Submodular optimization: Maximizing Cascades

Rik Sarkar

Projects

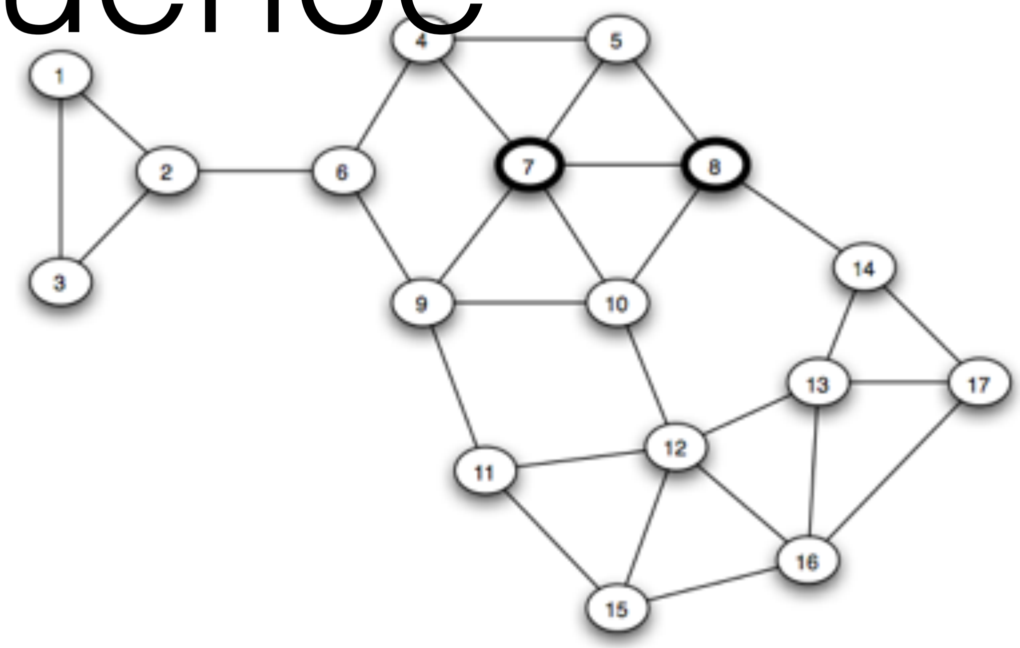
- Thanks for the proposals. We will try to give comments on piazza. Please continue your work till then
- If upload to piazza did not work, please try again
- Guidelines for final submission available soon

Projects: Main points:

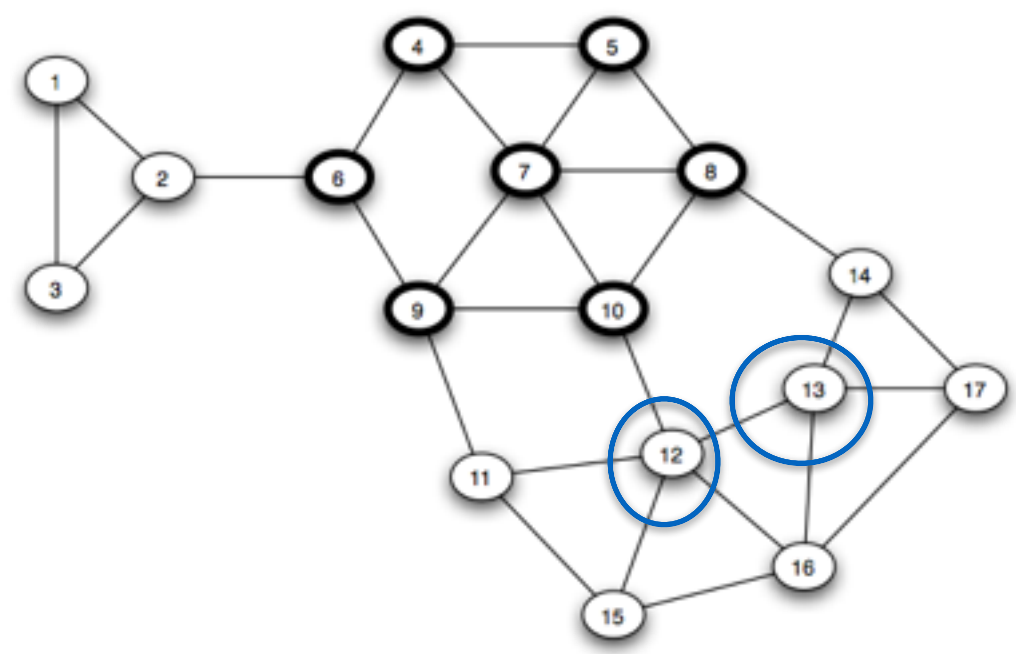
- There is no “right answer”. We don’t know the solutions
- We are happy to discuss with you and help you make the project better
- You will be marked for trying interesting ideas, justifying them and comparing and discussion of results
- Don’t be afraid to try risky/new ideas that may fail

Recap: Contagion, cascades, influence

- Contagion: something that spreads due to influence of neighbors (cascading)
    - Technology, product, innovation, idea, disease…
- The spreading process at a node is often called infection, activation etc…



