華東師範大學
EAST CHINA NORMAL UNIVERSITY

数据科学与工程算法基础

Algorithm Foundations of Data Science and Engineering

第十二章　子模函数及其应用

$$(1+x)^n = 1 + \frac{nx}{1!} + \frac{n(n-1)x^2}{2!} + \cdots$$

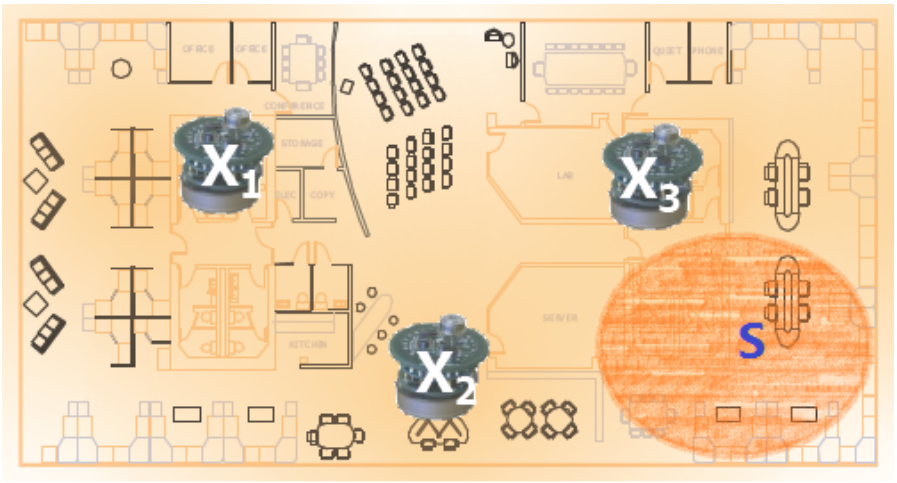
课程提纲

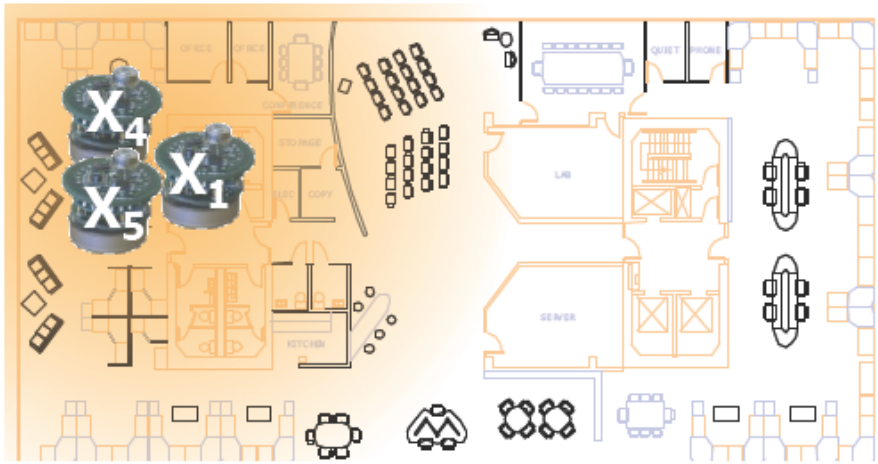
Content

课程提纲

Content

WiFi 布置



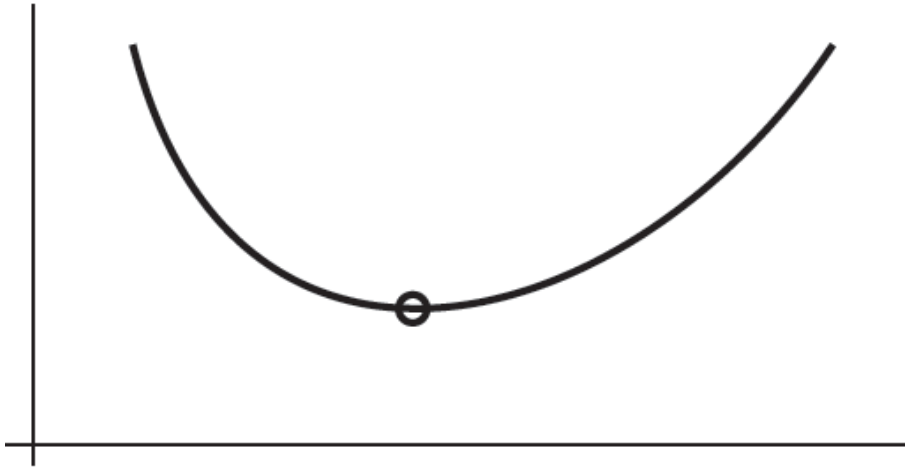


- 函数 $f(A)$ 度量所有布置的WiFi覆盖空间的大小
- 例如
  - 当 $A = \{X_1, X_2, X_3\}$ 时，覆盖范围广，因此 $f(A)$ 值大
  - 当 $B = \{X_1, X_4, X_5\}$ 时，覆盖范围小，因此 $f(B)$ 值小
- 类似的问题还有很多
  - 线下门店选址
  - 仓库选址
  - 分支机构设置
- 都是集合函数的实例

