Submodular Function Optimization

http://bicmr.pku.edu.cn/~wenzw/bigdata2018.html

Acknowledgement: this slides is based on Prof. Andreas Krause's, Prof. Jeff Bilmes, Prof. Francis Bach and Prof. Shaddin Dughmi lecture notes

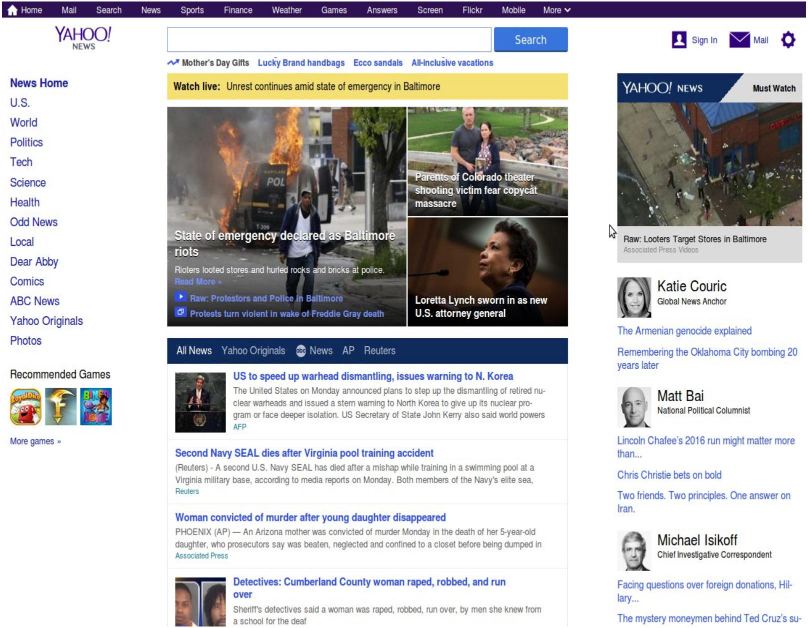
Outline

1. What is submodularity?
   - Examples in recommendation sets
   - Definition
2. Submodular maximization
3. Submodular minimization
4. Applications of submodular maximization

Outline

1. What is submodularity?
   - Examples in recommendation sets
   - Definition
2. Submodular maximization
3. Submodular minimization
4. Applications of submodular maximization

Case study: News article recommendation

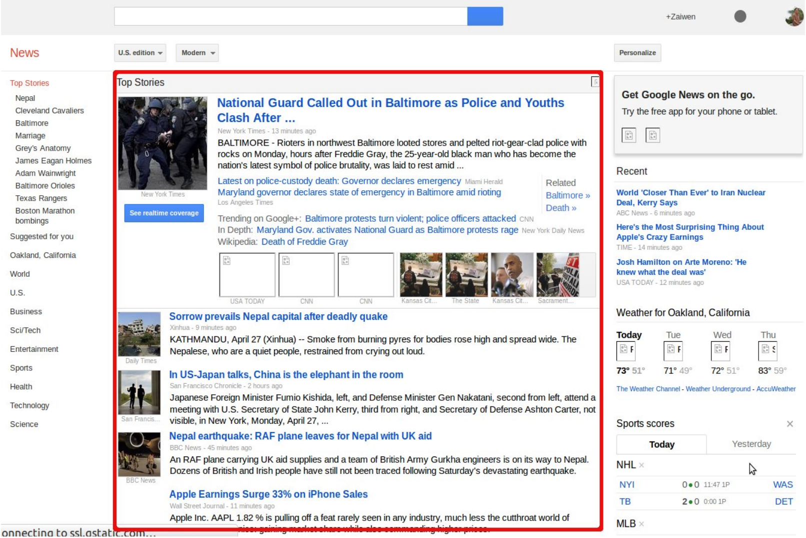


Case study: News article recommendation

Interactive recommendation

- Number of recommendations k to choose from large
  - Similar articles → similar click-through rates!
- Performance depends on query / context
  - Similar users → similar click-through rates!
- Need to compile sets of k recommendations. (instead of only one)
  - Similar sets→ similar click-through rates!

News recommendation

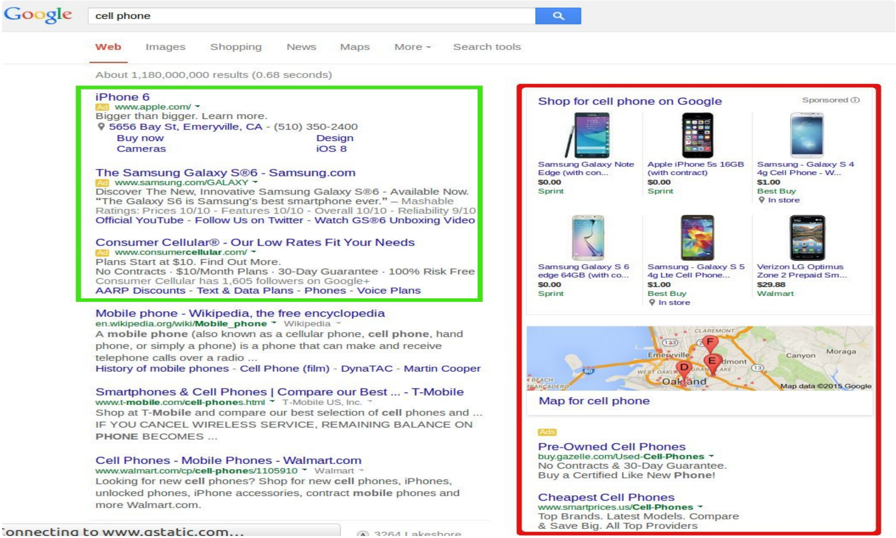


Which set of articles satisfies most users?

News recommendation

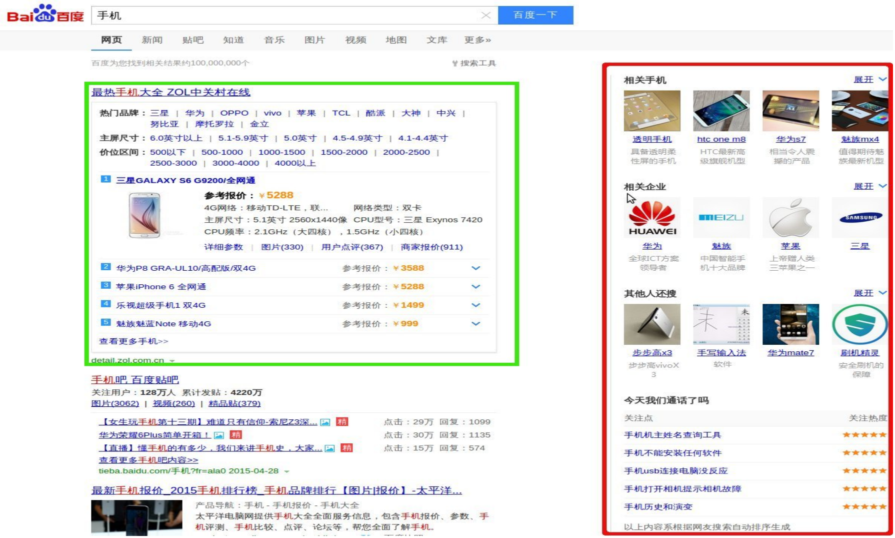
Which set of articles satisfies most users?

Sponsored search



Which set of ads should be displayed to maximize revenue?

