p$ for good, user $i$ buys good if $v_i(S) \geq p$.
- We choose initial price $p$ and initial set of users $A \subseteq V$ who get the good for free.
Let $V$ be a set of users.

Let $v_i(S)$ be the value that user $i$ has for a good if $S \subseteq V$ already own the good — e.g. $v_i(S) = \omega_i + f_i(\sum_{j \in S} w_{ij})$ where $\omega_i$ is inherent value, and $f_i$ might be a concave function, and $w_{ij}$ is now important $j \in S$ is to $i$ (e.g., a network).

Given price $p$ for good, user $i$ buys good if $v_i(S) \geq p$.

We choose initial price $p$ and initial set of users $A \subseteq V$ who get the good for free.

Define $S_1 = \{i \notin A : v_i(A) \geq p\}$ initial set of buyers.
Optimization Problem Involving Network Externalities

- Let $V$ be a set of users.
- Let $v_i(S)$ be the value that user $i$ has for a good if $S \subseteq V$ already own the good — e.g. $v_i(S) = \omega_i + f_i(\sum_{j \in S} w_{ij})$ where $\omega_i$ is inherent value, and $f_i$ might be a concave function, and $w_{ij}$ is now important $j \in S$ is to $i$ (e.g., a network).
- Given price $p$ for good, user $i$ buys good if $v_i(S) \geq p$.
- We choose initial price $p$ and initial set of users $A \subseteq V$ who get the good for free.
- Define $S_1 = \{i \notin A : v_i(A) \geq p\}$ initial set of buyers.
- $S_2 = \{i \notin A \cup S_1 : v_i(A \cup S_1) \geq p\}$.
Optimization Problem Involving Network Externalities

- Let $V$ be a set of users.
- Let $v_i(S)$ be the value that user $i$ has for a good if $S \subseteq V$ already own the good — e.g. $v_i(S) = \omega_i + f_i(\sum_{j \in S} w_{ij})$ where $\omega_i$ is inherent value, and $f_i$ might be a concave function, and $w_{ij}$ is now important $j \in S$ is to $i$ (e.g., a network).
- Given price $p$ for good, user $i$ buys good if $v_i(S) \geq p$.
- We choose initial price $p$ and initial set of users $A \subseteq V$ who get the good for free.
- Define $S_1 = \{i \notin A : v_i(A) \geq p\}$ initial set of buyers.
- $S_2 = \{i \notin A \cup S_1 : v_i(A \cup S_1) \geq p\}$.
- This starts a cascade. Let $S_k = \{\cup_{j < k} S_j \cup A \mid v_j(\cup_{j < k} S_j \cup A) \geq p\}$,
Optimization Problem Involving Network Externalities

- Let $V$ be a set of users.
- Let $v_i(S)$ be the value that user $i$ has for a good if $S \subseteq V$ already own the good — e.g. $v_i(S) = \omega_i + f_i(\sum_{j \in S} w_{ij})$ where $\omega_i$ is inherent value, and $f_i$ might be a concave function, and $w_{ij}$ is now important $j \in S$ is to $i$ (e.g., a network).
- Given price $p$ for good, user $i$ buys good if $v_i(S) \geq p$.
- We choose initial price $p$ and initial set of users $A \subseteq V$ who get the good for free.
- Define $S_1 = \{i \notin A : v_i(A) \geq p\}$ initial set of buyers.
- $S_2 = \{i \notin A \cup S_1 : v_i(A \cup S_1) \geq p\}$.
- This starts a cascade. Let $S_k = \{\cup_{j < k} S_j \cup A | v_j(\cup_{j < k} S_j \cup A) \geq p\}$,
- and let $S_{k^*}$ be the saturation point, lowest value of $k$ such that $S_k = S_{k+1}$.
Optimization Problem Involving Network Externalities

- Let $V$ be a set of users.
- Let $v_i(S)$ be the value that user $i$ has for a good if $S \subseteq V$ already own the good — e.g. $v_i(S) = \omega_i + f_i(\sum_{j \in S} w_{ij})$ where $\omega_i$ is inherent value, and $f_i$ might be a concave function, and $w_{ij}$ is now important $j \in S$ is to $i$ (e.g., a network).
- Given price $p$ for good, user $i$ buys good if $v_i(S) \geq p$.
- We choose initial price $p$ and initial set of users $A \subseteq V$ who get the good for free.
- Define $S_1 = \{i \notin A : v_i(A) \geq p\}$ initial set of buyers.
- $S_2 = \{i \notin A \cup S_1 : v_i(A \cup S_1) \geq p\}$.
- This starts a cascade. Let $S_k = \{\cup_{j<k} S_j \cup A | v_j(\cup_{j<k} S_j \cup A) \geq p\}$,
  and let $S_{k^*}$ be the saturation point, lowest value of $k$ such that $S_k = S_{k+1}$
- Goal: find $A$ and $p$ to maximize $p \times |S_{k^*}|$. 
Image Segmentation

- an image needing to be segmented.
labeled data in the form of some pixels being marked foreground (red) and others being marked background (blue).
the foreground is removed from the background.
Markov random fields and image segmentation

Markov random field

\[
\log p(x) \propto \sum_{v \in V(G)} e_v(x_v) + \sum_{(i,j) \in E(G)} e_{ij}(x_i, x_j) \tag{1.8}
\]

When \( G \) is a 2D grid graph, we have
We can create auxiliary graph that involves two new nodes $s$ and $t$ and connect each of $s$ and $t$ to all of the original nodes.

