



# 超图(Hypergraph)理论与应用

# 动机(Motivation)

- 什么是共指消解(Coreference Resolution)
- 共指消解的各种方法
- 图分割(Graph Partitioning)方法
- 简单图分割方法的潜在缺陷
- 引入超图(Hypergraph)的意义

# 超图(Hypergraph)

- 超图的定义
- 超图的分割
- 超图真比简单图优越吗？
- 如何将超图运用到共指消解中

# 什么是共指消解

[李明<sub>i</sub>]怕[高妈妈<sub>j</sub>]一人呆在家里寂寞,[他<sub>i</sub>]便将[他自己<sub>i</sub>]家里的电视搬了过来给[她<sub>j</sub>]。

# 共指消解的方法

## ■ 规则方法

利用句法层面的知识，进行启发式消解。

## ■ 统计方法

基于训练语料库，统计出概率分布，然后进行预测。

## ■ 机器学习

决策树、朴素贝叶斯、规则学习等等。

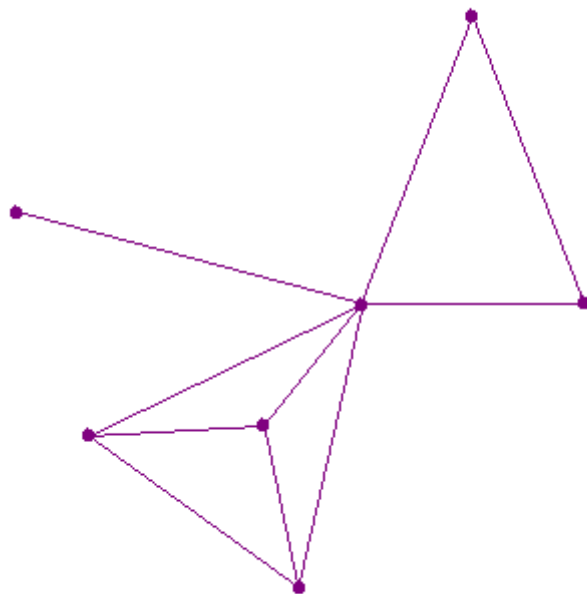
## ■ 图方法

以节点表示名词短语，以边表示名词短语间的共指关联度。

# 图方法

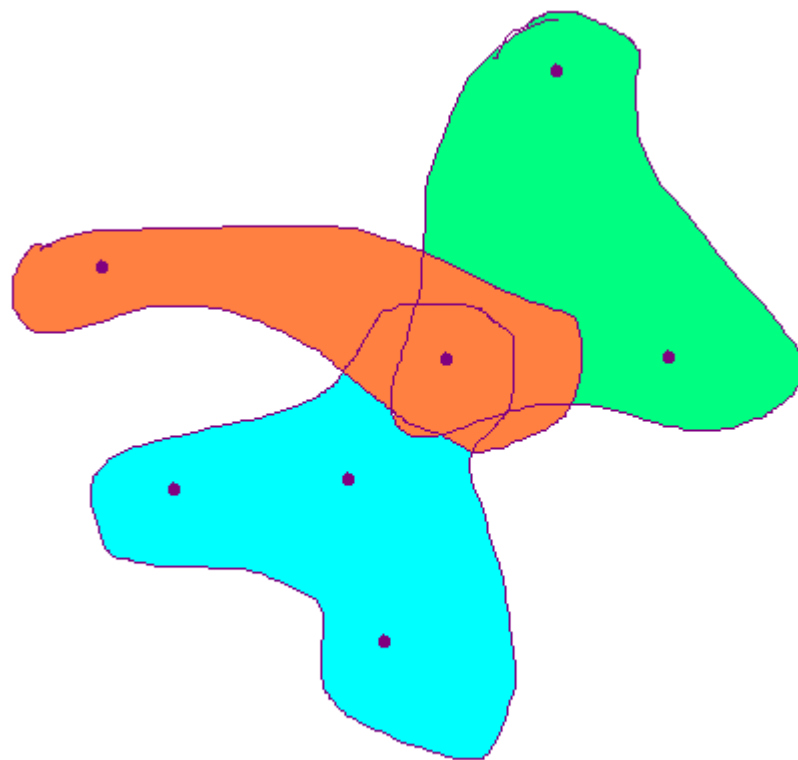
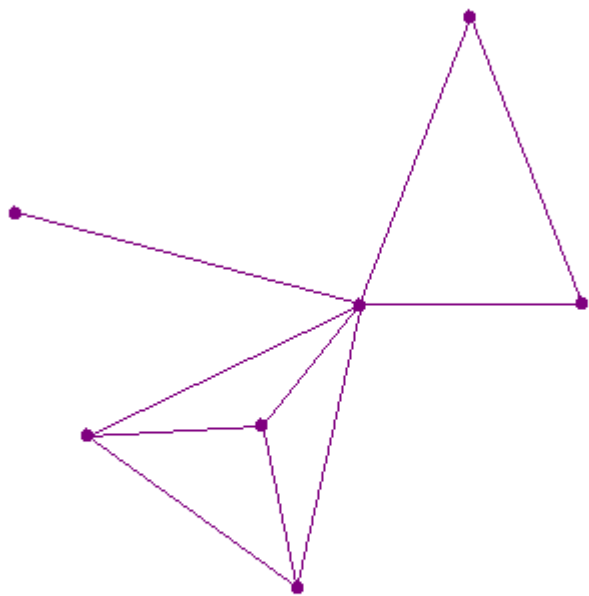
- 节点表示名词短语
- 边表示短语与短语之间的某种关联(这种关联必须要对“共指”起到贡献，如人称、性别、单复数等属性)
- 边的权值用来表示这种关联对共指起到的贡献的大小