(a) *Two nodes are the initial adopters*

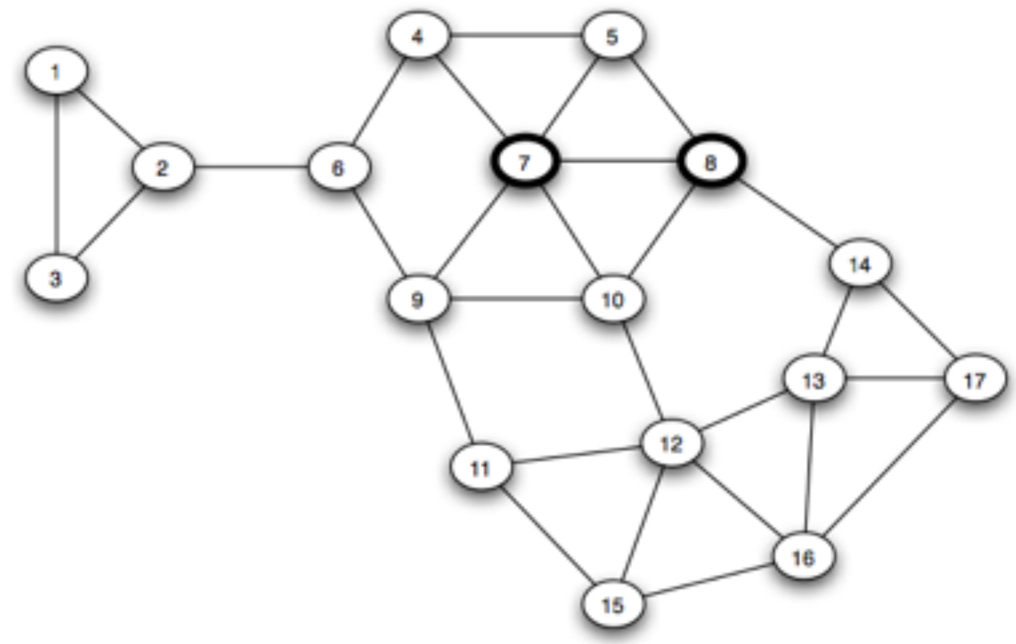


(b) *The process ends after three steps*

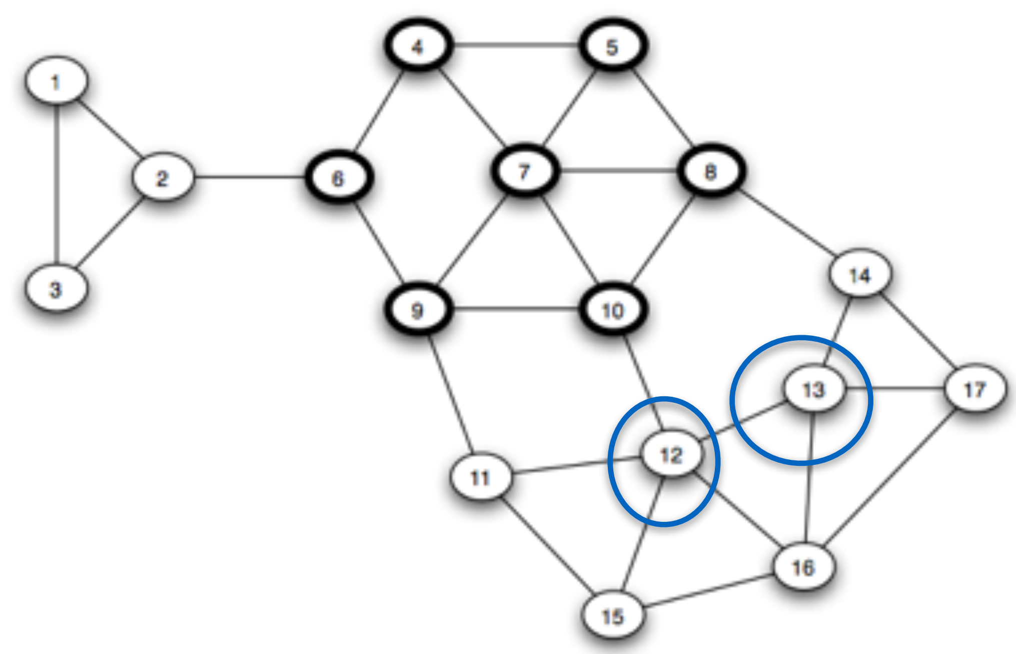
Recap

- Tight knit communities stop the cascade

- Carefully picking some nodes to activate can cause a large cascade



(a) *Two nodes are the initial adopters*



(b) *The process ends after three steps*

α - strong communities

- A set S of nodes forms an α-strong (or α-dense) community if for each node v in S, $d_S(v) \geq \alpha d(v)$
- That is, at least α fraction of neighbors of each node is within the community

Theorem

- A cascade with contagion threshold q cannot penetrate an α-dense community with α > 1 - q
- Therefore, for a cascade with threshold q, and set X of initial adopters of A:
  1. If the rest of the network contains a cluster of density > 1-q, then the cascade from X does not result in a complete cascade
  2. If the cascade is not complete, then the rest of the network must contain a cluster of density > 1-q

Proof

- In Kleinberg & Easley

1. By contradiction: The first node in the cluster that converts, cannot convert.

2. If set S is exactly the set of unconverted nodes at the end, then any v in S must have 1-q fraction edges in S, else v would have converted.

Extensions

- The model extends to the case where each node v has
  - different a_v and b_v , hence different q_v
  - Exercise: What can be a form for the theorem on the previous slide for variable q_v?