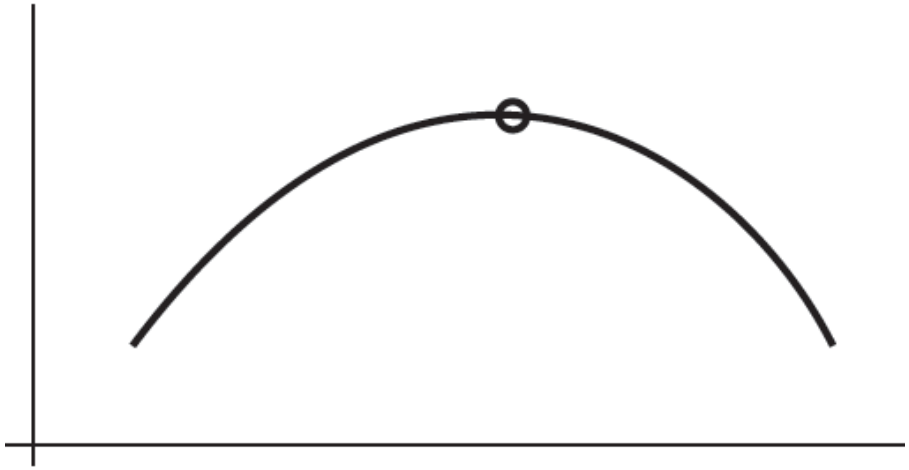
集合函数

- 给定有限集 $V = \{1,2,\ldots,n\}$，集合函数定义为 $f: 2^V \to \mathbb{R}$ 或 $f: \{0,1\}^n \to \mathbb{R}$，其中 2^V 为 V 的幂集。即对 $\forall A \subseteq V, f(A) \in \mathbb{R}$。
- 集合函数的重要性质
  - 单调性：$\forall A \subseteq B \subseteq V,\ f(A) \leq f(B)$
  - 非负性：$\forall A \subseteq V,\ f(A) \geq 0$
  - 规范性：$f(\varnothing) = 0$
- 例如，WiFi 信号覆盖、设施选址、文本摘要等都是集合函数

连续优化



- 如果函数 $f:\mathbb{R}^n \to \mathbb{R}$ 是凸函数，则能最小化



- 如果函数 $f:\mathbb{R}^n \to \mathbb{R}$ 是凹函数，则能最大化

离散优化

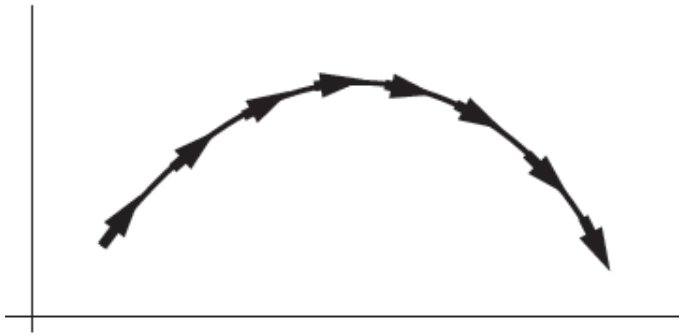
如果 $f(\cdot)$ 是集合函数，如何求解 $f(\cdot)$ 的最大值或者最小值?

课程提纲

Content

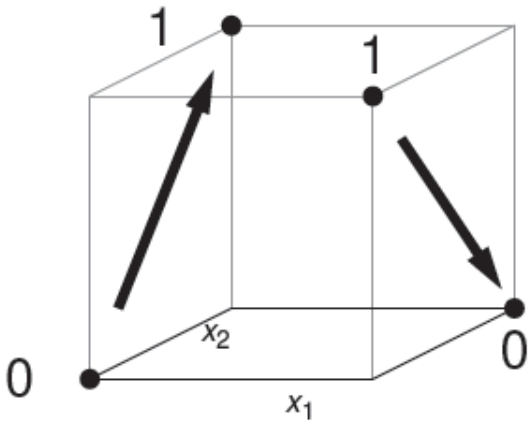
从凹函数到子模

Concavity:



- 如果连续函数 $f:\mathbb{R}^n \to \mathbb{R}$ 的导数 $f'(x)$ 是非增的，则 $f(x)$ 是凹函数

Submodularity:

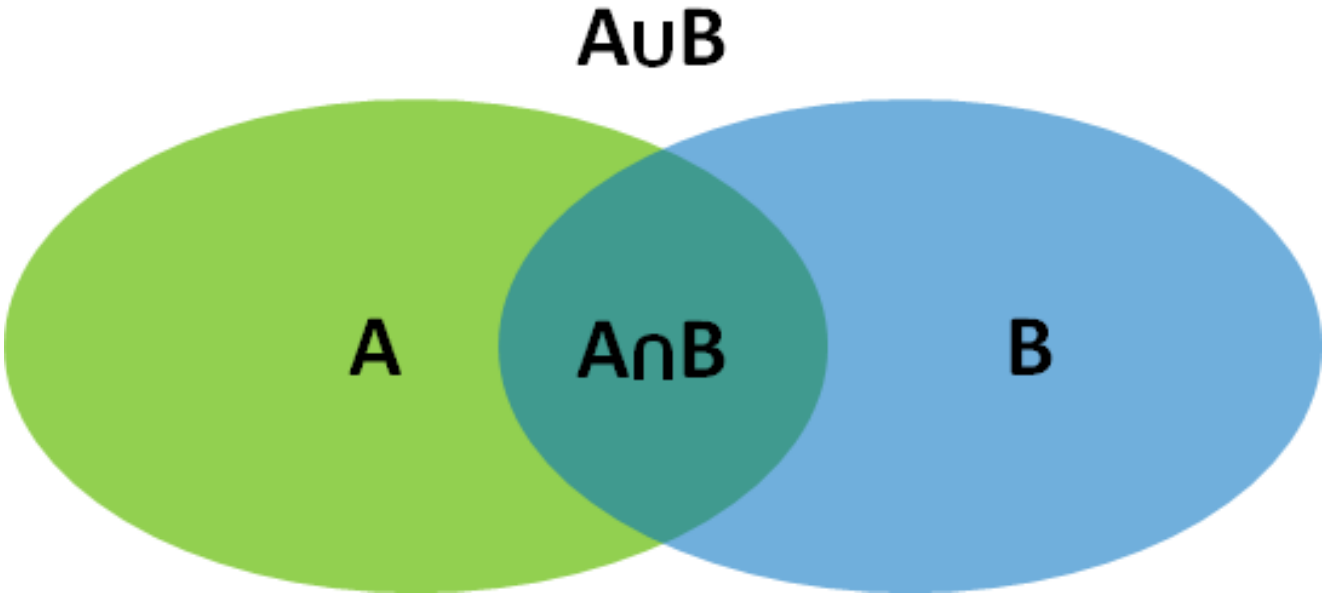


- 对集合函数 $f:\{0,1\}^n \to \mathbb{R}$，$\forall i, \partial_i f(x) = f(x+e_i) - f(x)$ 是非增的，则 $f(x)$ 是子模函数