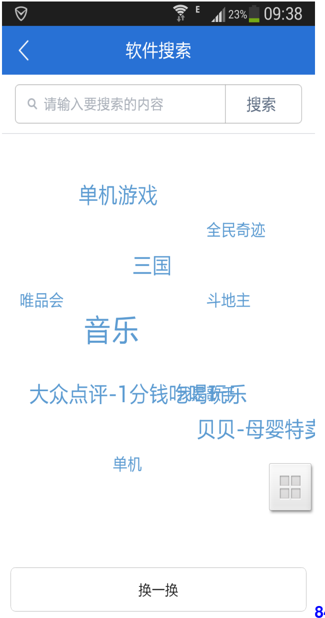
Sponsored search



Which set of ads should be displayed to maximize revenue?

Relevance vs. Diversity

- Users may have different interests / queries may be ambiguous
  - E.g., "jaguar", "squash",· · ·
- Want to choose a set that is relevant to as many users as possible
  - Users may choose from the set the article they’re most interested in
- Want to optimize both relevance and diversity



Simple abstract model

- Suppose we're given a set W of users and a collection V of articles/ads
- Each article i is relevant to a set of users S_i
  - For now suppose this is known!
- For each set of articles define

$$F(A) = |\cup_{i \in A} S_i|$$

- Want to select k articles to maximize "users covered"

$$\max_{|A|<k} F(A)$$

- Number of sets A grows exponential in k!
- Finding optimal A is NP-hard

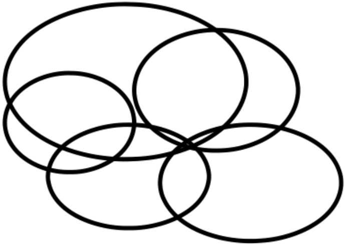
Maximum coverage

- **Given:** Collection V of sets, utility function F(A)

Want: $A^* \subseteq V$ such that

$$\mathcal{A}^* = \text{argmax}_{|\mathcal{A}| \leq k} F(\mathcal{A})$$

NP-hard!



Greedy algorithm:

Start with $A_0 = \{\}$
For i=1 to k
$s^* = \text{argmax}_s F(A_{i-1} \cup \{s\})$
$A_i = A_{i-1} \cup \{s^*\}$

How well does this simple heuristic do?

Approximation guarantee

Theorem

Under some natural conditions,greedy algorithm produces a solution A, where $F(A) \geq (1 - 1/e)$ * optimal-value $(\sim 63\%)$.

[Nemhauser,Fisher,Wolsey'78]

- This result holds for utility functions F with 2 properties:
  - F is (nonnegative) monotone:
    if $A \subseteq B$ then $0 \leq F(A) \leq F(B)$
  - F is submodular:
    "diminishing returns"

Outline

1. What is submodularity?
   - Examples in recommendation sets
   - Definition
2. Submodular maximization
3. Submodular minimization
4. Applications of submodular maximization

Set Functions

- Ground set: subsets of some finite set
- Given a set X, the set $V := 2^X = \{A \mid A \subseteq X\}$
- A set function takes as input a set, and outputs a real number
  - Inputs are subsets of some ground set X
  - $F : 2^X \to \mathbb{R}$
- It is common in the literature to use either X or V as the ground set.
- We will follow this inconsistency in the literature and will inconsistently use either X or V as our ground set (hopefully not in the same equation, if so, please point this out).

Set Functions

- If F is a modular function, then for any $A, B \subseteq V$, we have

$$F(A) + F(B) = F(A \cap B) + F(A \cup B)$$

- If F is a modular function, it may be written as

$$F(A) = F(\emptyset) + \sum_{a \in A} (F(\{a\}) - F(\emptyset))$$

- modular set functions
    - Associate a weight w_i with each $i \in X$, and set $F(S) = \sum_{i \in S} w_i$
    - Discrete analogue of linear functions
- Other possibly useful properties a set function may have:
    - Monotone: if $A \subseteq B \subseteq X$, then $F(A) \leq F(B)$
    - Nonnegative: $F(A) \geq 0$ for all $S \subseteq X$
    - Normalized: $F(\emptyset) = 0$.

Submodular Functions

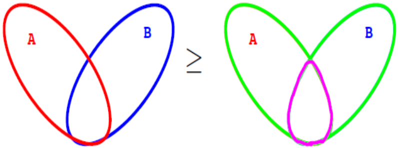
Definition 1

A set function $F : V \to \mathbb{R}$ is submodular if and only if

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

for all $A, B \subseteq V$.

- "Uncrossing" two sets reduces their total function value



Definition

A set function $F : V \to \mathbb{R}$ is supmodular if and only if $-F$ is submodular.

Submodular Functions

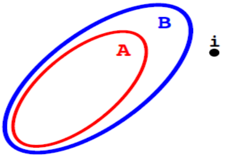
Definition 2 (diminishing returns)

A set function $F : V \to \mathbb{R}$ is submodular if and only if

$$\underbrace{F(B \cup \{s\}) - F(B)}_{\text{Gain of adding a set s to a large solution}} \leq \underbrace{F(A \cup \{s\}) - F(A)}_{\text{Gain of adding a set s to a small solution}}$$

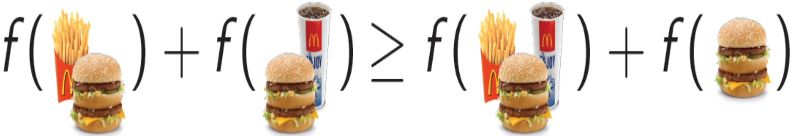
for all $A \subseteq B \subseteq V$ and $s \notin B$.

- The marginal value of an additional element exhibits “diminishing marginal returns”
- This means that the incremental “value”, “gain”, or “cost” of s decreases (diminishes) as the context in which s is considered grows from A to B.

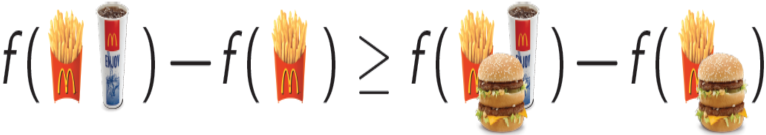


Submodular: Consumer Costs of Living

- Consumer costs are very often submodular. For example:



- When seen as diminishing returns:



Submodular Functions

Definition 3 (group diminishing returns)

A set function $F : V \to \mathbb{R}$ is submodular if and only if

$$F(B \cup C) - F(B) \leq F(A \cup C) - F(A)$$

for all $A \subseteq B \subseteq V$ and $C \subseteq V \setminus B$.

- This means that the incremental “value”, “gain”, or “cost” of set C decreases (diminishes) as the context in which C is considered grows from A to B.

Equivalence of Definitions

Definition 2 $\Longrightarrow$ Definition 3
Let $C = \{c_1, \ldots, c_k\}$. The Definition 2 implies

$$\begin{aligned}
& F(A \cup C) - F(A) \\
= \; & F(A \cup C) - \sum_{i=1}^{k-1} (F(A \cup \{c_1, \ldots, c_i\}) - F(A \cup \{c_1, \ldots, c_i\})) - F(A) \\
= \; & \sum_{i=1}^{k} (F(A \cup \{c_1, \ldots, c_i\}) - F(A \cup \{c_1, \ldots, c_{i-1}\})) \\
\geq \; & \sum_{i=1}^{k} (F(B \cup \{c_1, \ldots, c_i\}) - F(B \cup \{c_1, \ldots, c_{i-1}\})) \\
= \; & F(B \cup C) - F(B)
\end{aligned}$$

Equivalence of Definitions

Definition 1 $\Longrightarrow$ Definition 2
To prove (2), let $A' = A \cup \{i\}$ and $B' = B$ and apply (1)

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

Definition 2 $\Longrightarrow$ Definition 1
Assume $A \neq B$. Define $A' = A \cap B$, $C = A \backslash B$ and $B' = B$. Then

$$\begin{aligned} & F(A' \cup C) - F(A') \geq F(B' \cup C) - F(B') \\ \Longleftrightarrow\quad & F((A \cap B) \cup (A \backslash B)) + F(B) \geq F(B \cup (A \backslash B)) + F(A') \\ \Longleftrightarrow\quad & F(A) + F(B) \geq F(A \cup B) + F(A \cap B) \end{aligned}$$

Submodularity

- 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
- 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.