I.e., $G_a = (V \cup \{s, t\}, E + \bigcup_{v \in V}((s, v) \cup (v, t)))$. 
Markov random fields and image segmentation

Original Graph: \( \log p(x) \propto \sum_{v \in V(G)} e_v(x_v) + \sum_{(i,j) \in E(G)} e_{ij}(x_i, x_j) \)
Markov random fields and image segmentation

Augmented graph-cut graph. The edge weights of graph are derived from \( \{e_v\}_{v \in V} \) and \( \{e_{ij}\}_{(i,j) \in E(G)} \).
Augmented graph-cut graph with indicated cut corresponding to particular vector $\bar{x} \in \{0, 1\}^n$. Each cut $\bar{x}$ has a score corresponding to $p(\bar{x})$. 
Other applications in or related to computer vision

- Image denoising
- Multi-label graph cuts
- Graphical model inference, computing the Viterbi (or the MPE or the MAP) assignment of a set of random variables.
- Clustering of data
Given a set of random variables $\{X_i\}_{i \in V}$ indexed by set $V$, how do we partition them so that we can best block-code them within each block.
Given a set of random variables \( \{X_i\}_{i \in V} \) indexed by set \( V \), how do we partition them so that we can best block-code them within each block.

I.e., how do we form \( S \subseteq V \) such that \( I(X_S; X_{V \setminus S}) \) is as small as possible, where \( I(X_A; X_B) \) is the mutual information between random variables \( X_A \) and \( X_B \).
Given a set of random variables $\{X_i\}_{i \in V}$ indexed by set $V$, how do we partition them so that we can best block-code them within each block.
Given a set of random variables \( \{X_i\}_{i \in V} \) indexed by set \( V \), how do we partition them so that we can best block-code them within each block.

I.e., how do we form \( S \subseteq V \) such that \( I(X_S; X_{V \setminus S}) \) is as small as possible, where \( I(X_A; X_B) \) is the mutual information between random variables \( X_A \) and \( X_B \).
A network of senders/receivers
A network of senders/receivers

Each sender \( X_i \) is trying to communicate simultaneously with each receiver \( Y_i \) (i.e., for all \( i \), \( X_i \) is sending to \( \{Y_i\}_i \))
A network of senders/receivers

Each sender $X_i$ is trying to communicate simultaneously with each receiver $Y_i$ (i.e., for all $i$, $X_i$ is sending to $\{Y_i\}_i$)

The $X_i$ are not necessarily independent.
A network of senders/receivers

Each sender $X_i$ is trying to communicate simultaneously with each receiver $Y_i$ (i.e., for all $i$, $X_i$ is sending to $\{Y_i\}_i$)

The $X_i$ are not necessarily independent.

Communication rates from $i$ to $j$ are $R^{(i\rightarrow j)}$ to send message $W^{(i\rightarrow j)} \in \{1, 2, \ldots, 2^{nR^{(i\rightarrow j)}}\}$.
A network of senders/receivers
Each sender $X_i$ is trying to communicate simultaneously with each receiver $Y_i$ (i.e., for all $i$, $X_i$ is sending to $\{Y_i\}_i$)
The $X_i$ are not necessarily independent.

- Communication rates from $i$ to $j$ are $R^{(i\to j)}$ to send message $W^{(i\to j)} \in \{1, 2, \ldots, 2^{nR^{(i\to j)}}\}$.
- Goal: necessary and sufficient conditions for achievability as we’ve done for other channels.
A network of senders/receivers

Each sender $X_i$ is trying to communicate simultaneously with each receiver $Y_i$ (i.e., for all $i$, $X_i$ is sending to $\{Y_i\}_i$)

The $X_i$ are not necessarily independent.

Communication rates from $i$ to $j$ are $R^{(i\rightarrow j)}$ to send message $W^{(i\rightarrow j)} \in \{1, 2, \ldots, 2^{nR^{(i\rightarrow j)}}\}$.

Goal: necessary and sufficient conditions for achievability as we’ve done for other channels.

I.e., can we find functions $f$ such that any rates must satisfy

$$\forall S \subseteq V, \sum_{i \in S, j \in V \setminus S} R^{(i\rightarrow j)} \leq f(S)$$  \hspace{1cm} (1.9)
Anecdote

From David Brooks, NYT's column, March 28th, 2011 on “Tools for Thinking”. In response to Steven Pinker (Harvard) asking a number of people “What scientific concept would improve everybody’s cognitive toolkit?”