# 简单图



一条边只能连接两个顶点

# 超图

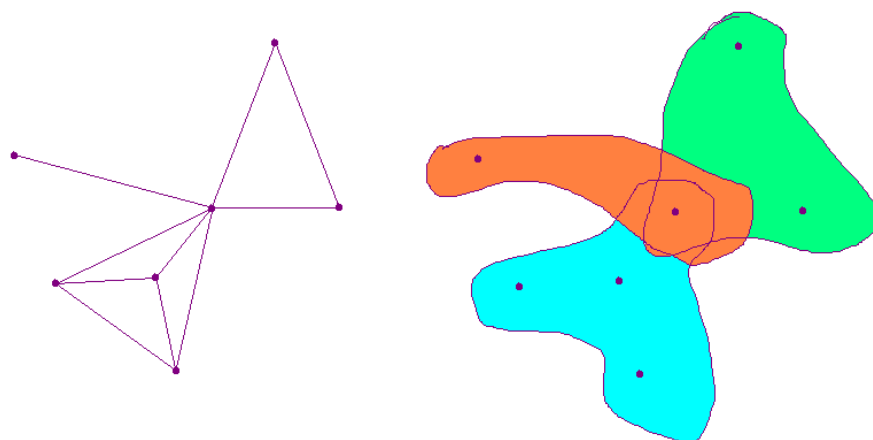


一条边可以连接多个顶点



# 为什么引入超图(一个例子)

顶点代表文章，每条边代表两个顶点（文章）享有同一个作者

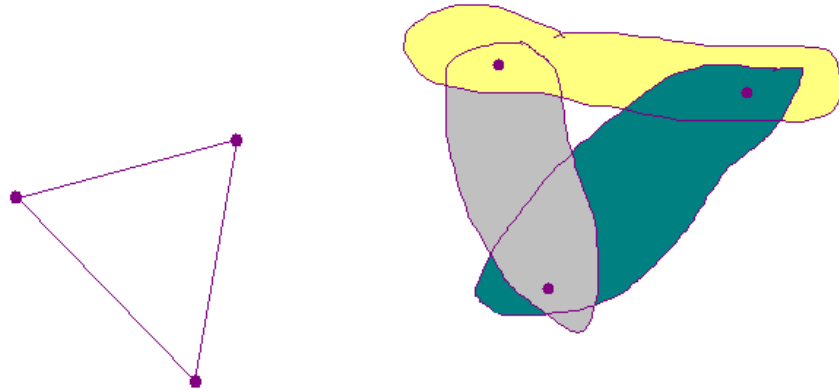


简单图版本丢失了“**同一作者的多篇文章**”这一信息，而超图版本则保存了这一信息。

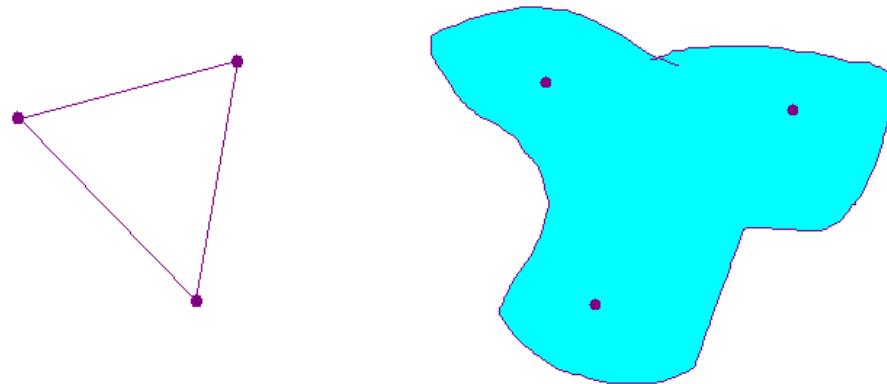
在共指消解里面，也有类似的信息，比如“多个指代的性别(**gender**)相同”、“多个指代的数量相同”(即同为单数或同为复数)等。