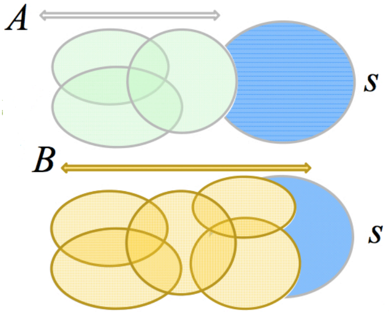
Example: Set cover

- F is submodular: $A \subseteq B$

$$\underbrace{F(A \cup \{s\}) - F(A)}_{\text{Gain of adding a set s to a small solution}} \geq \underbrace{F(B \cup \{s\}) - F(B)}_{\text{Gain of adding a set s to a large solution}}$$

- Natural example:
  - Set $S_1, S_2, \cdots, S_n$
  - F(A)=size of union of S_i (e.g.,,number of satisfied users)

$$F(A) = |\cup_{i \in A} S_i|$$



Closedness properties

- $F_1, \cdots, F_m$ submodular functions on V and $\lambda_1, \cdots, \lambda_m \geq 0$
- Then: $F(A) = \sum_i \lambda_i F_i(A)$ is submodular!

- Submodularity closed under nonnegative linear combinations
- Extremely useful fact:
  - $F_\theta(A)$ submodular $\Rightarrow \sum_\theta P(\theta) F_\theta(A)$ submodular!
  - Multi-objective optimization:
    $F_1, \cdots, F_m$ submodular, $\lambda_i > 0 \Rightarrow \sum_i \lambda_i F_i(A)$ submodular

Probabilistic set cover

- Document coverage function:
  $\text{cover}_d(c)$=probability document d covers concept c, e.g., how strongly d covers c
  It can model how relevant is concept c for user u
- Set coverage function:

$$\text{cover}_A(c) = 1 - \Pi_{d \in A}(1 - \text{cover}_d(c))$$

  Probability that at least one document in A covers c
- Objective:

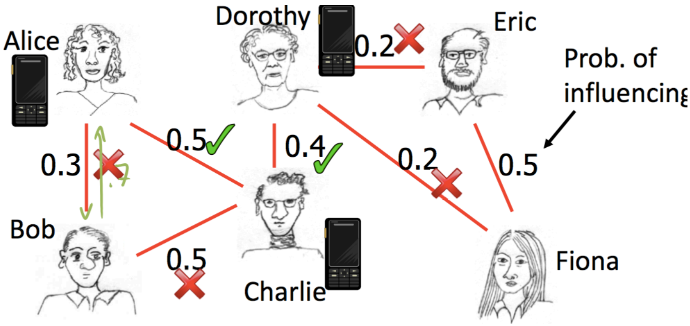
$$\max_{|A| \leq k} \quad F(A) = \sum_c w_c \text{cover}_A(c)$$

  w_c is the concept weights
- The objective function is submodular

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 $f : 2^V \to Z^+$ 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 $v \in V \backslash S$ 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.

Example: Influence in social networks [Kempe, Kleinberg, Tardos KDD'03]

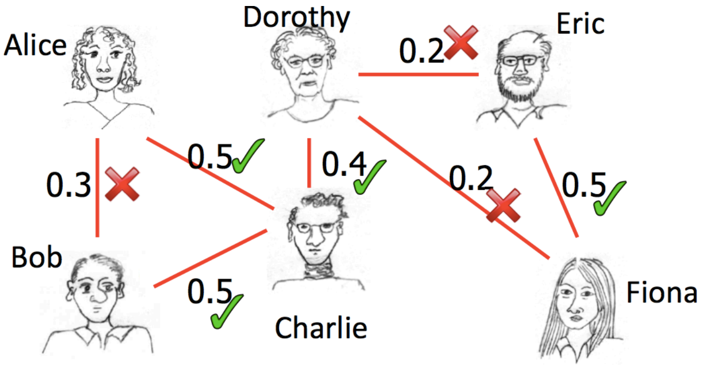


Which nodes are most influential?

V = Alice,Bob,Charlie,Dorothy,Eric,Fiona

F(A) = Expected number of people influenced by set A

Influence in social networks is submodular [Kempe, Kleinberg, Tardos KDD'03]



Key idea: Flip coins c in advance $\rightarrow$ "live" edges

$F_c(A)$ = People influenced under outcome c

$F(A) = \sum_c P(c) F_c(A)$ is submodular as well!

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 $F(S)$ and how?
- Let $F(S)$ be the value of the set of friends S. Is submodular or supermodular a good model?

Information and Summarization

- 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.

Restriction

Restriction

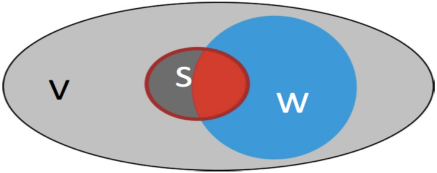
If $F(S)$ is submodular on V and $W \subseteq V$. Then $F'(S) = F(S \cap W)$ is submodular

Proof: Given $A \subseteq B \subseteq V \backslash \{i\}$, prove:

$$F((A \cup \{i\}) \cap W) - F(A \cap W) \geq F((B \cup \{i\}) \cap W) - F(B \cap W).$$

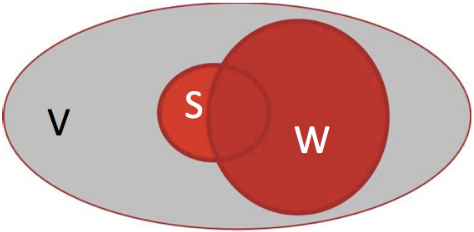
If $i \notin W$, then both differences on each size are zero. Suppose that $i \in W$, then $(A \cup \{i\}) \cap W = (A \cap W) \cup \{i\}$ and $(B \cup \{i\}) \cap W = (B \cap W) \cup \{i\}$. We have $A \cap W \subseteq B \cap W$, the submodularity of F yields

$$F((A \cap W) \cup \{i\}) - F(A \cap W) \geq F((B \cap W) \cup \{i\}) - F(B \cap W).$$



Conditioning

If $F(S)$ is submodular on V and $W \subseteq V$. Then $F'(S) = F(S \cup W)$ is submodular

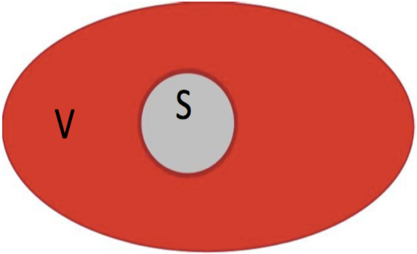


Reflection

If $F(S)$ is submodular on V. Then $F'(S) = F(V \setminus S)$ is submodular

Proof: Since $V \backslash (A \cup B) = (V \backslash A) \cap (V \backslash B)$ and $V \backslash (A \cap B) = (V \backslash A) \cup (V \backslash B)$, then

$$F(V \backslash A) + F(V \backslash B) \geq F(V \backslash (A \cup B)) + F(V \backslash (A \cap B)))$$