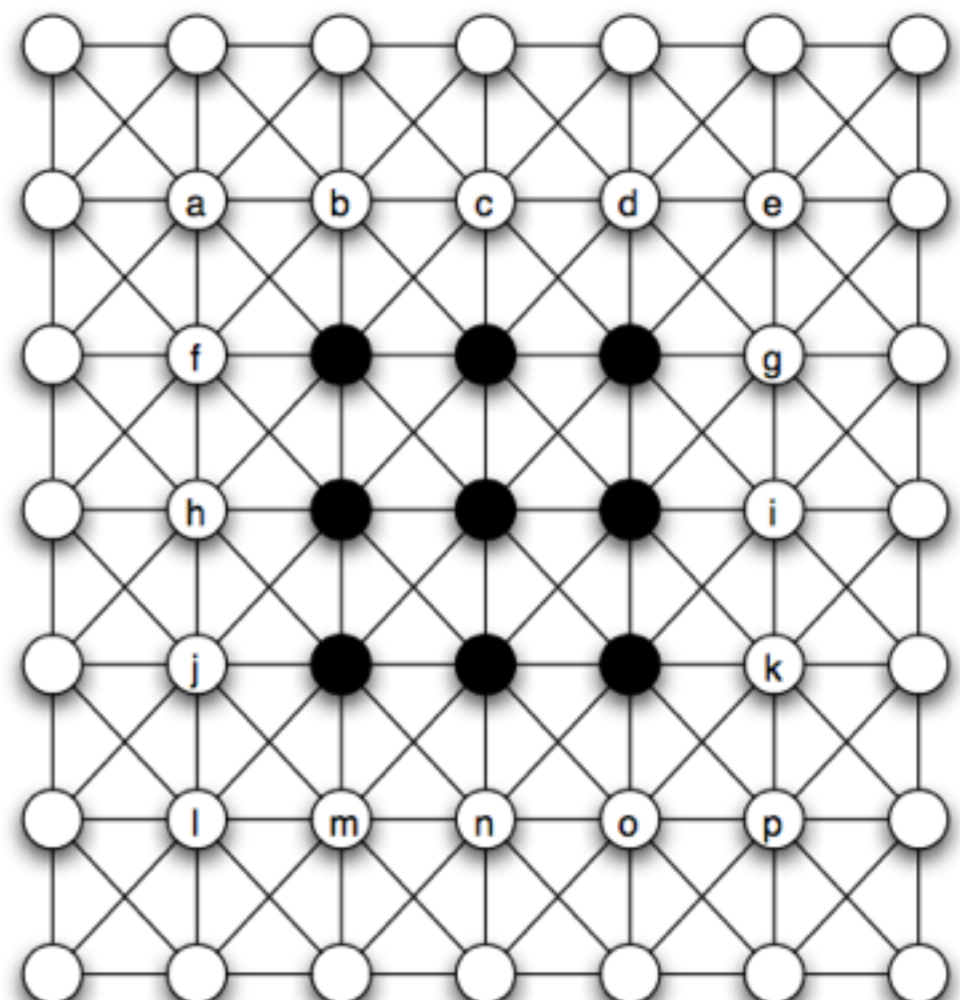
Cascade capacity

- Upto what threshold q can a small set of early adopters cause a full cascade?
- definition: Small: A finite set in an infinite network

Cascade capacities

- 1-D grid:

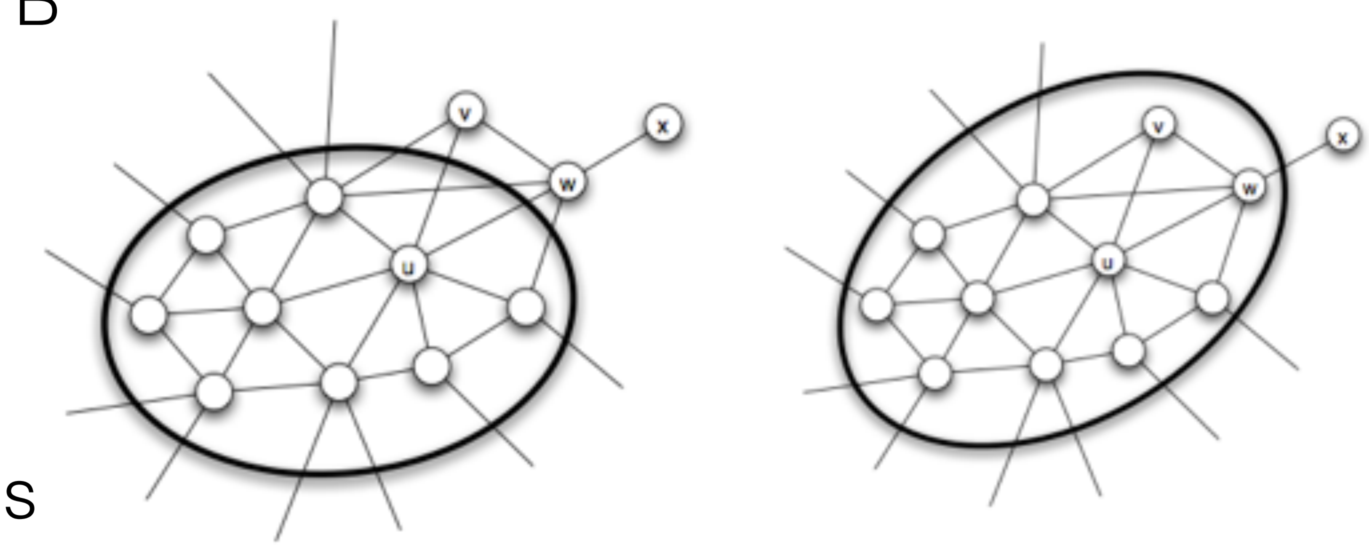
- capacity = 1/2



- 2-D grid with 8 neighbors:
- capacity 3/8

Theorem

- No infinite network has cascade capacity > 1/2
- Show that the interface/boundary shrinks
  - Number of edges at boundary decreases at every step
- Take a node w at the boundary that converts in this step
  - w had x edges to A, y edges to B
  - q > 1/2 implies x > y
- True for all nodes
- Implies boundary edges decreases



(a) *Before v and w adopt A* (b) *After v and w adopt A*

Other models

- Non-monotone: an infected/converted node can become un-converted
- Schelling's model, granovetter's model: People are aware of choices of all other nodes (not just neighbors)

Causing large spread of cascade

- Viral marketing with restricted costs
- Suppose you have a budget of reaching k nodes
- Which k nodes should you convert to get as large a cascade as possible?

Models

- Linear contagion threshold model:
    - The model we have used: node activates to use A if benefit of using $p > q$
- Independent activation model:
    - If node u activates to use A, then u causes neighbor v to activate and use A with probability
        - $p_{u,v}$
        - That is, every edge has an associated probability of spreading influence (like the strength of the tie)

Hardness

- In both the models, finding the exact set of k initial nodes to maximize the influence cascade is NP-Hard
    - Intractable, unlikely that polynomial time algorithms exist unless P = NP

Approximation

- There is a polynomial time algorithm that spreads the cascade to $\left(1 - \frac{1}{e}\right) \cdot OPT$ nodes
    - OPT : The optimum result — in this case, the largest number of nodes reachable with a cascade starting with k nodes

- To prove this, we will use a property called submodularity
- Let us take a detour into understanding submodular functions
- After that, we will complete the proof.