Emergent systems are ones in which many different elements interact. The pattern of interaction then produces a new element that is greater than the sum of the parts, which then exercises a top-down influence on the constituent elements.
Submodular Motivation

- Given a set of objects \( V = \{v_1, \ldots, v_n\} \) and a function \( f : 2^V \rightarrow \mathbb{R} \) that returns a real value for any subset \( S \subseteq V \).

- Suppose we are interested in finding the subset that either maximizes or minimizes the function, e.g., \( \arg \max_{S \subseteq V} f(S) \), possibly subject to some constraints.

- In general, this problem has exponential time complexity.

- Example: \( f \) might correspond to the value (e.g., information gain) of a set of sensor locations in an environment, and we wish to find the best set \( S \subseteq V \) of sensors locations given a fixed upper limit on the number of sensors \( |S| \).

- In many cases (such as above) \( f \) has properties that make its optimization tractable to either exactly or approximately compute.

- One such property is submodularity.
Submodular Definitions

Definition 1.4.1 (submodular)

A function \( f : 2^V \rightarrow \mathbb{R} \) is submodular if for any \( A, B \subseteq V \), we have that:

\[
f(A) + f(B) \geq f(A \cup B) + f(A \cap B)
\]

(1.10)

An alternate and equivalent definition is:

Definition 1.4.2 (diminishing returns)

A function \( f : 2^V \rightarrow \mathbb{R} \) is submodular if for any \( A \subseteq B \subset V \), and \( v \in V \setminus B \), we have that:

\[
f(A \cup \{v\}) - f(A) \geq f(B \cup \{v\}) - f(B)
\]

(1.11)

This means that the incremental “value”, “gain”, or “cost” of \( v \) decreases (diminishes) as the context in which \( v \) is considered grows from \( A \) to \( B \).
Definition 1.4.3 (subadditive)

A function $f : 2^V \rightarrow \mathbb{R}$ is subadditive if for any $A, B \subseteq V$, we have that:

$$f(A) + f(B) \geq f(A \cup B) \quad (1.12)$$

This means that the “whole” is less than the sum of the parts.
Supermodular Definitions

**Definition 1.4.4 (supermodular)**

A function \( f : 2^V \rightarrow \mathbb{R} \) is supermodular if for any \( A, B \subseteq V \), we have that:

\[
f(A) + f(B) \leq f(A \cup B) + f(A \cap B)
\]  

(1.13)

An alternate and equivalent definition is:

**Definition 1.4.5 (increasing returns)**

A function \( f : 2^V \rightarrow \mathbb{R} \) is supermodular if for any \( A \subseteq B \subset V \), and \( v \in V \setminus B \), we have that:

\[
f(A \cup \{v\}) - f(A) \leq f(B \cup \{v\}) - f(B)
\]  

(1.14)

The incremental “value”, “gain”, or “cost” of \( v \) increases as the context in which \( v \) is considered grows from \( A \) to \( B \).
Superadditive Definitions

Definition 1.4.6 (superadditive)

A function $f : 2^V \to \mathbb{R}$ is superadditive if for any $A, B \subseteq V$, we have that:

$$f(A) + f(B) \leq f(A \cup B) \quad (1.15)$$

This means that the “whole” is greater than the sum of the parts.
Modular Definitions

Definition 1.4.7 (modular)
A function that is both submodular and supermodular is called **modular**.

If $f$ is a modular function, than for any $A, B \subseteq V$, we have

$$f(A) + f(B) = f(A \cap B) + f(A \cup B) \quad (1.16)$$

Modular functions have no interaction, and have value based only on singleton values.

Proposition 1.4.8

*If $f$ is modular, it may be written as*

$$f(A) = f(\emptyset) + \sum_{a \in A} \left( f(\{a\}) - f(\emptyset) \right) \quad (1.17)$$
Proof.

We inductively construct the value for $A = \{a_1, a_2, \ldots, a_k\}$.

\[ f(a_1) + f(a_2) = f(a_1, a_2) + f(\emptyset) \]  \hspace{1cm} (1.18)

implies

\[ f(a_1, a_2) = f(a_1) - f(\emptyset) + f(a_2) - f(\emptyset) + f(\emptyset) \]  \hspace{1cm} (1.19)

then

\[ f(a_1, a_2) + f(a_3) = f(a_1, a_2, a_3) + f(\emptyset) \]  \hspace{1cm} (1.20)

implies

\[ f(a_1, a_2, a_3) = f(a_1, a_2) - f(\emptyset) + f(a_3) - f(\emptyset) + f(\emptyset) \]  \hspace{1cm} (1.21)

\[ = f(\emptyset) + \sum_{i=1}^{3} f(a_i) - f(\emptyset) \]  \hspace{1cm} (1.22)
Given a function $f : 2^V \rightarrow \mathbb{R}$, we can find a complement function $\bar{f} : 2^V \rightarrow \mathbb{R}$ as $\bar{f}(A) = f(V \setminus A)$ for any $A$.