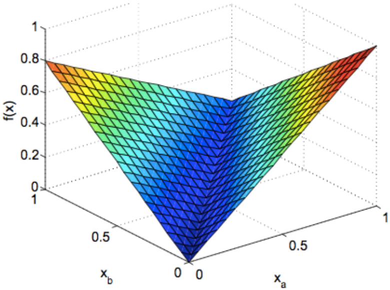


Convex aspects

- Submodularity as discrete analogue of convexity

- Convex extension
- Duality
- Polynomial time minimization!

$$A^* = \arg\min_{A \subseteq V} F(A)$$



- Many applications (computer vision,ML, $\cdots$)

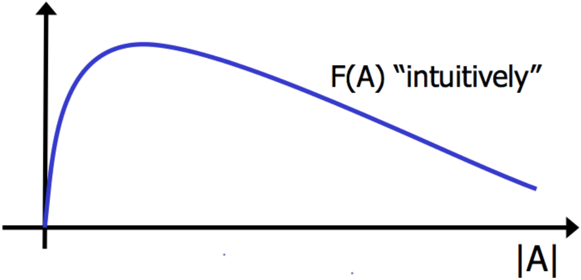
Concave aspects

- Marginal gain $\triangle_F(s|A) = F(\{s\} \cup A) - F(A)$
- Submodular:

$$\forall A \subseteq B, s \notin B: \quad F(A \cup \{s\}) - F(A) \geq F(B \cup \{s\}) - F(B)$$

- Concave:

$$\forall a \leq b, s > 0 \qquad g(a+s) - g(a) \geq g(b+s) - g(b)$$



$$\forall a \leq b, s > 0 \qquad g(a+s) - g(a) \geq g(b+s) - g(b)$$

- Suppose that $a + s \in [a, b]$
- Apply the concavity of $g(x)$ to $[a, a+s, b+s]$:

$$\begin{aligned} g(a+s) &\geq \frac{b-a}{b+s-a} g(a) + \frac{s}{b+s-a} g(b+s) \\ \iff g(a+s) - g(a) &\geq \frac{-s}{b+s-a} g(a) + \frac{s}{b+s-a} g(b+s) \end{aligned}$$

- Apply the concavity of $g(x)$ to $[a+s, b, b+s]$:

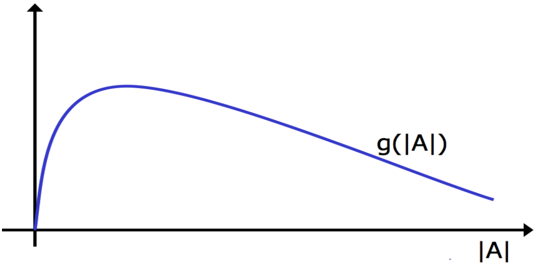
$$\begin{aligned} g(b) &\geq \frac{s}{b+s-a} g(a) + \frac{b-a}{b+s-a} g(b+s) \\ \iff g(b+s) - g(b) &\leq \frac{-s}{b+s-a} g(a) + \frac{s}{b+s-a} g(b+s) \end{aligned}$$

Submodularity and Concavity

Let $m \in \mathbb{R}_+^X$ be a modular function, and g a concave function over $\mathbb{R}$. Define $F(A) = g(m(A))$. Then F(A) is submodular.

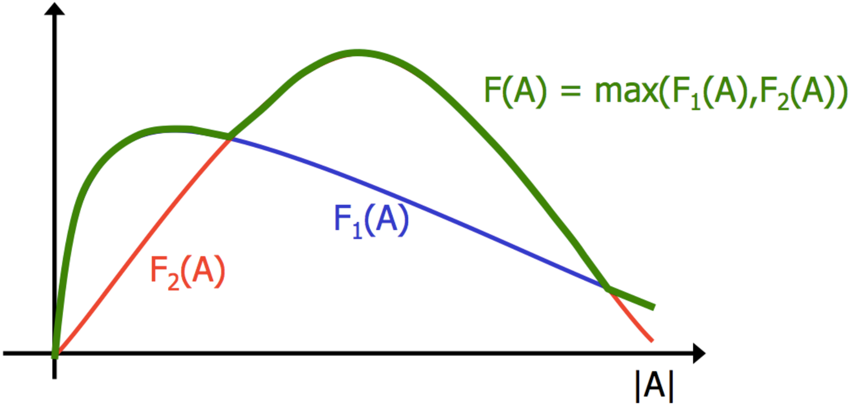
Proof: Given $A \subseteq B \subseteq X \backslash v$, we have $0 \leq a = m(A) \leq b = m(B)$, and $0 \leq s = m(v)$. For g concave, we have $g(a+s) - g(a) \geq g(b+s) - g(b)$, which implies

$$g(m(A) + m(v)) - g(m(A)) \geq g(m(B) + m(v)) - g(m(B))$$



Maximum of submodular functions

Suppose $F_1(A)$ and $F_2(A)$ submodular.
Is $F(A) = \max(F_1(A), F_2(A))$ submodular?



$\max(F_1, F_2)$ not submodular in general!

Minimum of submodular functions

Well,maybe $F(A) = \min(F_1(A), F_2(A))$ instead?

	$F_1(A)$	$F_2(A)$
{}	0	0
{a}	1	0
{b}	0	1
{a,b}	1	1

F({b}) – F({})=0

<

F({a,b}) – F({a})=1

$\max(F_1, F_2)$ not submodular in general!

Max - normalized

Given V, let $c \in \mathbb{R}_+^V$ be a given fixed vector. Then $F : 2^V \to \mathbb{R}_+$, where

$$F(A) = \max_{j \in A} c_j$$

is submodular and normalized (we take $F(\emptyset) = 0$).

Proof: Since

$$\max(\max_{j \in A} c_j, \max_{j \in B} c_j) = \max_{j \in A \cup B} c_j$$

and

$$\min(\max_{j \in A} c_j, \max_{j \in B} c_j) \geq \max_{j \in A \cap B} c_j,$$

we have

$$\max_{j \in A} c_j + \max_{j \in B} c_j \geq \max_{j \in A \cup B} c_j + \max_{j \in A \cap B} c_j$$

Monotone difference of two functions

Let F and G both be submodular functions on subsets of V and let $(F-G)(\cdot)$ be either monotone increasing. Then $h : 2^V \to R$ defined by $h(A) = \min(F(A), G(A))$ is submodular.