Submodular functions

- Suppose function f(x) represents the total benefit of selecting x
    - And f(S) the benefit of selecting set S
- Function f is submodular if:

$$
S \subseteq T \implies
$$

$$
f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T)
$$

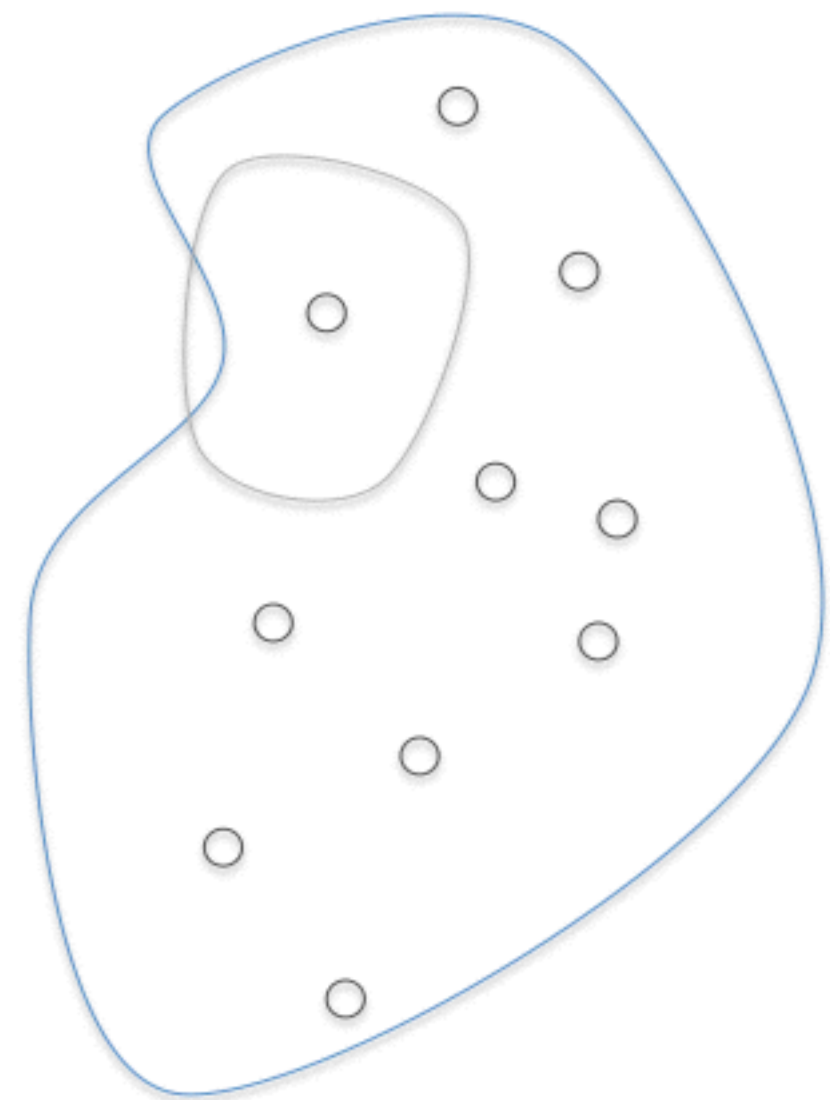
Submodular functions

$$S \subseteq T \implies$$
$$f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T)$$

- Means diminishing returns
- Selecting x gives smaller benefits if many others have been selected

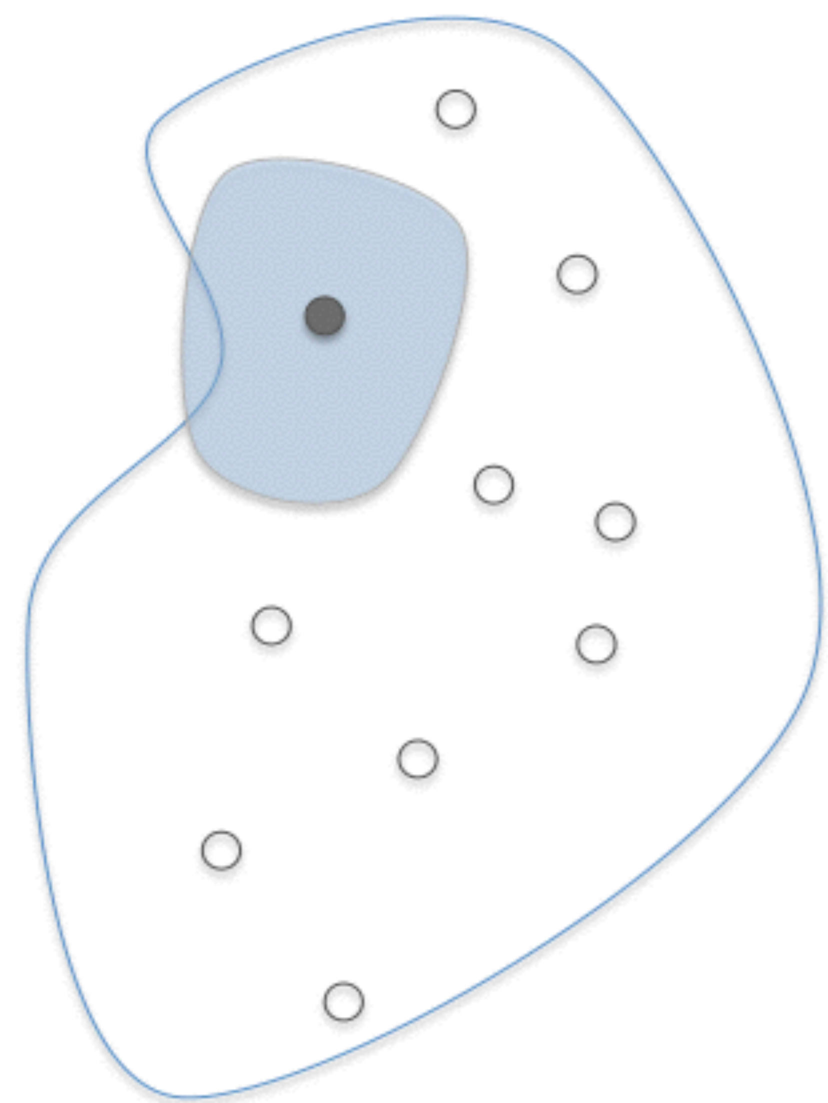
Example: Sensor coverage

- Suppose you are placing sensors to monitor a region (eg. cameras, or chemical sensors etc)
- There are n possible camera locations
- Each sensor can “see” a region
- A region that is in the view of one or more sensors is *covered*
- With a budget of k sensors, we want to cover the largest possible area
    - Function f: Area covered