子模函数

- 集合函数 $f:\{0,1\}^n \to \mathbb{R}$ 是子模函数当且仅当对 $\forall A, B \subseteq V$，以下不等式成立

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

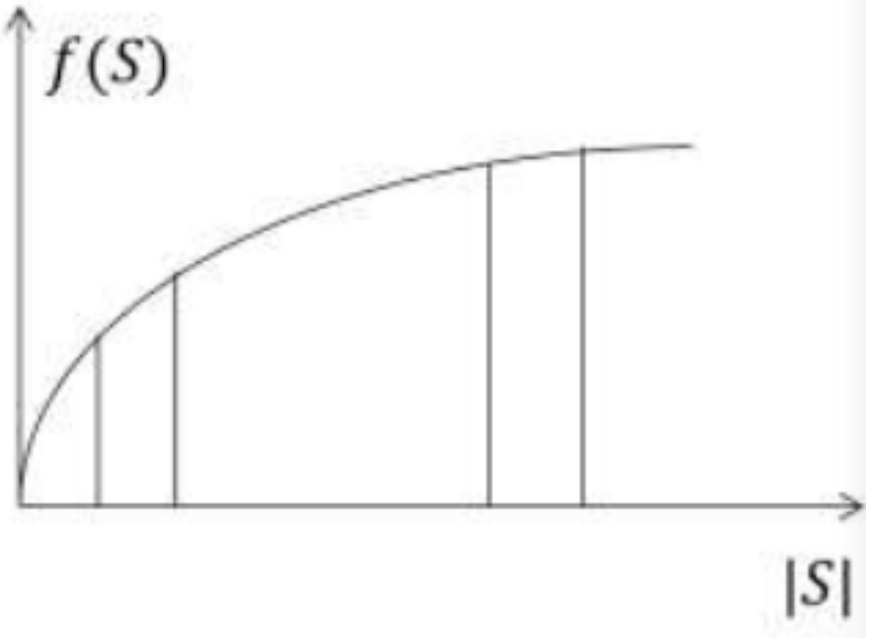


子模函数的等价定义

- 两个等价定义
  - 边际效用递减：$\forall A \subset B \subseteq V$ 和 $\forall v \in V \backslash B$，满足

    $f(A+v)-f(A) \geq f(B+v)-f(B)$
  - 集合效用递减：$\forall A \subset B \subseteq V$ 和 $\forall C \subseteq V \backslash B$，满足

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



等价性证明

- $f(A)+f(B)\geq f(A\cap B)+f(A\cup B)\Leftrightarrow$
  $f(A+v)-f(A)\geq f(B+v)-f(B)$
- **证明：（必要性）**

  令 $A\subset B$，给定集合 $A\cup\{v\}$ 和 B，若 $v\notin B$，则

  $$f(A\cup\{v\})+f(B)\geq f((A\cup\{v\})\cap B)+f((A\cup\{v\})\cup B)$$

  注意到 $A\cup\{v\}\cup B=B\cup\{v\}$ 和 $(A\cup\{v\})\cap B=A$

  因此，我们得到

  $$f(A\cup\{v\})+f(B)\geq f(A)+f(B\cup\{v\})$$

  即得证。

等价性证明（续）

- $f(A)+f(B)\geq f(A\cap B)+f(A\cup B)\Leftrightarrow$
  $f(A+v)-f(A)\geq f(B+v)-f(B)$
- **证明：（充分性）**

对任意两个集合 $A,B\subseteq V$，$B\backslash A=\{v_1,\ldots,v_k\}$，$B_j=\{v_1,\ldots,v_j\}$，$S_j=(A\cap B)\cup B_j$ 和 $T_j=A\cup B_j$，容易知道 $\forall j=0,1,\ldots,k-1$