# 为什么引入超图(一个例子)

- 假设有三篇文章， $v_1$ ， $v_2$ ， $v_3$ 。它们的作者分别是： $v_1:A,B$   $v_2:B,C$   $v_3:C,D$

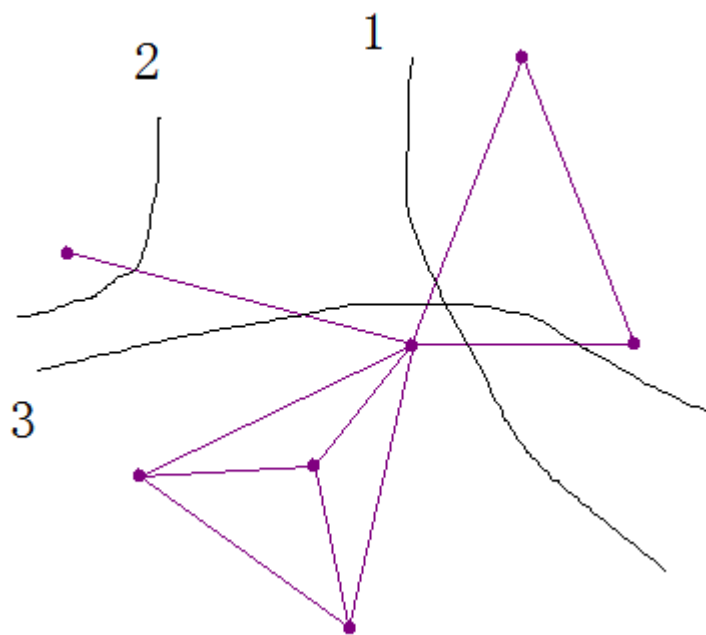


- 如果 $v_1:A,B$   $v_2:A,C$   $v_3:A,D$



# 简单图的分割

- **目标**：使分割出来的两个子图之间的关联最小



- **问题**：如何定义“关联最小”？

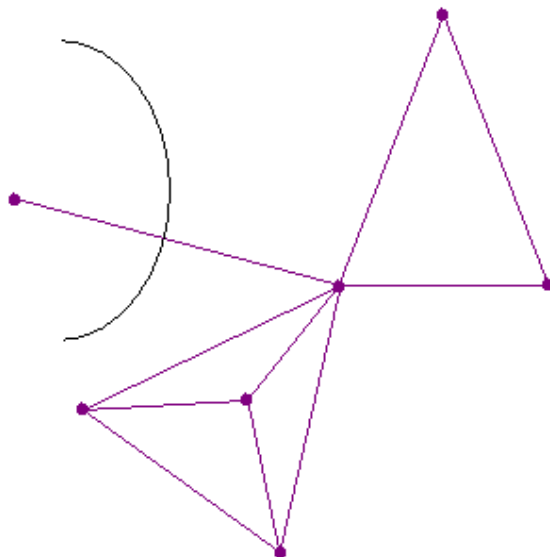
# 简单图分割的数学表达

$$cut(G^+, G^-) = \sum_{u \in G^+, v \in G^-} w(u, v)$$

- 分割子图间关联最小 = 跨分割边界的所有边的权值之和最小
- 邻接矩阵(Adjacency Matrix)
- $A(i, j)$  = 顶点 $i$ 和顶点 $j$ 之间的所有边的权值之和
- $\text{Min Cut}(G^+, G^-)$ , 根据二次型表达式