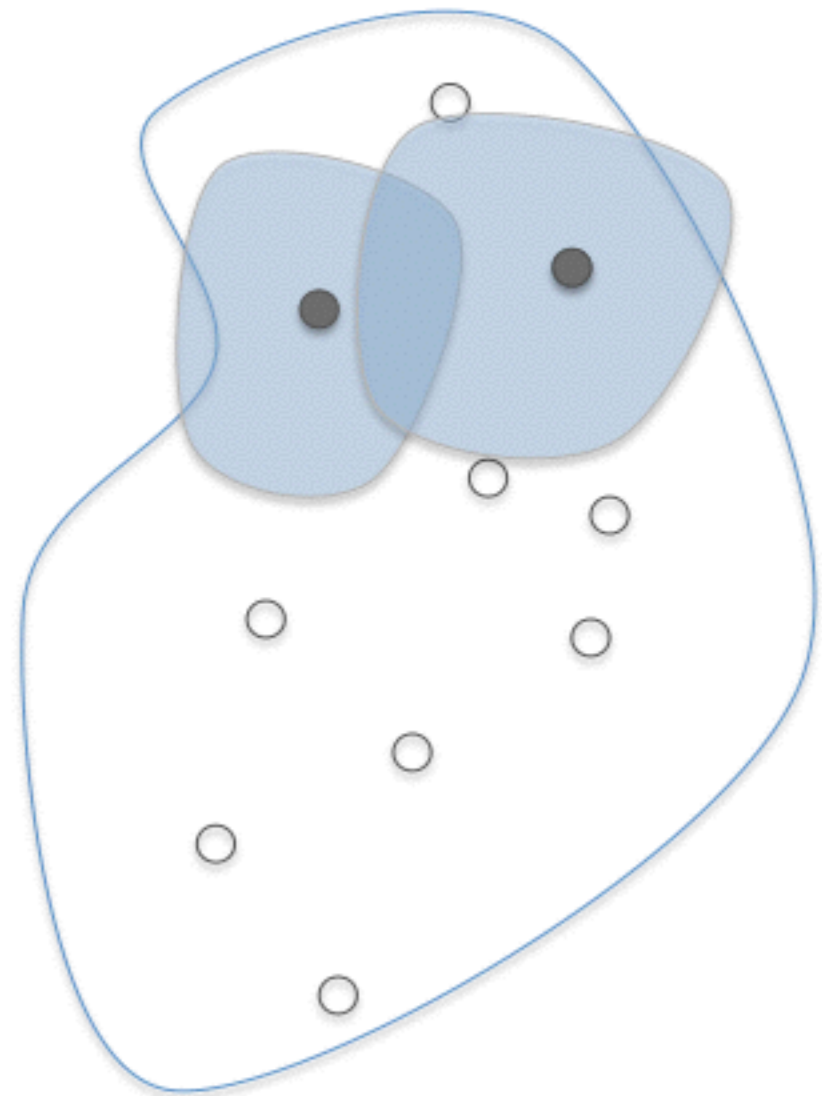
Marginal gains

- Observe:
- Marginal coverage depends on other sensors in the selection



Marginal gains

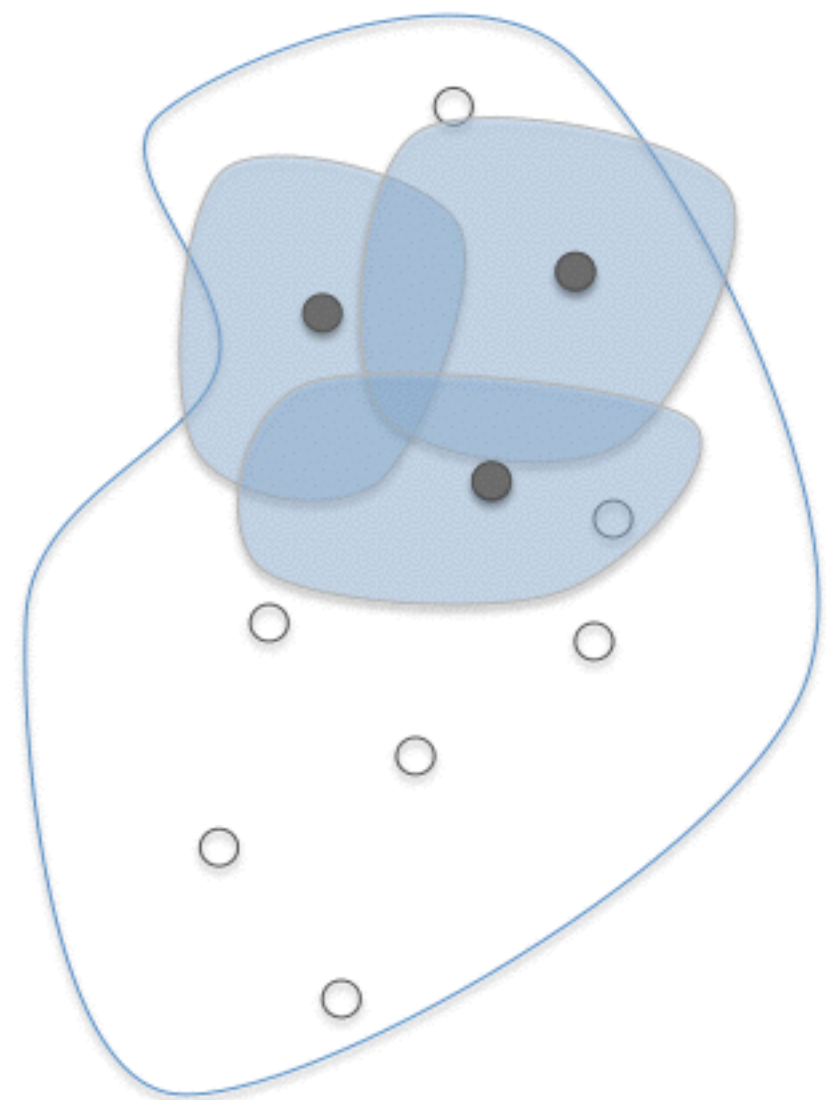
- Observe:
- Marginal coverage depends on other sensors in the selection