$$f(S_j+v_{j+1})-f(S_j)\geq f(T_j+v_{j+1})-f(T_j)$$

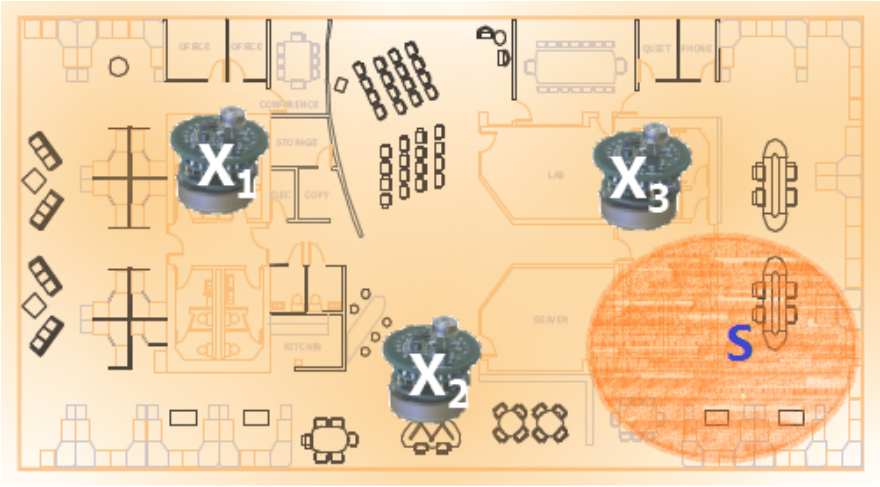
将 k 个式子求和得到

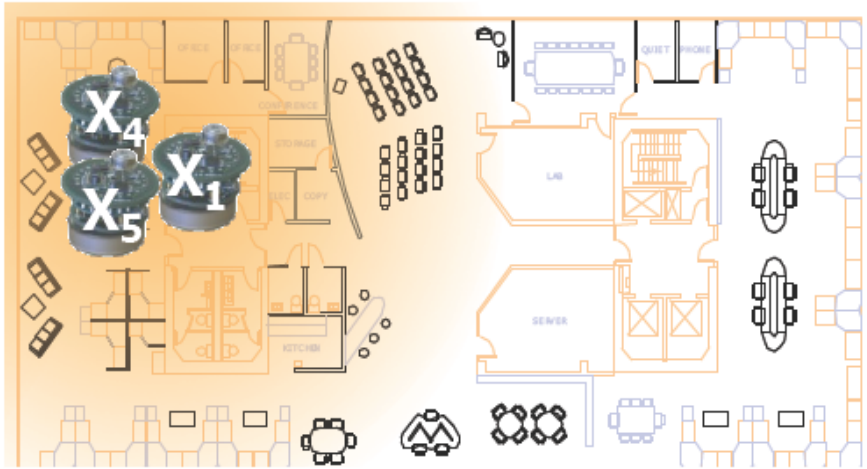
$$f(A)+f(B)\geq f(A\cap B)+f(A\cup B)$$

子模函数应用

- 博弈论
  - 子模函数是对边际效用递减规律的数学刻画
  - 在其他商品消费量保持不变的条件下，消费某商品的效用增量（即边际效用）是递减的
  - 边际效用递减的原因在于
    - 生理或心理的原因
    - 物品本身用途的多样性
- 机器学习
  - 子模函数经常作为机器学习任务的目标函数
  - 例如传感器布置、文本摘要、特征选取、信息传播等问题

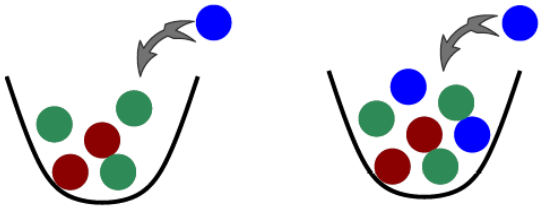
WiFi 布置

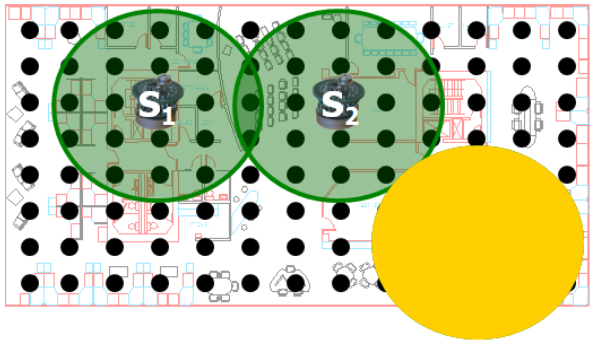


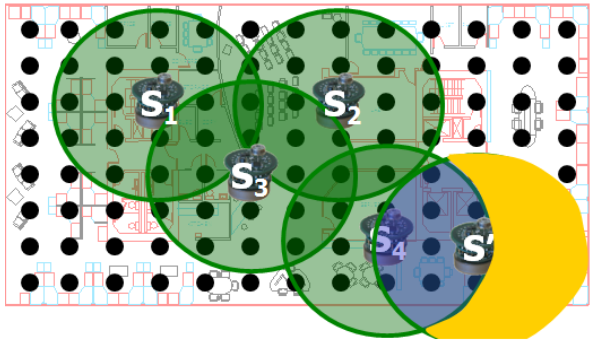


- 信号边际增益
  - $\Delta_f(s \mid A) = f(A + s) - f(A)$
  - 若 $A = \{X_1, X_2\}$，多布置一个 s
  - 若 $B = \{X_1, X_2, X_3\}$，多布置一个 s
- 信号边际收益递减
  - $\Delta_f(s \mid A) \geq \Delta_f(s \mid B)$
  - $f(A + s) - f(A) \geq f(B + s) - f(B)$