**Proposition 1.4.9**

$\bar{f}$ is submodular if $f$ is submodular.

**Proof.**

\[
\bar{f}(A) + \bar{f}(B) \geq \bar{f}(A \cup B) + \bar{f}(A \cap B) \tag{1.23}
\]

follows from

\[
f(V \setminus A) + f(V \setminus B) \geq f(V \setminus (A \cup B)) + f(V \setminus (A \cap B)) \tag{1.24}
\]

which is true because $V \setminus (A \cup B) = (V \setminus A) \cap (V \setminus B)$ and $V \setminus (A \cap B) = (V \setminus A) \cup (V \setminus B)$.

Submodular functions have a long history in economics, game theory, combinatorial optimization, electrical networks, and operations research.

They are gaining importance in machine learning as well (one of our main motivations for offering this course).

Arbitrary set functions are hopelessly difficult to optimize, while the minimum of submodular functions can be found in polynomial time, and the maximum can be constant-factor approximated in low-order polynomial time.

Submodular functions share properties in common with both convex and concave functions.
Attractions of Convex Functions

Why do we like Convex Functions? (Quoting Lovász 1983):

1. Convex functions occur in many mathematical models in economy, engineering, and other sciences. Convexity is a very natural property of various functions and domains occurring in such models; quite often the only non-trivial property which can be stated in general.

2. Convexity is preserved under many natural operations and transformations, and thereby the effective range of results can be extended, elegant proof techniques can be developed as well as unforeseen applications of certain results can be given.

3. Convex functions and domains exhibit sufficient structure so that a mathematically beautiful and practically useful theory can be developed.

4. There are theoretically and practically (reasonably) efficient methods to find the minimum of a convex function.
Attractions of Submodular Functions

In this course, we wish to demonstrate that submodular functions also possess attractions of these four sorts as well.
Consider an urn containing colored balls. Given a set $S$ of balls, $f(S)$ counts the number of distinct colors.

Submodularity: Incremental Value of Object Diminishes in a Larger Context (diminishing returns).

Thus, $f$ is submodular.
Ex. Submodular: Consumer Costs of Living

- Costs to a consumer are submodular. For example:
Costs to a consumer are submodular. For example:

$$f(\text{fries}) + f(\text{drink}) \geq f(\text{fries}) + f(\text{burger})$$
Ex. Submodular: Consumer Costs of Living

- Costs to a consumer are submodular. For example:
  \[ f(\text{fries}) + f(\text{drink}) \geq f(\text{fries}) + f(\text{hamburger}) \]

- When seen as diminishing returns:
Ex. **Submodular: Consumer Costs of Living**

- Costs to a consumer are submodular. For example:
  \[ f(\text{food}) + f(\text{drink}) \geq f(\text{food}) + f(\text{food}) \]

- When seen as diminishing returns:
  \[ f(\text{food}) - f(\text{drink}) \geq f(\text{food}) - f(\text{drink}) \]
Let $V$ be a set of indices, and each $v \in V$ indexes a given sub-area of some region. Let $\text{area}(v)$ be the area corresponding to item $v$.

Let $f(S) = \bigcup_{s \in S} \text{area}(s)$ be the union of the areas indexed by elements in $A$.

Then $f(S)$ is submodular.
Area of the union of areas indexed by $A$

Union of areas of elements of $A$ is given by:

$$f(A) = f\left(\{a_1, a_2, a_3, a_4\}\right)$$
Area of the union of areas indexed by $A$

Area of $A$ along with $v$:

$$f(A \cup \{v\}) = f(\{a_1, a_2, a_3, a_4\} \cup \{v\})$$
Gain (value) of $v$ in context of $A$:

$$f(A \cup \{v\}) - f(A) = f(\{v\})$$

We get full value $f(\{v\})$ in this case since the area of $v$ has no overlap with that of $A$. 
Area of the union of areas indexed by $A$

Area of $A$ once again.

\[ f(A) = f(\{a_1, a_2, a_3, a_4\}) \]
Area of the union of areas indexed by $A$

Union of areas of elements of $B \supset A$, where $v$ is not included:

$$f(B) \text{ where } v \notin B \text{ and where } A \subseteq B$$
Area of the union of areas indexed by $A$

Area of $B$ now also including $v$:

$$f(B \cup \{v\})$$
Area of the union of areas indexed by $A$

Incremental value of $v$ in the context of $B$.

$$f(B \cup \{v\}) - f(B) < f(\{v\}) = f(A \cup \{v\}) - f(A)$$

So benefit of $v$ in the context of $A$ is greater than the benefit of $v$ in the context of $B \supseteq A$. 
• Entropy is submodular. Let $V$ be the index set of a set of random variables, then the function

$$f(A) = H(X_A) = -\sum_{x_A} p(x_A) \log p(x_A)$$  \hspace{1cm} (1.25)$$

is submodular.