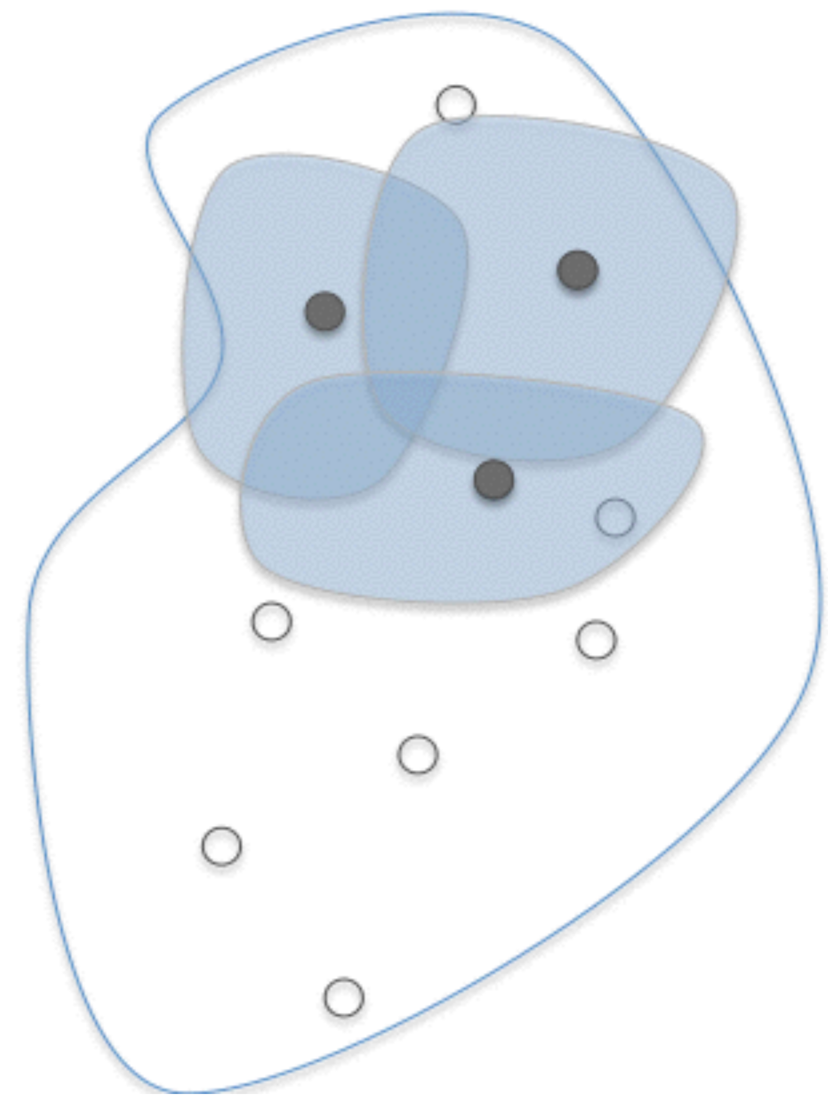


- Observe:

- Marginal coverage depends on other sensors in the selection

- More selected sensors means less marginal gain from each individual



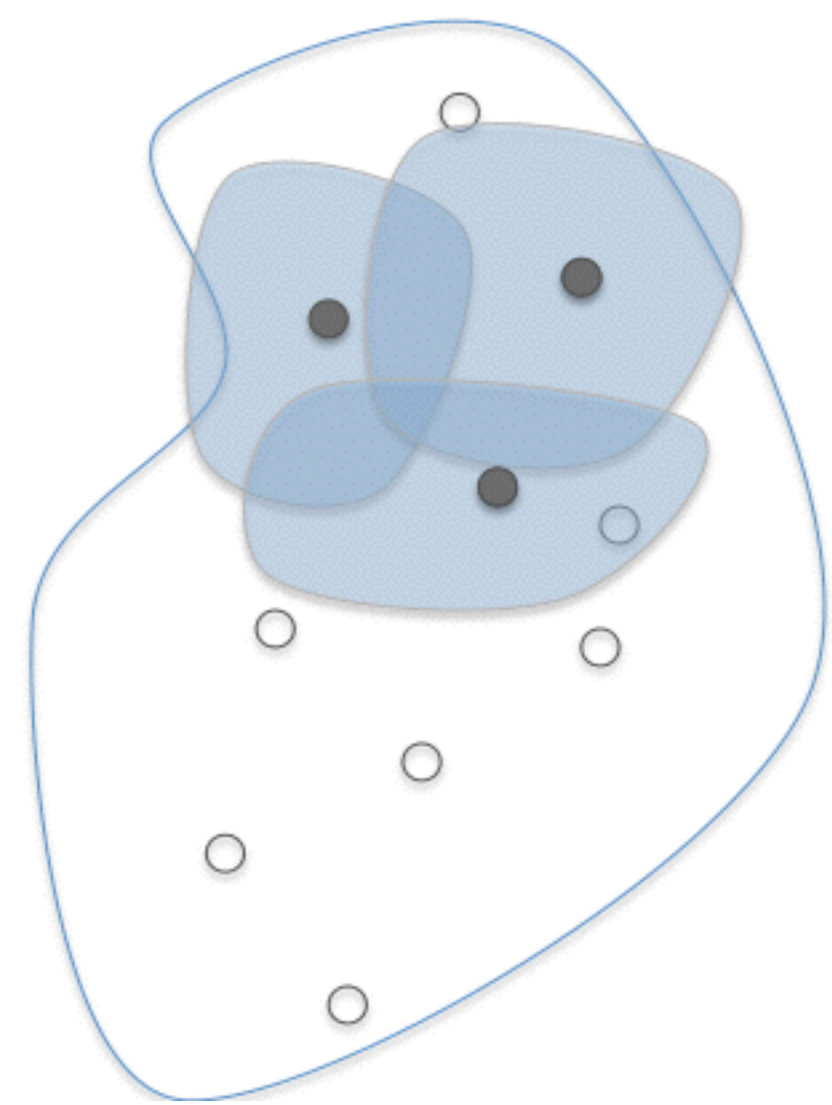


$$S \subseteq T \implies$$

$$f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T)$$

- Our Problem: select locations set of size k maximizes coverage

- NP-Hard



Greedy Approximation algorithm

- Start with empty set S = ∅
- Repeat k times:
  - Find v that gives maximum marginal gain:

$$f(S \cup \{v\}) - f(S)$$

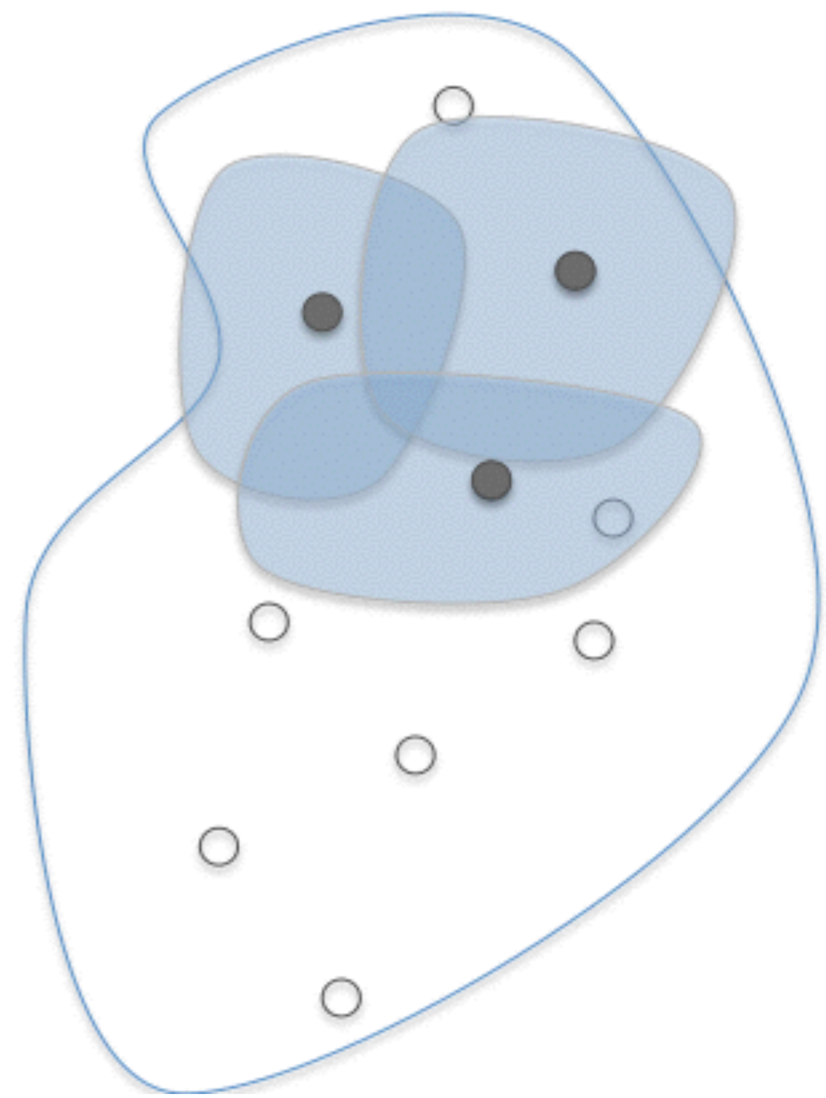
  - Add insert v into S

- Observation 1: Coverage function is submodular

- Observation 2: Coverage function is monotone:

- Adding more sensors always increases coverage

$$S \subseteq T \Rightarrow f(S) \le f(T)$$



Theorem

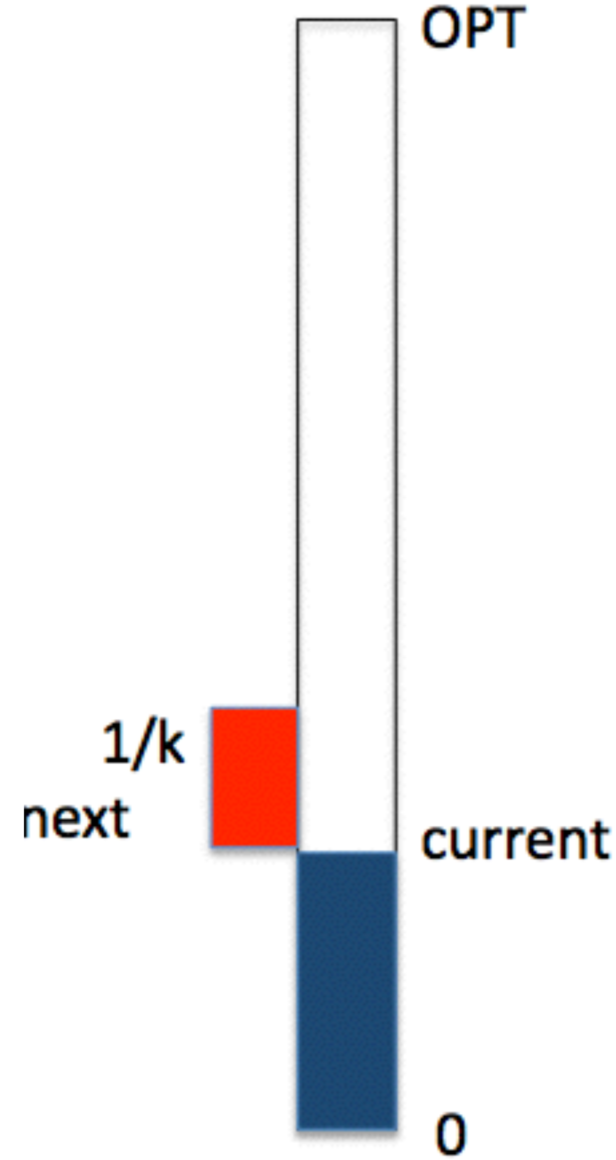
- For monotone submodular functions, the greedy algorithm produces an $\left(1 - \frac{1}{e}\right)$ approximation