颜色计数和集合覆盖



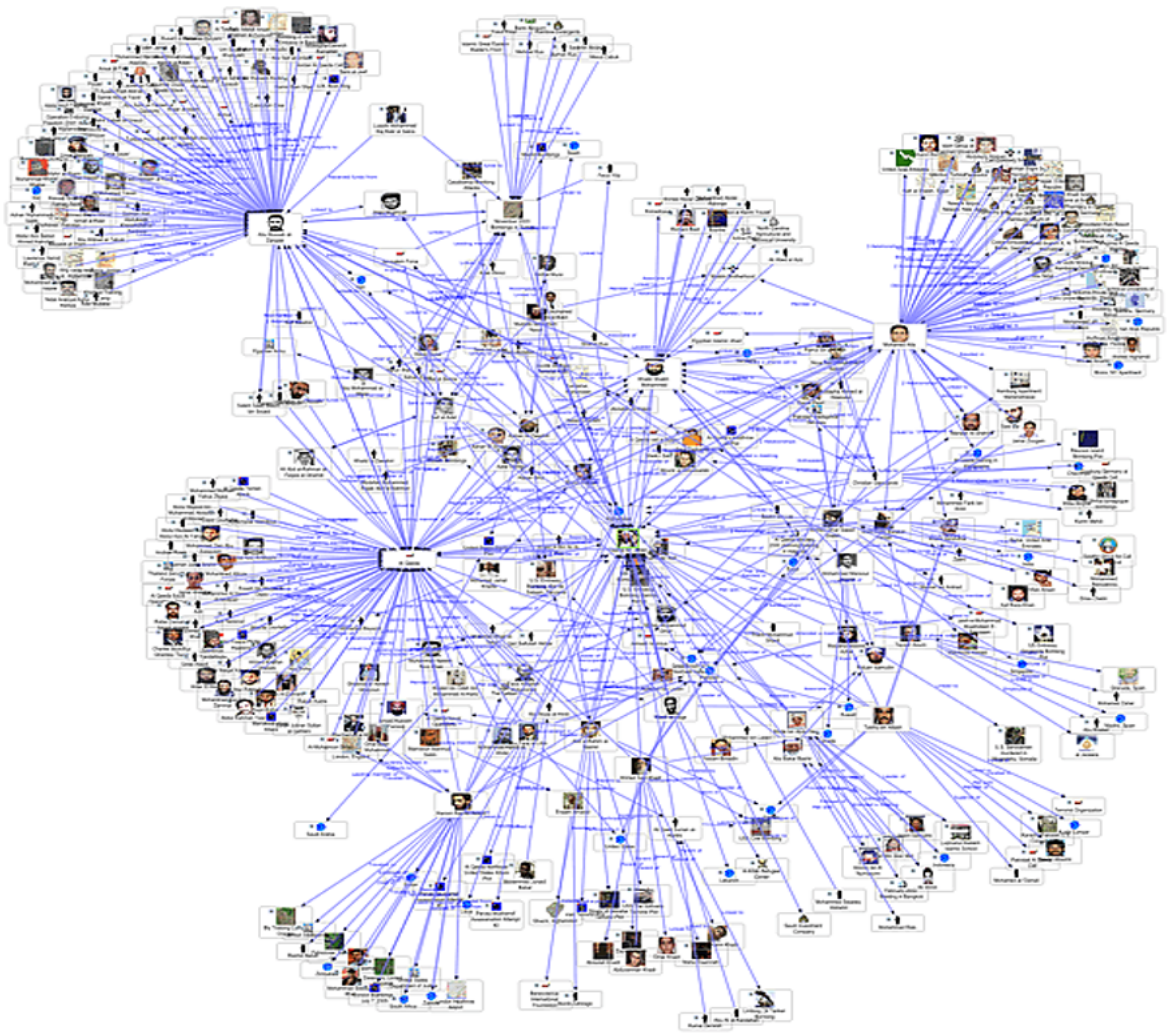




- 颜色计数
  - 假设罐子中多个球构成集合 S， $f(S)$ 衡量罐子中不同颜色球的个数
  - 罐子中球越多，往其中放一个球，和已有颜色不同的概率越低
  - 因此， $f(S)$ 是一个子模函数
- 集合覆盖
  - 若 $A = \{S_1, S_2\}$， $B = \{S_1, S_2, S_3, S_4\}$ 表示句子集合,定义 $f(A)$ 为句子集合 A 中词的数量
  - $\Delta_f(s \mid A) \geq \Delta_f(s \mid B)$
  - 因此， $f(A)$ 为一个子模函数

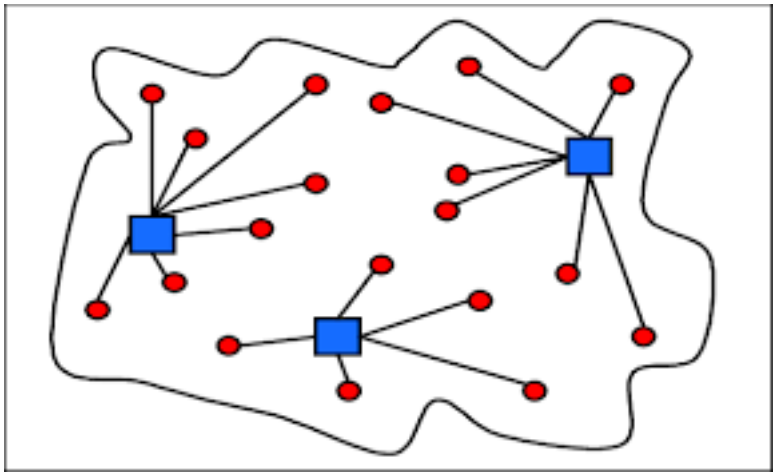
社交网络影响力

- 网络拓扑结构
- “种子用户”
- 以“种子用户”为起始点信息在网络上传播的范围



设施选址

- 用户位置集合
- 候选设施位置集合
- 选取k个设施位置，使得用户距离它们的位置最近
- $f(S) = \sum_{i=1}^{m} \max_{j \in S} M_{ij}$
- 其中 M_{ij} 与设施 j 和用户 i 的距离负相关



子模的性质

- 给定集合 V，f_1, f_2 均为子模函数
  - $\forall a_1, a_2 \geq 0$，$a_1 f_1 + a_2 f_2$ 也是一个子模函数
  - 对 $S \subseteq V$，集合函数 $f'(A) = f(A \cap S)$ 是一个子模函数
  - $\bar{f}(A) = f(V \backslash A)$ 是一个子模函数
  - 其他性质见课本定理 12.2
- 更有用的性质
  - $f_\theta(A)$ 是一个子模函数，则期望 $\sum_\theta f_\theta(A) p(\theta)$ 是一个子模函数
  - f_i 为子模函数且$\lambda_i \geq 0$，则 $\sum_i \lambda_i f_i(A)$ 是一个子模函数

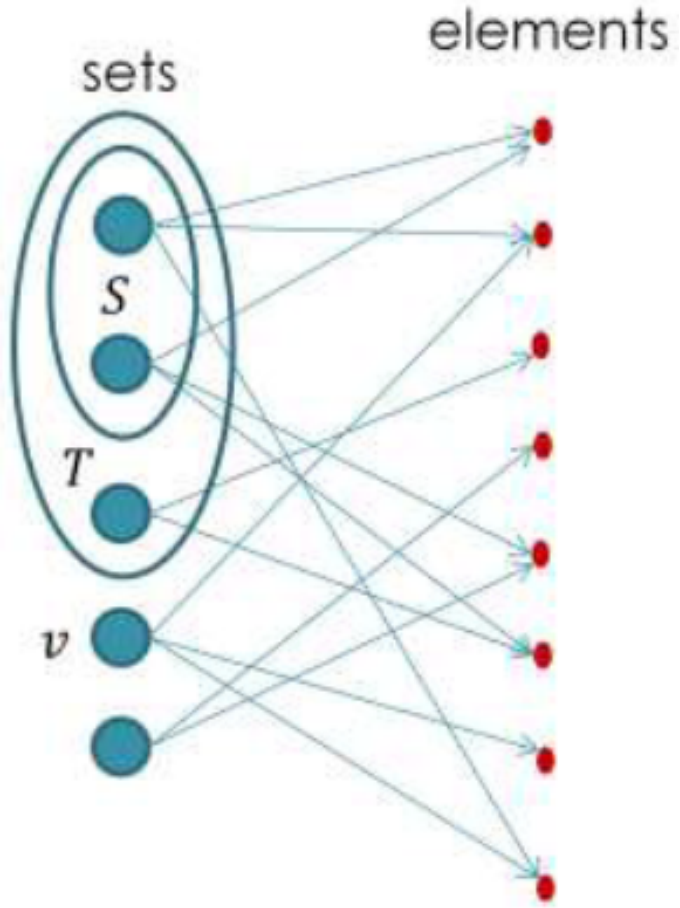
组合优化

- 组合优化可以表示成 $\arg\max_{S \in \mathcal{F}} f(S)$ 或者 $\arg\min_{S \in \mathcal{F}} f(S)$
- 其中 $\mathcal{F}$ 为离散的可行解集合
  - 很多问题可以建模成组合优化问题，例如最小切割、最大切割、顶点覆盖、集合覆盖等
  - 其中有一些目标函数是子模函数
    - WiFi 信号覆盖
    - 集合覆盖
    - 信息传播
    - 设施选址
    - ……