- If $h(A)$ agrees with either f or g on both X and Y , the result follows since

$$\begin{matrix} F(X)+F(Y) \\ G(X)+G(Y) \end{matrix} \geq \min(F(X \cup Y), G(X \cup Y)) + \min(F(X \cap Y), G(X \cap Y))$$

- otherwise, w.l.o.g., $h(X) = F(X)$ and $h(Y) = G(Y)$, giving

$$h(X)+h(Y) = F(X)+G(Y) \geq F(X \cup Y) + F(X \cap Y) + G(Y) - F(Y)$$

Assume $F - G$ is monotonic increasing. Hence, $F(X \cup Y) + G(Y) - F(Y) \geq G(X \cup Y)$ giving

$$h(X)+h(Y) \geq G(X \cup Y) + F(X \cap Y) \geq h(X \cup Y) + h(X \cap Y)$$

Min

- Let $F : 2^V \to \mathbb{R}$ be an increasing or decreasing submodular function and let k be a constant. Then the function $h : 2^V \to \mathbb{R}$ defined by

$$h(A) = \min(k; F(A))$$

  is submodular

- In general, the minimum of two submodular functions is not submodular. However, when wishing to maximize two monotone non-decreasing submodular functions, we can define function $h : 2^V \to R$ as

$$h(A) = \frac{1}{2}(\min(k, F) + \min(k, G))$$

  then h is submodular, and $h(A) \geq k$ if and only if both $F(A) \geq k$ and $G(A) \geq k$

Outline

Submodular maximization a Cardinality Constraint

Problem Definition

Given a non-decreasing and normalized submodular function $F : 2^X \to \mathbb{R}^+$ on a finite ground set X with $|X| = n$, and an integer $k \leq n$:

$$\max \quad F(A), \text{ s.t. } |A| \leq k$$

Greedy Algorithm

- $A_0 \leftarrow \emptyset$, set $i = 0$
- While $|A_i| \leq k$
  - Choose $s \in X$ maximizing $F(A_i \cup \{s\})$
  - $A_{i+1} \leftarrow A_i \cup \{s\}$

Greedy maximization is near-optimal

Theorem[Nemhauser, Fisher& Wolsey'78]

For monotonic submodular functions, Greedy algorithm gives constant factor approximation

$$F(A_{\text{greedy}}) \geq \underbrace{(1 - 1/e)}_{\sim 63\%} F(A^*)$$

- Greedy algorithm gives near-optimal solution!
- For many submodular objectives: Guarantees best possible unless P=NP
- Can also handle more complex constraints

Contraction/Conditioning

Let $F : 2^X \to \mathbb{R}$ and $A \subseteq X$. Define $F_A(S) = F(A \cup S) - F(A)$.
Lemma: If F is monotone and submodular, then F_A is monotone, submodular, and normalized for any A.

- Proof: Monotone:
  - Let $S \subseteq T$, then $F_A(S) = F(A \cup S) - F(A) \leq F(A \cup T) - F(A) = F_A(T)$
- Submodular. Let $S, T \subseteq X$:

$$
\begin{aligned}
F_A(S) + F_A(T) &= F(S \cup A) - F(A) + F(T \cup A) - F(A) \\
&\geq F(S \cup T \cup A) - F(A) + F((S \cap T) \cup A) - F(A) \\
&= F_A(S \cup T) + F_A(S \cap T)
\end{aligned}
$$

Lemma

If F is normalized and submodular, and $A \subseteq X$, then there is $j \in A$ such that $F(\{j\}) \geq \frac{1}{|A|} F(A)$

- Proof. If A_1 and A_2 partition A, i.e., $A = A_1 \cup A_2$ and $A_1 \cap A_2 = \emptyset$, then

$$F(A_1) + F(A_2) \geq F(A_1 \cup A_2) + F(A_1 \cap A_2) = F(A)$$

- Applying recursively, we get

$$\sum_{j \in A} F(\{j\}) \geq F(A)$$

- Therefore, $\max_{j \in A} F(\{j\}) \geq \frac{1}{|A|} F(A)$

Greedy maximization is near-optimal

Theorem[Nemhauser, Fisher& Wolsey'78]

For monotonic submodular functions, Greedy algorithm gives constant factor approximation

$$F(A_{\text{greedy}}) \geq (1 - 1/e)F(A^*)$$

- Proof: Let A_i be the working set in the algorithm
- Let A^* be optimal solution.
- We will show that the suboptimality $F(A^*) - F(A)$ shrinks by a factor of $(1 - 1/k)$ each iteration
- After k iterations, it has shrunk to $(1 - 1/k)^k \leq 1/e$ from its original value
- The algorithm choose $s \in X$ maximizing $F(A_i \cup \{s\})$. Hence:

$$F(A_{i+1}) = F(A_i) + F(A_i \cup \{s\}) - F(A_i) = F(A_i) + \max_j F_{A_i}(\{j\})$$

- By our lemmas, there is $j \in A^*$ s.t.

$$\begin{aligned} F_{A_i}(\{j\}) &\geq \frac{1}{|A^*|} F_{A_i}(A^*) \quad \text{(apply lemma to } F_{A_i}) \\ &= \frac{1}{k}(F(A_i \cup A^*) - F(A_i)) \\ &\geq \frac{1}{k}(F(A^*) - F(A_i)) \end{aligned}$$

- Therefore

$$\begin{aligned} F(A^*) - F(A_{i+1}) &= F(A^*) - F(A_i) - \max_j F_{A_i}(\{j\}) \\ &\leq \left(1 - \frac{1}{k}\right)(F(A^*) - F(A_i)) \\ &\leq \left(1 - \frac{1}{k}\right)^k (F(A^*) - F(\emptyset)) \end{aligned}$$

Scaling up the greedy algorithm [Minoux'78]

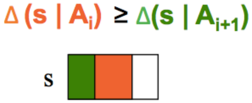
In round i+1,

- have picked $A_i = s_1, \cdots, s_i$
- pick $s_{i+1} = \arg\max_s F(A_i \cup \{s\}) - F(A_i)$

i.e., maximize "marginal benefit" $\triangle(s|A_i)$

$$\triangle(s|A_i) = F(A_i \cup \{s\}) - F(A_i)$$

Key observation: Submodularity implies

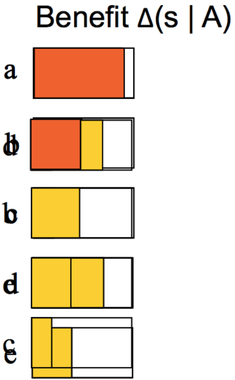


Marginal benefits can never increase!

"Lazy" greedy algorithm [Minoux'78]

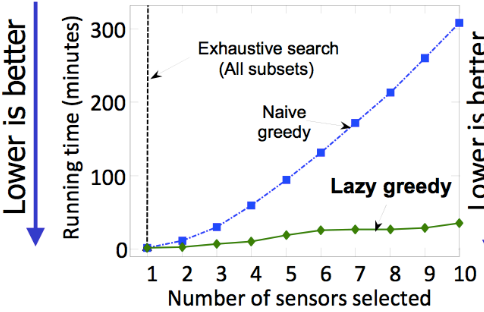
Lazy greedy algorithm:

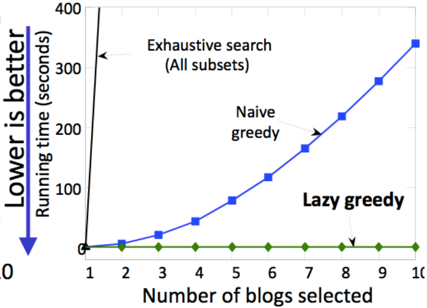
- First iteration as usual
- Keep an ordered list of marginal benefits $\triangle_i$ from previous iteration
- Re-evaluate $\triangle_i$ only for top element
- If $\triangle_i$ stays on top, use it, otherwise re-sort



Note: Very easy to compute online bounds, lazy evaluations, etc. [Leskovec,Krause et al.'07]

Empirical improvements [Leskovec, Krause et al'06]





Sensor placement

Blog selection

30x speedup

700x speedup

Outline

1. What is submodularity?
   - Examples in recommendation sets
   - Definition
2. Submodular maximization
3. Submodular minimization
4. Applications of submodular maximization

Optimizing Submodular Functions

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

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

Representation

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

Problem Definition

Given a submodular function $f : 2^X \to \mathbb{R}$ on a finite ground set X,

$$\begin{aligned} \min \quad & F(S) \\ \text{s.t.} \quad & S \subseteq X \end{aligned}$$

- We denote $n = |X|$
- We assume $F(S)$ is a rational number with at most b bits
- Representation: in order to generalize all our examples, algorithmic results are often posed in the value oracle model. Namely, we only assume we have access to a subroutine evaluating $F(S)$ in constant time.