• Proof: conditioning reduces entropy. With $A \subseteq B$ and $v \notin B$,

$$H(X_v|X_B) = H(X_{B+v}) - H(X_B) \leq H(X_{A+v}) - H(X_A) = H(X_v|X_A)$$  \hspace{1cm} (1.26)$$
Alternate Proof: Mutual Information is non-negative.

Mutual information between two sets of random variables $X_A$ and $X_B$ is given by

$$I(X_A; X_B) = \sum_{x_{A \cup B}} p(x_{A \cup B}) \log \frac{p(x_{A \cup B})p(x_{A \cap B})}{p(x_A)p(x_B)} \geq 0 \quad (1.27)$$

then

$$I(X_A; X_B) = H(X_A) + H(X_B) - H(X_{A \cup B}) - H(X_{A \cap B}) \geq 0 \quad (1.28)$$

so entropy satisfies

$$H(X_A) + H(X_B) \geq H(X_{A \cup B}) + H(X_{A \cap B}) \quad (1.29)$$
Example Submodular: Mutual Information

Theory

Also, symmetric mutual information is submodular,

\[ f(A) = I(X_A; X_{V\setminus A}) = H(X_A) + H(X_{V\setminus A}) - H(X_V) \]

(1.30)

Note that \( f(A) = H(X_A) \) and \( \bar{f}(A) = H(X_{V\setminus A}) \), and adding submodular functions preserves submodularity (which we will see quite soon).
Undirected Graphs

Let $G = (V, E)$ be a graph with vertices $V = V(G)$ and edges $E \subseteq V \times V = E(G)$.

If $G$ is undirected, define

$E(X, Y) = \{\{x, y\} \in E(G) : x \in X \setminus Y, y \in Y \setminus X\}$ as the edges between $X$ and $Y$.

Nodes define cuts, and define $\delta(X) = E(X, V \setminus X)$.

$G = (V, E)$

$S = \{a, b, c\}$

$\delta_G(S) = \{\{u, v\} \in E : u \in S, v \in V \setminus S\} = \{\{a, d\}, \{b, d\}, \{b, e\}, \{c, e\}, \{c, f\}\}$
Directed Graphs

If $G$ is directed, define
$$E^+(X, Y) = \{(x, y) \in E(G) : x \in X \setminus Y, y \in Y \setminus X\}$$
as the edges from $X$ to $Y$.

Nodes define cuts, and define edges leaving $X$ as
$$\delta^+(X) = E^+(X, V \setminus X)$$
and edges entering $X$ as
$$\delta^-(X) = E^+(V \setminus X, X).$$

$$\delta_G(S) = \{(v, u) \in E : u \in S, v \in V \setminus S\}.$$
Neighbors function in undirected graphs

Given a set $X \subseteq V$, the neighbors of $X$ is defined as

$$\Gamma(X) = \{ v \in V(G) \setminus X : E(X, \{v\}) \neq \emptyset \}.$$
Lemma 1.6.1

For a digraph $G = (V, E)$ and any $X, Y \subseteq V$: we have

\[ |\delta^+(X)| + |\delta^+(Y)| = |\delta^+(X \cap Y)| + |\delta^+(X \cup Y)| + |E^+(X, Y)| + |E^+(Y, X)| \]

(1.31)

\[ |\delta^-(X)| + |\delta^-(Y)| = |\delta^-(X \cap Y)| + |\delta^-(X \cup Y)| + |E^-(X, Y)| + |E^-(Y, X)| \]

(1.32)
Proof.

We can prove this using a simple geometric counting argument ($\delta^-(X)$ is similar)

\[ |\delta^+(X)| \]
\[ V \setminus Y \]
\[ |\delta^+(X \cap Y)| \]
\[ V \setminus Y \]
\[ |E^+(X, Y)| \]
\[ V \setminus Y \]
\[ |\delta^+(Y)| \]
\[ V \setminus Y \]
\[ |\delta^+(X \cup Y)| \]
\[ V \setminus Y \]
\[ |E^+(Y, X)| \]
\[ V \setminus Y \]
Directed Cut functions

Lemma 1.6.2

For a digraph $G = (V, E)$ and any $X, Y \subseteq V$: both functions $|\delta^+(X)|$ and $|\delta^-(X)|$ are submodular.

Proof.

$|E^+(X, Y)| \geq 0$ and $|E^-(X, Y)| \geq 0$. 

□
Lemma 1.6.3

For an undirected graph $G = (V, E)$ and any $X, Y \subseteq V$: we have

\[
|\delta(X)| + |\delta(Y)| = |\delta(X \cap Y)| + |\delta(X \cup Y)| + 2|E(X, Y)| \tag{1.33}
\]

\[
|\Gamma(X)| + |\Gamma(Y)| \geq |\Gamma(X \cap Y)| + |\Gamma(X \cup Y)| \tag{1.34}
\]

Proof.