课程提纲

Content

集合覆盖问题



- 集合覆盖
  - S 表示句子，为单词或者关键词构成的集合
  - 定义 $f(A)$ 为句子集合 A 中包含词的数量
  - 已知 $f(A)$ 是子模函数
- 集合全覆盖
  - 找到最小的句子集合包含所有的单词
- k–最大覆盖问题
  - 找到 k 个句子使得其包含的单词数量最多

例子：2–最大覆盖

- 给定全集 $U = \{a, b, c, \ldots, l\}$
- 子集 $A_1 = \{a, b, c, d\}$，$A_2 = \{e, f, g, h\}$，
  $A_3 = \{i, j, k, l\}$，$A_4 = \{a, e\}$，$A_5 = \{b, f, g, i\}$，
  $A_6 = \{c, d, g, h, k, l\}$，$A_7 = \{l\}$
  - 集合 A_6 包含 6 个元素，A_1, A_2, A_3, A_5 包含 4 个元素
  - 由于$|A_6 \cup A_1| = 8$，$|A_6 \cup A_2| = 8$，$|A_6 \cup A_3| = 8$，
    $|A_6 \cup A_5| = 9$
- 因此，$\{A_5, A_6\}$ 为 2–最大集合覆盖

抽取式文本摘要

- **定义：**关键词集合 $V = \{w_1, \ldots, w_n\}$ 和句子集合$\mathcal{S} = \{S_1, \ldots, S_m\}$，其中 $S_i \subseteq V$，抽取式文本摘要的目标是找 k 个句子可能多的覆盖关键词
  - 令 $\mathscr{C}$ 为找出的 k 个句子，设

$$x_i = \begin{cases} 1, S_i \in \mathscr{C} \\ 0, S_i \notin \mathscr{C} \end{cases} \text{ 和 } y_j = \begin{cases} 1, w_j \in \bigcup_{S \in \mathcal{T}} S \\ 0, w_j \notin \bigcup_{S \in \mathcal{T}} S \end{cases}$$

  - 整数规划问题

$$\begin{aligned} \max \quad & \sum_j y_j \\ \text{s.t.} \quad & \sum_i x_i \leq k \\ & \sum_{w_j \in S_i} x_i \geq y_j \\ & x_i \in \{0,1\}, \forall i \in [m] \\ & y_j \in \{0,1\}, \forall j \in [n] \end{aligned}$$

- 除了求解整数规划问题，还可以利用贪心算法近似求解

课程提纲

Content

爬山算法

1. 初始化 $\mathscr{C} = \varnothing$
2. 对 $i = 1,2,\ldots,k$ 到
3. 　　选择 $S^* = \arg\max_{S \in V \setminus \mathscr{C}} f(\mathscr{C} \cup \{S\}) - f(\mathscr{C})$
4. 　　$\mathscr{C} = \mathscr{C} \cup \{S^*\}$
5. 输出 $\mathscr{C}$

- **定理：**如果集合函数 $f(\cdot)$ 是单调非减的子模函数，且 $f(\varnothing) = 0$，则爬山算法的近似率至少是最优解的 $1 - \frac{1}{e} \approx 0.632$，即找到的 $\mathscr{C}$ 满足 $f(\mathscr{C}) \geq (1 - \frac{1}{e}) \cdot \max_{\mathscr{T} \subseteq \mathscr{S} : |\mathscr{T}| = k} f(\mathscr{T})$
- 该结论表明运用爬山算法的贪心策略解决满足子模定义的目标函数的最大化问题是一个还不错的选择

爬山算法的理论证明（1）

Proof Nemhauser et al. only discussed the case $\ell = k$, however their very elegant argument easily yields the slight generalization above. It goes as follows. Fix ℓ and k. Let $S^* \in \arg\max\{f(S) : |S| \leq k\}$ be an optimal set of size k (due to monotonicity of f we can assume w.l.o.g. it is of size exactly k), and order the elements of S^* arbitrarily as $\{v_1^*, \ldots, v_k^*\}$. Then we have the following sequence of inequalities for all $i < \ell$, which we explain below.

$$
\begin{aligned}
f(S^*) &\leq f(S^* \cup S_i) && (3)\\
&= f(S_i) + \sum_{j=1}^{k} \Delta\left(v_j^* \mid S_i \cup \left\{v_1^*, \ldots, v_{j-1}^*\right\}\right) && (4)\\
&\leq f(S_i) + \sum_{v \in S^*} \Delta(v \mid S_i) && (5)\\
&\leq f(S_i) + \sum_{v \in S^*} \left(f(S_{i+1}) - f(S_i)\right) && (6)\\
&\leq f(S_i) + k\left(f(S_{i+1}) - f(S_i)\right) && (7)
\end{aligned}
$$

Eq. (3) follows from monotonicity of f, Eq. (4) is a straightforward telescoping sum, Eq. (5) follows from the submodularity of f, Eq. (6) holds because S_{i+1} is built greedily from S_i in order to maximize the marginal benefit $\Delta(v \mid S_i)$, and Eq. (7) merely reflects the fact that $|S^*| \leq k$. Hence