Goal

An algorithm which runs in time polynomial in n and b.

Some more notations

- $E = \{1, 2, \ldots, n\}$
- $\mathbb{R}^E = \{x = (x_j \in \mathbb{R} : j \in E)\}$
- $\mathbb{R}^E_+ = \{x = (x_j \in \mathbb{R} : j \in E) : x \geq 0\}$
- Any vector $x \in \mathbb{R}^E$ can be treated as a normalized modular function, and vice verse. That is

  $$x(A) = \sum_{a \in A} x_a.$$

  Note that x is said to be normalized since $x(\emptyset) = 0$.
- Given $A \subseteq E$, define the vector $1_A \in \mathbb{R}^E_+$ to be

  $$1_A(j) = \begin{cases} 1 & \text{if } j \in A \\ 0 & \text{if } j \notin A \end{cases}$$

- given modular function $x \in \mathbb{R}^E$, we can write $x(A)$ in a variety of ways, i.e., $x(A) = x \cdot 1_A = \sum_{i \in A} x_i$

Continuous Extensions of a Set Function

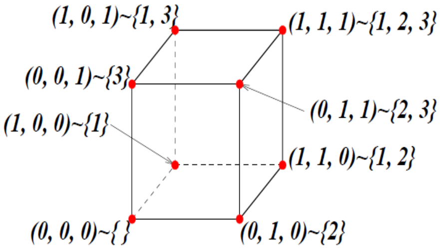
- A set function F on $X = \{1, \ldots, n\}$ can be thought of as a map from the vertices $\{0, 1\}^n$ of the n-dimensional hypercube to the real numbers.

Extension of a Set Function

Given a set function $F : \{0, 1\}^n \to \mathbb{R}$, an extension of F to the hypercube $[0, 1]^n$ is a function $g : [0, 1]^n \to \mathbb{R}$ satisfying $g(x) = F(x)$ for every $x \in \{0, 1\}^n$.

$$\min_{w \in \{0,1\}^n} \quad F(w)$$

with $\forall A \subseteq X$, $F(1_A) = F(A)$



Choquet integral - Lovász extension

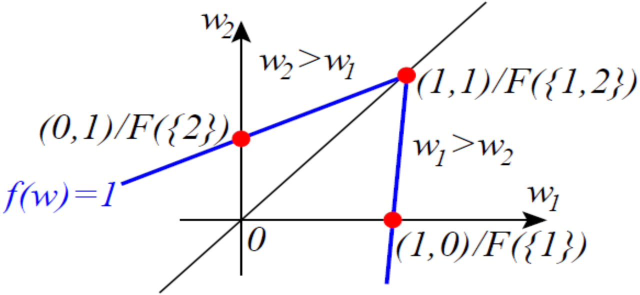
- Subsets may be identified with elements of $\{0,1\}^n$
- Given any set-function F and w such that $w_{j_1} \geq \ldots \geq w_{j_n}$, define

$$
\begin{aligned}
f(w) &= \sum_{k=1}^{n} w_{j_k}[F(\{j_1,\ldots,j_k\}) - F(\{j_1,\ldots,j_{k-1}\})] \\
&= \sum_{k=1}^{n-1} (w_{j_k} - w_{j_{k+1}}) F(\{j_1,\ldots,j_k\}) + w_{j_n} F(\{j_1,\ldots,j_n\})
\end{aligned}
$$

- If $w = 1_A$, $f(w) = F(A) \Longrightarrow$ extension from $\{0,1\}^n$ to $\mathbb{R}^n$

Choquet integral - Lovász extension, example: $p = 2$

- If $w_1 \geq w_2, f(w) = F(\{1\})w_1 + [F(\{1,2\}) - F(\{1\})]w_2$
- If $w_1 \leq w_2, f(w) = F(\{2\})w_2 + [F(\{1,2\}) - F(\{2\})]w_1$



level set $\{w \in \mathbb{R}^2, f(w) = 1\}$ is displayed in blue

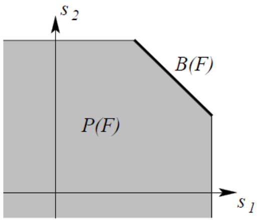
- Compact formulation: $f(w) = [F(\{1,2\}) - F(\{1\}) - F(\{2\})] \min(w_1, w_2) + F(\{1\})w_1 + F(\{2\})w_2$

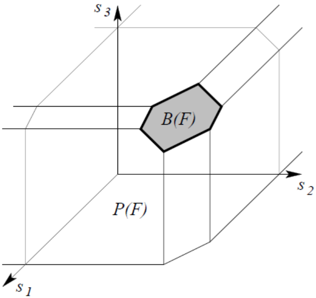
Links with convexity

Theorem (Lovász, 1982)

F is submodular if and only if f is convex

- Proof requires: Submodular and base polyhedra
- Submodular polyhedron: $P(F) = \{s \in \mathbb{R}^n, \forall A \subseteq V, s(A) \leq F(A)\}$
- Base polyhedron: $B(F) = P(F) \cap \{s(V) = F(V)\}$





Submodular and base polyhedra

- $P(F)$ has non-empty interior
- Many facets (up to 2^n), many extreme points (up to $n!$)

Fundamental property (Edmonds, 1970): If F is submodular, maximizing linear functions may be done by a "greedy algorithm"

- Let $w \in \mathbb{R}_+^n$ such that $w_{j_1} \geq \ldots \geq w_{j_n}$
- Let $s_{j_k} = F(\{j_1, \ldots, j_k\}) - F(\{j_1, \ldots, j_{k-1}\})$ for $k \in \{1, \ldots, n\}$
- Then

$$f(w) = \max_{s \in P(F)} w^\top s = \max_{s \in B(F)} w^\top s$$

- Both problems attained at s defined as above.
- proofs: pages 41-44 in `http://bicmr.pku.edu.cn/~wenzw/bigdata/submodular_fbach_mlss2012.pdf`

Links with convexity

Theorem (Lovász, 1982)

F is submodular if and only if f is convex

- If F is submodular, f is the maximum of linear functions. Then f is convex
- If f is convex, let $A, B \subseteq V$
  - $1_{A\cup B} + 1_{A\cap B} = 1_A + 1_B$ has components equal to 0 (on $V \backslash (A \cup B)$), 2 (on $A \cap B$) and 1 (on $A \Delta B = (A \backslash B) \cup (B \backslash A)$)
  - Thus $f(1_{A\cup B} + 1_{A\cap B}) = F(A \cup B) + F(A \cap B)$. Proof by writing out $f(1_{A\cup B} + 1_{A\cap B})$ and the definition of $f(w)$.
  - By homogeneity and convexity, $f(1_A + 1_B) \leq f(1_A) + f(1_B)$, which is equal to $F(A) + F(B)$, and thus F is submodular.