Eq. (1.33) directly follows from Eq. (1.31) by replacing each edge $\{u, v\}$ with two oppositely directed edges $(u, v)$ and $(v, u)$ and using the same counting argument.

Eq. (1.34) follows since

\[
|\Gamma(X)| + |\Gamma(Y)| = |\Gamma(X\cup Y)| + |\Gamma(X)\cap \Gamma(Y)| + |\Gamma(X)\cap Y| + |\Gamma(Y)\cap X| \geq |\Gamma(X \cap Y)| + |\Gamma(X \cup Y)|
\]
Graphically, we can count and see that

\[
\Gamma(X) = (a) + (c) + (f) + (g) + (d) \tag{1.35}
\]

\[
\Gamma(Y) = (b) + (c) + (e) + (h) + (d) \tag{1.36}
\]

\[
\Gamma(X \cup Y) = (a) + (b) + (c) + (d) \tag{1.37}
\]

\[
\Gamma(X \cap Y) = (c) + (g) + (h) \tag{1.38}
\]

So

\[
|\Gamma(X)| + |\Gamma(Y)| = (a) + (b) + 2(c) + 2(d) + (e) + (f) + (g) + (h)
\]

\[
\geq (a) + (b) + 2(c) + (d) + (g) + (h) = |\Gamma(X \cup Y)| + |\Gamma(X \cap Y)| \tag{1.39}
\]
Therefore, the undirected cut function $\delta(A)$ and the neighbor function $\Gamma(A)$ of a graph $G$ are both submodular.
Other graph functions that are submodular/supermodular

These come from Narayanan’s book 1997. Let $G$ be an undirected graph.

- Let $V(X)$ be the vertices adjacent to some edge in $X \subseteq E(G)$, then $|V(X)|$ (the vertex function) is submodular.
- Let $E(S)$ be the edges with both vertices in $S \subseteq V(G)$. Then $|E(S)|$ (the interior edge function) is supermodular.
- Let $I(S)$ be the edges with at least one vertex in $S \subseteq V(G)$. Then $|I(S)|$ (the incidence function) is submodular.
- Recall $\delta(S)$, is the set of edges with exactly one vertex in $S \subseteq V(G)$ is submodular. Thus, we see that $I(S) = E(S) + \delta(S)$ and $|I(S)| = |E(S)| + |\delta(S)|$. So we can get a submodular function by summing a submodular and a supermodular function.
These come from Narayanan’s book 1997. Let $G$ be an undirected graph.

- Let $V(X)$ be the vertices adjacent to some edge in $X \subseteq E(G)$, then $|V(X)|$ (the vertex function) is submodular.
- Let $E(S)$ be the edges with both vertices in $S \subseteq V(G)$. Then $|E(S)|$ (the interior edge function) is supermodular.
- Let $I(S)$ be the edges with at least one vertex in $S \subseteq V(G)$. Then $|I(S)|$ (the incidence function) is submodular.
- Recall $\delta(S)$, is the set of edges with exactly one vertex in $S \subseteq V(G)$ is submodular. Thus, we see that $I(S) = E(S) + \delta(S)$ and $|I(S)| = |E(S)| + |\delta(S)|$. So we can get a submodular function by summing a submodular and a supermodular function. If you had to guess, is this always the case?
Other graph functions that are submodular/supermodular

These come from Narayanan’s book 1997. Let $G$ be an undirected graph.

- Let $V(X)$ be the vertices adjacent to some edge in $X \subseteq E(G)$, then $|V(X)|$ (the vertex function) is submodular.
- Let $E(S)$ be the edges with both vertices in $S \subseteq V(G)$. Then $|E(S)|$ (the interior edge function) is supermodular.
- Let $I(S)$ be the edges with at least one vertex in $S \subseteq V(G)$. Then $|I(S)|$ (the incidence function) is submodular.
- Recall $\delta(S)$, is the set of edges with exactly one vertex in $S \subseteq V(G)$ is submodular. Thus, we see that $I(S) = E(S) + \delta(S)$ and $|I(S)| = |E(S)| + |\delta(S)|$. So we can get a submodular function by summing a submodular and a supermodular function.
Other graph functions that are submodular/supermodular

These come from Narayanan's book 1997. Let $G$ be an undirected graph.

- Let $V(X)$ be the vertices adjacent to some edge in $X \subseteq E(G)$, then $|V(X)|$ (the vertex function) is submodular.
- Let $E(S)$ be the edges with both vertices in $S \subseteq V(G)$. Then $|E(S)|$ (the interior edge function) is supermodular.
- Let $I(S)$ be the edges with at least one vertex in $S \subseteq V(G)$. Then $|I(S)|$ (the incidence function) is submodular.
- Recall $\delta(S)$, is the set of edges with exactly one vertex in $S \subseteq V(G)$ is submodular. Thus, we see that $I(S) = E(S) + \delta(S)$ and $|I(S)| = |E(S)| + |\delta(S)|$. So we can get a submodular function by summing a submodular and a supermodular function. If you had to guess, is this always the case?
- Consider $f(A) = |\delta^+[A]| - |\delta^+[V \setminus A]|$. Guess, submodular, supermodular, modular, or neither? Exercise: determine which one and prove it.
Other graph functions that are submodular/supermodular

- Given a graph, for each $X \subseteq E(G)$, let $c(X)$ denote the number of connected components of the subgraph $(V(G), X)$. Then $c(X)$ is supermodular.