爬山算法的理论证明（2）

$|S^*| \leq k$. Hence

$$f(S^*) - f(S_i) \leq k\left(f(S_{i+1}) - f(S_i)\right). \tag{8}$$

Now define $\delta_i := f(S^*) - f(S_i)$, which allows us to rewrite Eq. (8) as $\delta_i \leq k\left(\delta_i - \delta_{i+1}\right)$, which can be rearranged to yield

$$\delta_{i+1} \leq \left(1 - \frac{1}{k}\right)\delta_i \tag{9}$$

Hence $\delta_\ell \leq \left(1 - \frac{1}{k}\right)^\ell \delta_0$. Next note that $\delta_0 = f(S^*) - f(\emptyset) \leq f(S^*)$ since f is nonnegative by assumption, and by the well-known inequality $1 - x \leq e^{-x}$ for all $x \in \mathbb{R}$ we have

$$\delta_\ell \;\leq\; \left(1 - \frac{1}{k}\right)^\ell \delta_0 \;\leq\; e^{-\ell/k} f(S^*). \tag{10}$$

Substituting $\delta_\ell = f(S^*) - f(S_\ell)$ and rearranging then yields the claimed bound of $f(S_\ell) \geq \left(1 - e^{-\ell/k}\right) f(S^*)$. □

集合覆盖：单调子模函数

- 关键词覆盖 $f(\mathscr{C})$ 定义为 $f(\mathscr{C}) = |\bigcup_{S \in \mathscr{C}} S|$
- $f(\mathscr{C})$ 满足三个关键特性
  - $f(\varnothing) = 0$
  - 单调性
  - 子模性
- 因此，运用爬山算法解决 k–最大集合覆盖问题是一个不错的选择

抽取式文本摘要示例（第一轮）

- 提取关键词集合 $V = \{w_1, w_2, \ldots, w_8\}$
- 候选句子 $S_1 = \{w_1, w_2, w_8\}$，$S_2 = \{w_1, w_3, w_7\}$，$S_3 = \{w_1, w_6\}$，$S_4 = \{w_1, w_3, w_7, w_8\}$，$S_5 = \{w_1, w_5, w_6\}$，$S_6 = \{w_1, w_5, w_8\}$，$S_7 = \{w_5\}$，$S_8 = \{w_1, w_4, w_6\}$，$S_9 = \{w_2, w_8\}$

句子	f(C)	f(C + {S})	Delta(S|C)
S1	0	3	3
S2	0	3	3
S3	0	2	2
S4	0	4	4
S5	0	3	3
S6	0	3	3
S7	0	1	1
S8	0	3	3
S9	0	2	2

句子 S_4 对关键词覆盖的边际贡献最大，因此第一轮迭代选择句子 S_4

抽取式文本摘要示例（第二轮）

- 提取关键词集合 $V = \{w_1, w_2, \ldots, w_8\}$
- 候选句子 $S_1 = \{w_1, w_2, w_8\}$，$S_2 = \{w_1, w_3, w_7\}$，$S_3 = \{w_1, w_6\}$，$S_4 = \{w_1, w_3, w_7, w_8\}$，$S_5 = \{w_1, w_5, w_6\}$，$S_6 = \{w_1, w_5, w_8\}$，$S_7 = \{w_5\}$，$S_8 = \{w_1, w_4, w_6\}$，$S_9 = \{w_2, w_8\}$

句子	f(C)	f(C + {S})	Delta(S|C)
S1	4	5	1
S2	4	4	0
S3	4	5	1
S5	4	6	2
S6	4	5	1
S7	4	5	1
S8	4	6	2
S9	4	5	1

句子 S_5 和 S_8 对关键词覆盖的边际贡献最大，因此第二轮迭代选择其中之一 S_5

抽取式文本摘要示例（第三轮）

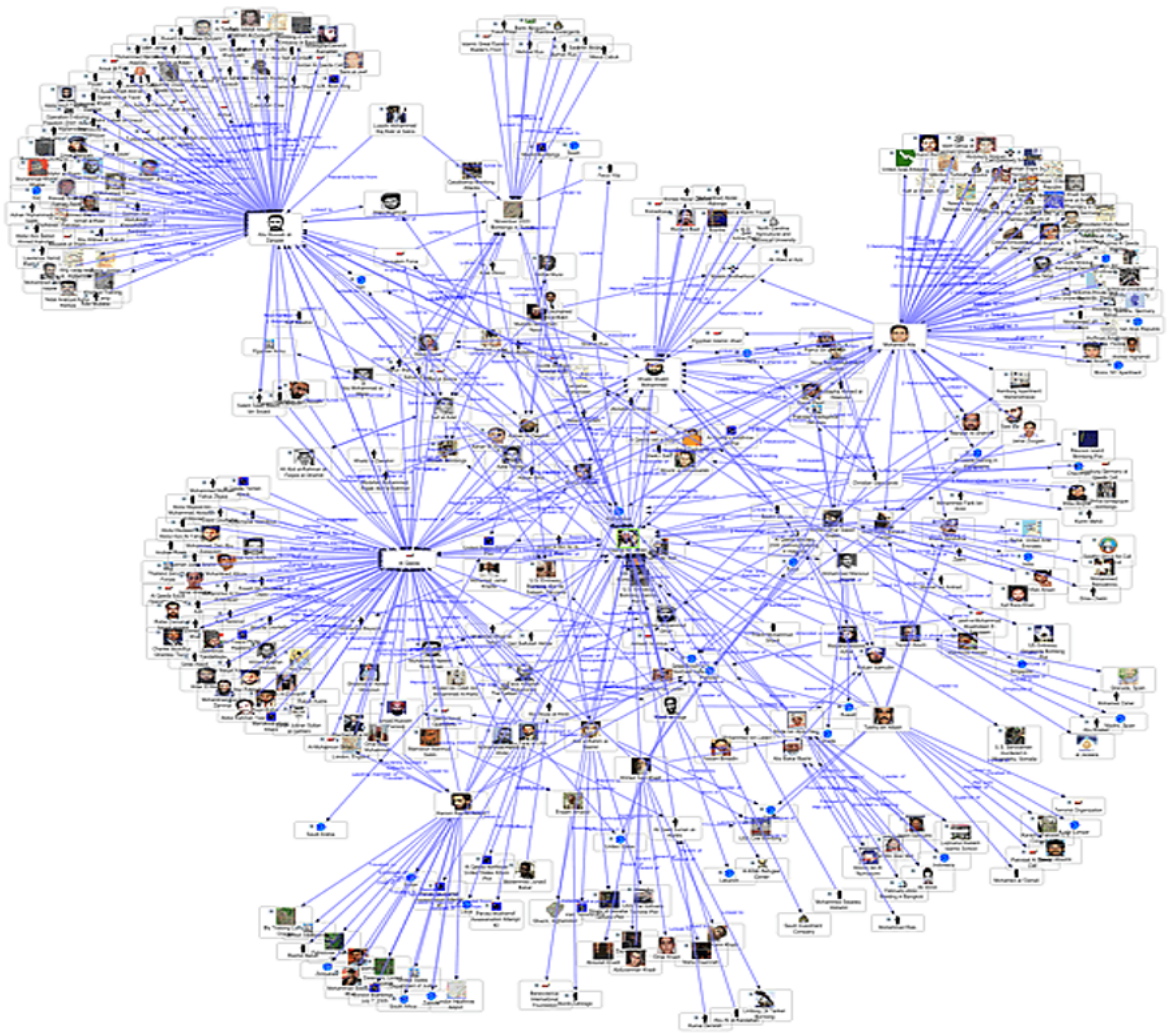
- 提取关键词集合 $V = \{w_1, w_2, \ldots, w_8\}$
- 候选句子 $S_1 = \{w_1, w_2, w_8\}$，$S_2 = \{w_1, w_3, w_7\}$，$S_3 = \{w_1, w_6\}$，$S_4 = \{w_1, w_3, w_7, w_8\}$，$S_5 = \{w_1, w_5, w_6\}$，$S_6 = \{w_1, w_5, w_8\}$，$S_7 = \{w_5\}$，$S_8 = \{w_1, w_4, w_6\}$，$S_9 = \{w_2, w_8\}$