Links with convexity

Theorem (Lovász, 1982)

If F is submodular, then

$$\min_{A \subseteq V} F(A) = \min_{w \in \{0,1\}^n} f(w) = \min_{w \in [0,1]^n} f(w)$$

- Since f is an extension of F,

$$\min_{A \subseteq V} F(A) = \min_{w \in \{0,1\}^n} f(w) \geq \min_{w \in [0,1]^n} f(w)$$

- Any $w \in [0,1]^n$ can be decomposed as $w = \sum_{i=1}^m \lambda_i 1_{B_i}$, where $B_1 \subseteq \ldots \subseteq B_m = V$, where $\lambda \geq 0$ and $\lambda(V) \leq 1$:
  - Since $\min_{A \subseteq V} F(A) \leq 0$ ($F(\emptyset) = 0$),

$$f(w) = \sum_{i=1}^m \lambda_i F(B_i) \geq \sum_{i=1}^m \lambda_i \min_{A \subseteq V} F(A) \geq \min_{A \subseteq V} F(A)$$

  - Thus $\min_{w \in [0,1]^n} f(w) \geq \min_{A \subseteq V} F(A)$.

Links with convexity

- Any $w \in [0,1]^n$, sort $w_{j_1} \geq \ldots \geq w_{j_n}$. Find λ such that

$$\sum_{k=1}^{n} \lambda_{j_k} = w_{j_1}, \sum_{k=2}^{n} \lambda_{j_k} = w_{j_2}, \ldots, \lambda_{j_n} = w_{j_n},$$
$$B_1 = \{j_1\}, B_2 = \{j_1, j_2\}, \ldots, B_n = \{j_1, j_2, \ldots, j_n\}$$

  Then we have $w = \sum_{i=1}^{n} \lambda_i 1_{B_i}$, where $B_1 \subseteq \ldots \subseteq B_n = V$, where $\lambda \geq 0$ and $\lambda(V) = \sum_{i \in V} \lambda_i \leq 1$.

Submodular function minimization

- Let $F : 2^V \to \mathbb{R}$ be a submodular function (such that $F(\emptyset) = 0$)
- **convex duality**:

$$
\begin{aligned}
\min_{A \subseteq V} F(A) &= \min_{w \in [0,1]^n} f(w) \\
&= \min_{w \in [0,1]^n} \max_{s \in B(F)} w^\top s \\
&= \max_{s \in B(F)} \min_{w \in [0,1]^n} w^\top s = \max_{s \in B(F)} s_-(V)
\end{aligned}
$$

Submodular function minimization

Convex optimization

If F is submodular, then

$$\min_{A \subseteq V} F(A) = \min_{w \in \{0,1\}^n} f(w) = \min_{w \in [0,1]^n} f(w)$$

Using projected subgradient descent to minimize f on $[0, 1]^n$

- Iteration: $w_t = \Pi_{[0,1]^n}(w_{t-1} - \frac{C}{\sqrt{t}} s_t)$, where $s_t \in \partial f(w_{t-1})$
- $f(w) = \max_{s \in B(F)} w^\top s$
- Standard convergence results from convex optimization

$$f(w_t) - \min_{w \in [0,1]^n} f(w) \le \frac{C}{\sqrt{t}}$$

Outline

I have 10 minutes. Which blogs should I read to be most up to date?

[Leskovec, Krause, Guestrin, Faloutsos, VanBriesen, Glance'07]

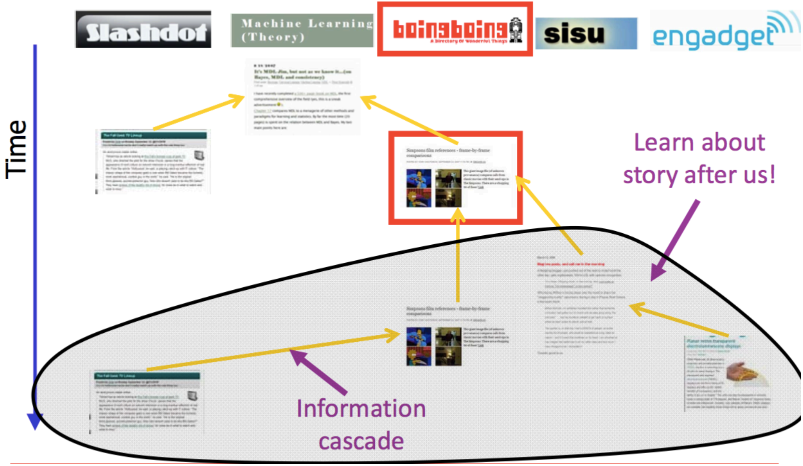
TIME
IN PARTNERSHIP WITH CNN

Thursday, Nov. 20, 2008

How Many Blogs Does the World Need?

By Michael Kinsley

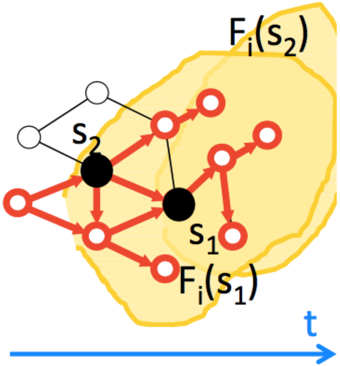
Detecting Cascades in the Blogosphere



Which blogs should we read to learn about big cascades early?

Reward function is submodular

- Consider cascade i:
  - $F_i(s_k)$ = benefit from blog s_k in event i
  - $F_i(A) = \max F_i(s_k), s_k \in A$
    $\Rightarrow F_i$ is submodular



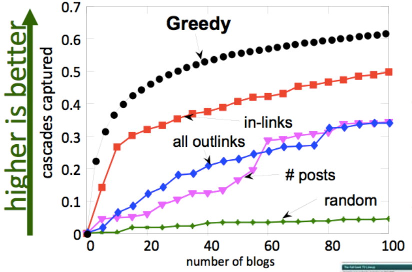
- Overall objective:

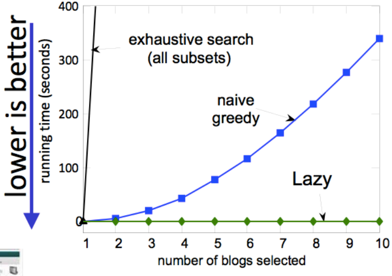
$$F(A) = \frac{1}{m}\sum_{i=1}^{m} F_i(A)$$

$\Rightarrow F$ is submodular

$\Rightarrow$ Can use greedy algorithm to solve $\max_{|A|\leq k} F(A)$!

Performance on Blog selection

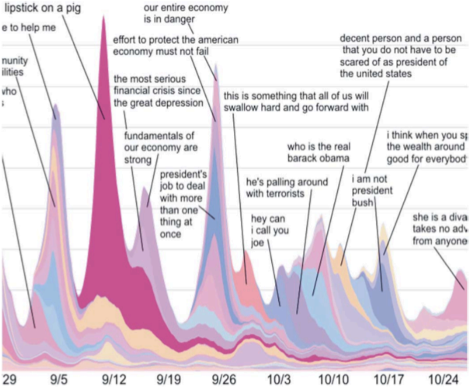




blog selection ~45k blogs

- Submodular formulation outperforms heuristics
- Lazy greedy gives 700x speedup

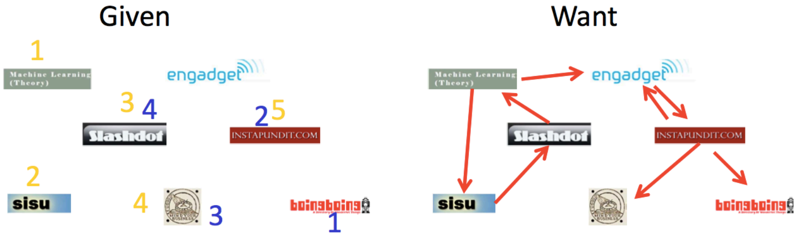
Application: Network inference



How can we learn who influences whom?