- $\bar{c}(X) = c(E \setminus X)$ is the number of connected components in $G$ when we remove $X$, is also supermodular. Maximizing $\bar{c}(X)$ might seem as a goal for a network attacker (choose a small number of edges to sever the graph into as many components as possible).
Matrix Rank functions

Let $V$ be an index set of a set of vectors in $\mathbb{R}^M$ for some $M$. So, for a given set $\{v, v_1, v_2, \ldots, v_k\}$ we might or might not have the possibility of

$$x_v = \sum_{i=1}^{k} \alpha_i x_i \quad (1.40)$$

and if not, then $x_v$ is linearly independent of $x_{v_1}, \ldots, x_{v_k}$.

Let $r(S)$ for $S \subseteq V$ be the rank of the set of vectors $S$. Then $r(\cdot)$ is a submodular function, and in fact is called a matrix matroid rank function.
Let $S$ be a set of subspaces of a linear space and let, for each $X \subseteq S$, $f(X)$ denote the dimensionality of the linear subspace spanned by the subspaces in $X$. We can think of $S$ as a set of sets of vectors from the previous example, and for each $s \in S$, let $X_s$ being an index of vectors. Then, defining

$$f(X) = r(\bigcup_{s \in S} X_s) \quad (1.41)$$

is submodular, and is known to be a polymatroid rank function. In general, polymatroid rank function are submodular, normalized $f(\emptyset) = 0$, and non-decreasing ($f(A) \leq f(B)$ whenever $A \subseteq B$).
Spanning trees

Let $E$ be a set of edges of some graph $G = (V, E)$, and let $r(S)$ for $S \subseteq E$ be the maximum size (in terms of number of edges) spanning forest in the vertex-induced graph induced by edges adjacent to $S$. Then $r(S)$ is submodular, and is another matroid rank function.
Supply Side Economies of scale: Cost of manufacturing a set of items

- Let $V$ be a set of possible items that a company might possibly wish to manufacture, and let $f(S)$ for $S \subseteq V$ be the cost to that company to manufacture subset $S$.

- Ex: $V$ might be colors of paint in a paint manufacturer: green, red, blue, yellow, white, etc.

- Producing green when you are already producing yellow and blue is probably cheaper than if you were only producing some other colors.

\[
f(\text{green, blue, yellow}) - f(\text{blue, yellow}) \leq f(\text{green, blue}) - f(\text{blue}) \quad (1.7)
\]

- So a submodular function would be a good model.
A model of Influence in Social Networks

- Given a graph $G = (V, E)$, each $v \in V$ corresponds to a person, to each $v$ we have an activation function $f_v : 2^V \to [0, 1]$ dependent only on its neighbors. i.e., $f_v(A) = f_v(A \cap \Gamma(v))$.

- Goal - Viral Marketing: find a small subset $S \subseteq V$ of individuals to directly influence, and thus indirectly influence the greatest number of possible other individuals (via the social network $G$).

- We define a function $f : 2^V \to \mathbb{Z}^+$ that models the ultimate influence of an initial set $S$ of nodes based on the following iterative process: At each step, a given set of nodes $S$ are activated, and we activate new nodes $v \in V \setminus S$ if $f_v(S) \geq U[0, 1]$ (where $U[0, 1]$ is a uniform random number between 0 and 1).

- It can be shown that for many $f_v$ (including simple linear functions, and where $f_v$ is submodular itself) that $f$ is submodular.
Let $V$ be a group of individuals. How valuable to you is a given friend $v \in V$?

It depends on how many friends you have.

Given a group of friends $S \subseteq V$, can you valuate them with a function $f(S)$ and how?

Let $f(S)$ be the value of the set of friends $S$. Is submodular or supermodular a good model?
Let $V$ be a set of information containing elements ($V$ might say be either words, sentences, documents, web pages, or blogs, each $v \in V$ is one element, so $v$ might be a word, a sentence, a document, etc.). The total amount of information in $V$ is measure by a function $f(V)$, and any given subset $S \subseteq V$ measures the amount of information in $S$, given by $f(S)$.

How informative is any given item $v$ in different sized contexts? Any such real-world information function would exhibit diminishing returns, i.e., the value of $v$ decreases when it is considered in a larger context.