句子	f(C)	f(C + {S})	Delta(S|C)
S1	6	7	1
S2	6	6	0
S3	6	6	0
S6	6	6	0
S7	6	6	0
S8	6	7	1
S9	6	7	1

句子 S_1，S_8 和 S_9 对关键词覆盖的边际贡献最大，因此第三轮迭代选择句子 S_1。最终文本摘要结果为 $\mathscr{C} = \{S_4, S_5, S_1\}$

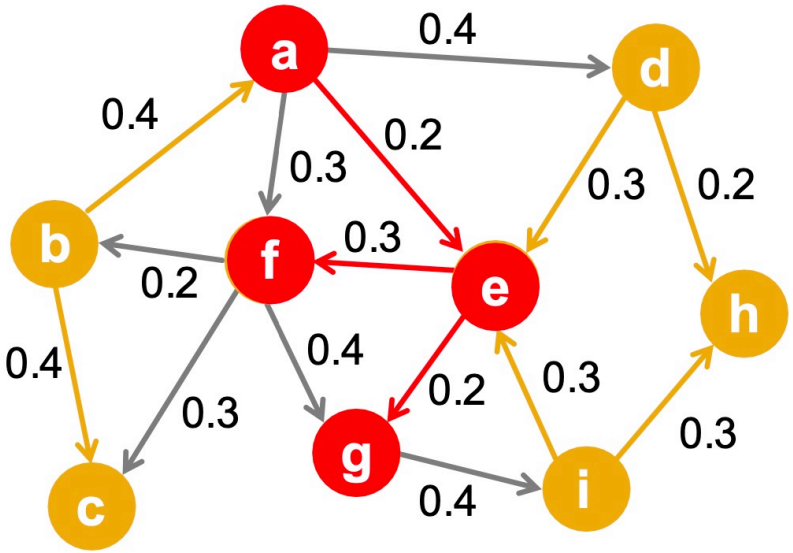
社交网络影响力最大化

- 网络拓扑结构
- “种子用户”
- 以“种子用户”为起始点信息在网络上传播的范围
- 如何选择k个“种子用户”?



Independent Cascade Model

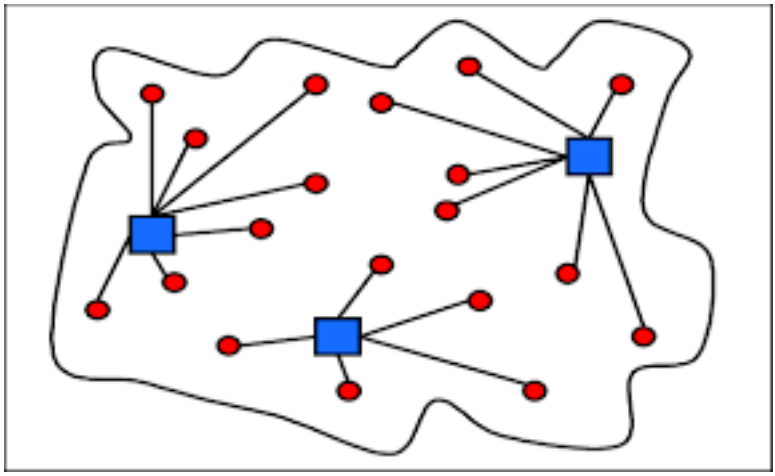
- **Initially some nodes *S* are active**
- Each edge (v, w) has probability (weight) p_{vw}



- **When node *v* becomes active:**
  - It activates each out-neighbor w with prob. p_{vw}
- **Activations spread through the network**

设施选址（如何解决？）

- 用户位置集合
- 候选设施位置集合
- 选取k个设施位置，使得用户距离它们的位置最近
- $f(S) = \sum_{i=1}^{m} \max_{j \in S} M_{ij}$
- 其中 M_{ij} 与设施 j 和用户 i 的距离负相关



本章小结

- 连续优化 VS. 组合优化
  - 凸凹性是解决连续优化问题的关键
  - 类似于凹函数，子模是离散函数边际效用递减的数学表述
  - 除了子模，还有超模函数，类似于连续情形下的凸函数
- 子模函数
  - 子模函数的运算性质
  - 运用爬山算法在保证一定精度的情况下解决单调子模优化问题
    - 集合覆盖
    - Wifi 布置
    - 文本摘要
    - ……