Inferring diffusion networks

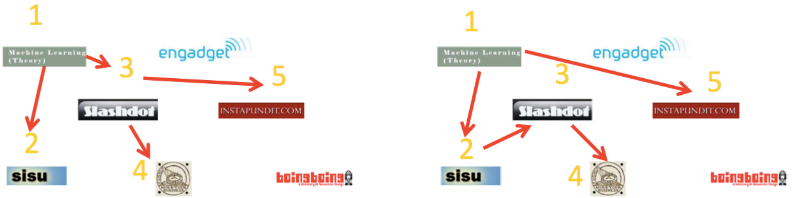
[Gomez Rodriguez, Leskovec, Krause ACM TKDE 2012]



Given traces of influence, wish to infer sparse directed network $G = (V, E)$
$\Rightarrow$ Formulate as optimization problem

$$E^* = \arg\max_{|E| \leq k} F(E)$$

Estimation problem

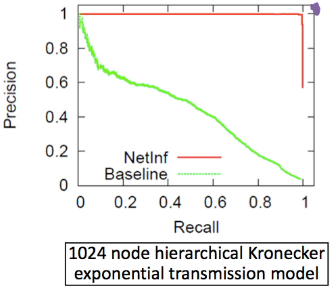


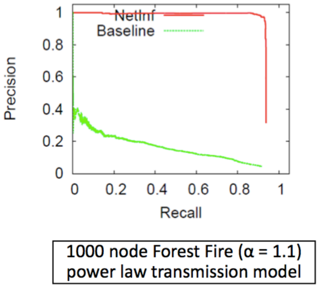
- Many influence trees T consistent with data
- For cascade C_i , model $P(C_i|T)$
- Find sparse graph that maximizes likelihood for all observed cascades

$\Rightarrow$ Log likelihood monotonic submodular in selected edges

$$F(E) = \sum_i \log \max_{\text{tree } T \subseteq E} P(C_i|T)$$

Evaluation: Synthetic networks

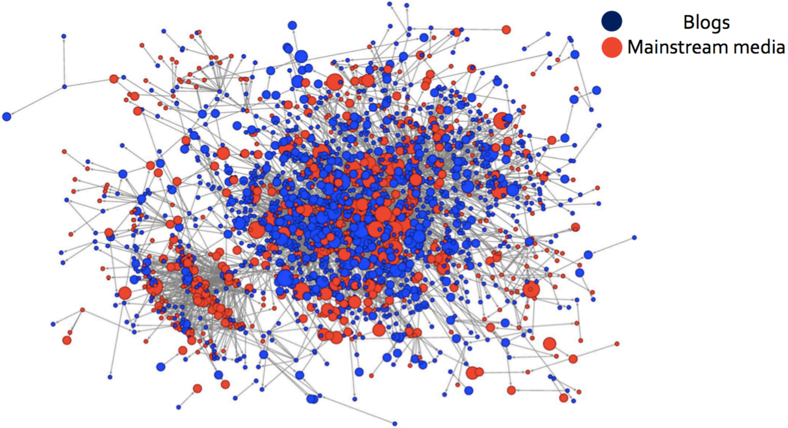




- Performance does not depend on the network structure:
  - Synthetic Networks: Forest Fire, Kronecker, etc.
  - Transmission time distribution: Exponential, Power Law
- Break-even point of > 90%

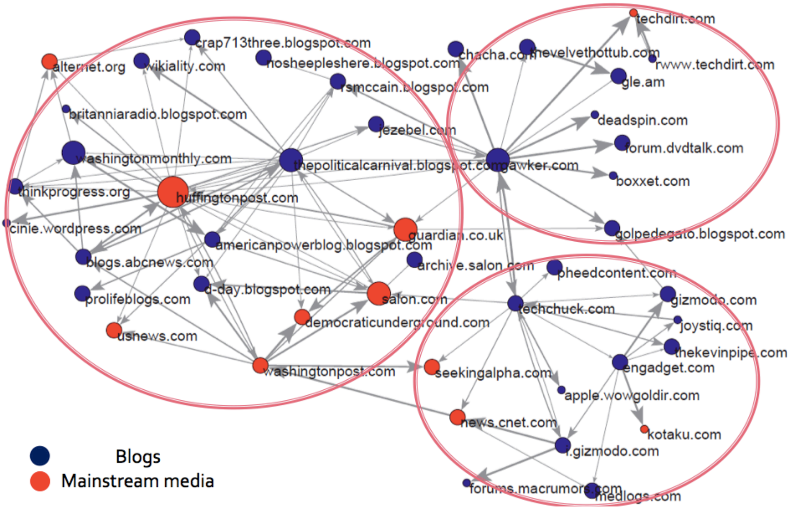
Inferred Diffusion Network

[Gomez Rodriguez, Leskovec, Krause ACM TKDE 2012]



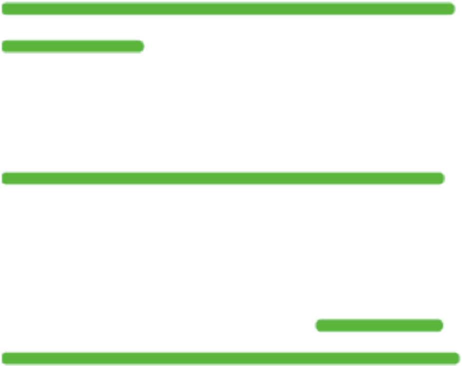
Actual network inferred from 172 million articles from 1 million news sources

Diffusion Network (small part)



Application: Document summarization

[Lin & Bilmes'11]



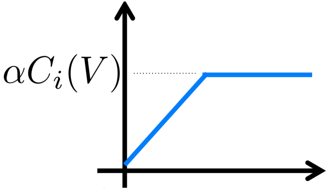
- Which sentences should we select that best summarize a document?

Marginal gain of a sentence

- Many natural notions of “document coverage” are submodular [Lin & Bilmes’11]

Relevance of a summary

$$F(S) = \underset{\substack{\uparrow \\ \text{Relevance}}}{R(S)} + \underset{\substack{\uparrow \\ \text{Diversity}}}{\lambda D(S)}$$



$$R(S) = \sum_i \min\{C_i(S), \alpha C_i(V)\}$$

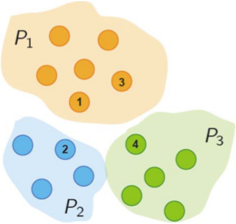
$$C_i(S) = \sum_{j \in S} \omega_{i,j}$$

- $C_i(S)$: How well is sentence i "covered" by S
- $\omega_{i,j}$: Similarity between i and j

Diversity of a summary

$$
\begin{aligned}
D(S) &= \sum_{i=1}^{K} \sqrt{\sum_{j \in P_i \cap S} r_j} \\
r_j &= \frac{1}{N} \sum_i \omega_{i,j}
\end{aligned}
$$

- r_j: Relevance of sentence j to doc.
- $\omega_{i,j}$: Similarity between i and j



Clustering of sentences in document

Can be made query-specific; multi-resolution; etc.

Summary

- Many problems of recommending sets can be cast as submodular maximization
- Greedy algorithm gives best set of size k
- Can use lazy evaluations to speed up
- Approximate submodular maximization possible under a variety of constraints:
  - Matroid
  - Knapsack
  - Multiple matroid and knapsack constraints
  - Path constraints (Submodular orienteering)
  - Connectedness (Submodular Steiner)
  - Robustness (minimax)