- That is, the value f(S) of the final set is at least

$$\left(1 - \frac{1}{e}\right) \cdot OPT$$

Proof

- Idea:
- OPT is the max possible
- On every step there is at least one element that covers 1/k of remaining:
  - (OPT - current) * 1/k
- Greedy selects that element

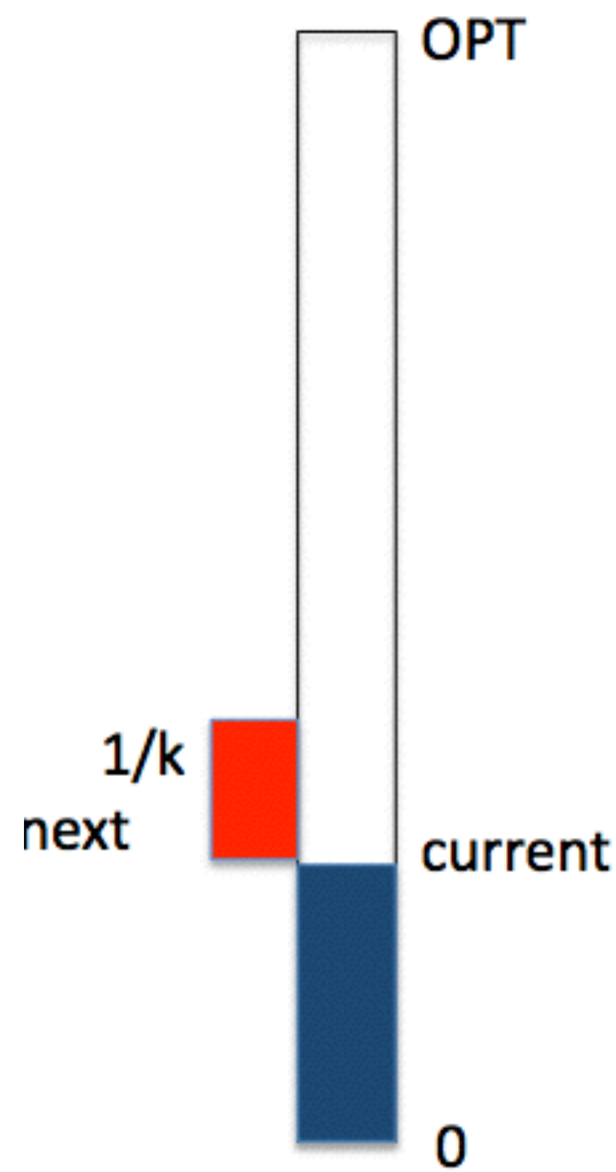


Proof

- At each step coverage remaining becomes

$$\left(1 - \frac{1}{k}\right)$$

- Of what was remaining after previous step



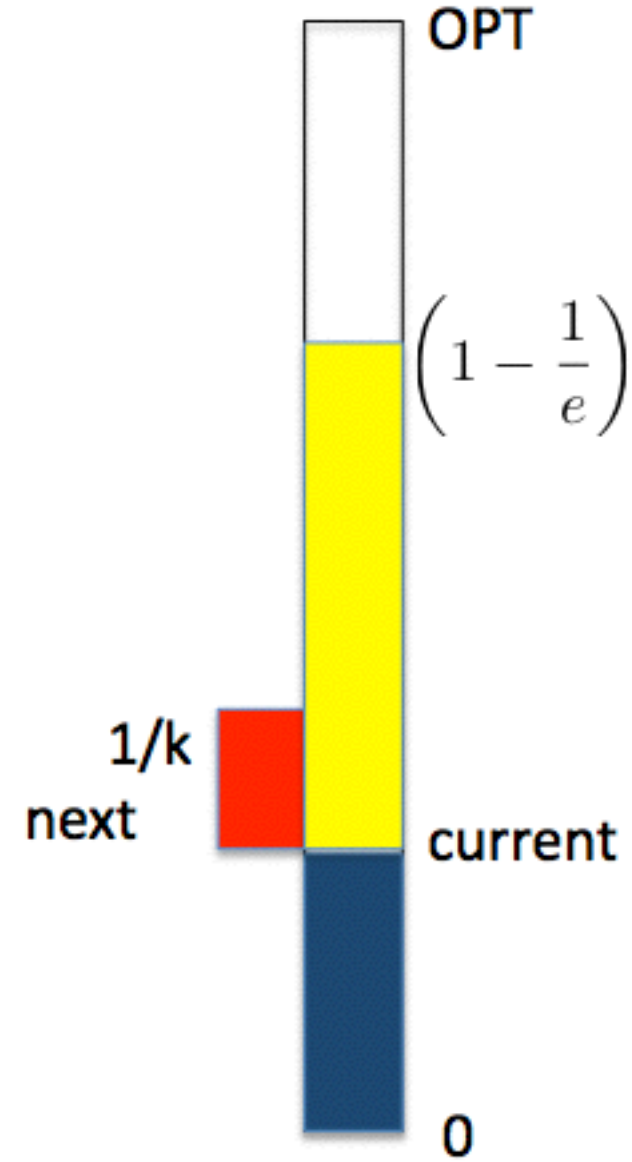
Proof

- After k steps, we have remaining coverage of OPT

$$\left(1 - \frac{1}{k}\right)^k \simeq \frac{1}{e}$$

- Fraction of OPT covered:

$$\left(1 - \frac{1}{e}\right)$$



- We have shown that monotone submodular maximization can be approximated using greedy selection

- To show that maximizing spread of cascading influence can be approximated:
  - We will show that the function is monotone and submodular