So a submodular function would likely be a good model.
Submodular Polyhedra

- Submodular functions have associated polyhedra with nice properties: when a set of constraints in a linear program is a submodular polyhedron, a simple greedy algorithm can find the optimal solution even though the polyhedron is formed via an exponential number of constraints.

\[
P_f = \{ x \in \mathbb{R}^E : x \geq 0, x(S) \leq f(S), \forall S \subseteq E \} \tag{1.42}
\]

- The linear programming problem is to, given \( c \in \mathbb{R}^E \), compute:

\[
\tilde{f}(c) = \max c^T x \text{ such that } x \in P_f
\]  

\( \tag{1.43} \)

- This can be solved using the greedy algorithm! Moreover, \( \tilde{f}(c) \) computed using greedy is convex if and only of \( f \) is submodular (we will go into this in some detail this quarter).
Definition 1.8.1 (submodular)

A function $f: 2^V \to \mathbb{R}$ is submodular if for any $A, B \subseteq V$, we have that:

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B) \quad (1.10)$$

An alternate and equivalent definition is:

Definition 1.8.2 (diminishing returns)

A function $f: 2^V \to \mathbb{R}$ is submodular if for any $A \subseteq B \subset V$, and $v \in V \setminus B$, we have that:

$$f(A \cup \{v\}) - f(A) \geq f(B \cup \{v\}) - f(B) \quad (1.11)$$

This means that the incremental "value", "gain", or "cost" of $v$ decreases (diminishes) as the context in which $v$ is considered grows from $A$ to $B$. 

Prof. Jeff Bilmes
EE596A/Fall 2012/Submodularity – Lecture 1 - September 26th, 2012
An alternate and equivalent definition is:

**Definition 1.8.1 (group diminishing returns)**

A function \( f : 2^V \rightarrow \mathbb{R} \) is submodular if for any \( A \subseteq B \subseteq V \), and \( C \subseteq V \setminus B \), we have that:

\[
f(A \cup C) - f(A) \geq f(B \cup C) - f(B)
\]  

This means that the incremental “value” or “gain” of set \( C \) decreases as the context in which \( v \) is considered grows from \( A \) to \( B \) (diminishing returns)
Proposition 1.8.2

Group diminishing returns \textit{implies} diminishing returns

Proof.

Obvious, set $C = \{v\}$.

Proposition 1.8.3

Diminishing returns \textit{implies} group diminishing returns
Proof.

Let $C = \{c_1, c_2, \ldots, c_k\}$. Then diminishing returns implies the series of inequalities

$$f(A \cup C) - f(A)$$

$$= f(A \cup C) - \sum_{i=1}^{k-1} \left( f(A \cup \{c_1, \ldots, c_i\}) - f(A \cup \{c_1, \ldots, c_{i-1}\}) \right) - f(A) \quad (1.46)$$

$$= \sum_{i=1}^{k} f(A \cup \{c_1 \ldots c_i\}) - f(A \cup \{c_1 \ldots c_{i-1}\}) \quad (1.47)$$

$$\leq \sum_{i=1}^{k} f(B \cup \{c_1 \ldots c_i\}) - f(B \cup \{c_1 \ldots c_{i-1}\}) \quad (1.48)$$

$$= f(B \cup C) - \sum_{i=1}^{k-1} \left( f(B \cup \{c_1, \ldots, c_i\}) - f(B \cup \{c_1, \ldots, c_{i-1}\}) \right) - f(B) \quad (1.49)$$

$$f(B \cup C) - f(B) \quad (1.50)$$
Proposition 1.8.4

The two aforementioned definitions of submodularity submodular and diminishing returns are identical.
Submodular Definitions are equivalent (cont. II)

Proof.

Assume submodular. Assume \( A \subset B \) as otherwise trivial.
Let \( B \setminus A = \{v_1, v_2, \ldots, v_k\} \) and define \( A^i = A \cup \{v_1 \ldots v_i\} \), so \( A^0 = A \).
Then by submodular,

\[
f(A^i + v) + f(A^i + v_{i+1}) \geq f(A^i + v + v_{i+1}) + f(A^i)
\]  
(1.51)

or

\[
f(A^i + v) - f(A^i) \geq f(A^i + v_{i+1} + v) - f(A^i + v_{i+1})
\]  
(1.52)

we apply this inductively, and use

\[
f(A^{i+1} + v) - f(A^{i+1}) = f(A^i + v_{i+1} + v) - f(A^i + v_{i+1})
\]  
(1.53)

and that \( A^{k-1} + v_k = B \).
...cont.

Assume *group diminishing returns*. Assume $A \neq B$ otherwise trivial. Define $A' = A \cap B$, $C = A \setminus B$, and $B' = B$. Then

$$f(A' + C) - f(A') \geq f(B' + C) - f(B')$$  \hspace{1cm} (1.54)

giving

$$f(A' + C) + f(B') \geq f(B' + C) + f(A')$$  \hspace{1cm} (1.55)

or

$$f(A \cap B + A \setminus B) + f(B) \geq f(B + A \setminus B) + f(A \cap B)$$  \hspace{1cm} (1.56)