- 等价于:  $\text{Max}_Y Y^T A Y$ , 其中  $Y_i \in \{+1, -1\}$ ;

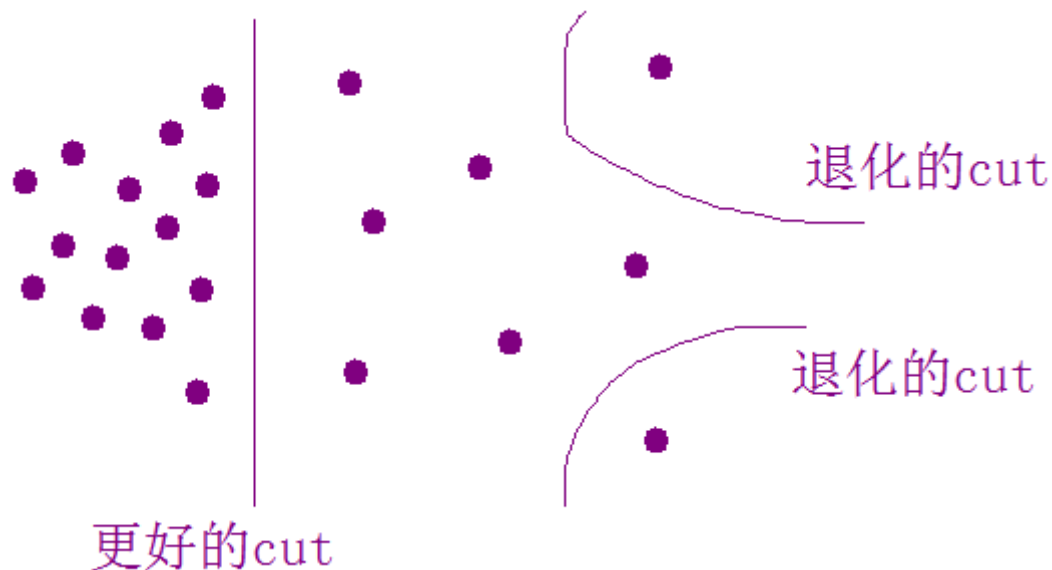
# 简单图分割的问题

- 问题：导致退化的分割



# Normalized-Cut

- 仅仅做到跨边界的权值和最小还不够，因为可能存在一些孤立点，它们跟外界的联系本身就极小，于是很可能被独立分割出来。



# Normalized-Cut

- **解决思想**：一个cut是“好的”当且仅当对任意一个子图来说，从子图中的节点出发跨越分割边界的边的权值和 相比于从子图节点出发的所有边的权值和的比例越小越好。通俗来说就是：任一分割出来的子图跟外界的联系主要来自该子图内部。

$$Ncut(G^+, G^-) = \frac{cut(G^+, G^-)}{asso(G^+, G)} + \frac{cut(G^+, G^-)}{asso(G^-, G)}$$

$$asso(G^+, G) = \sum_{u \in G^+, t \in G} w(u, t)$$

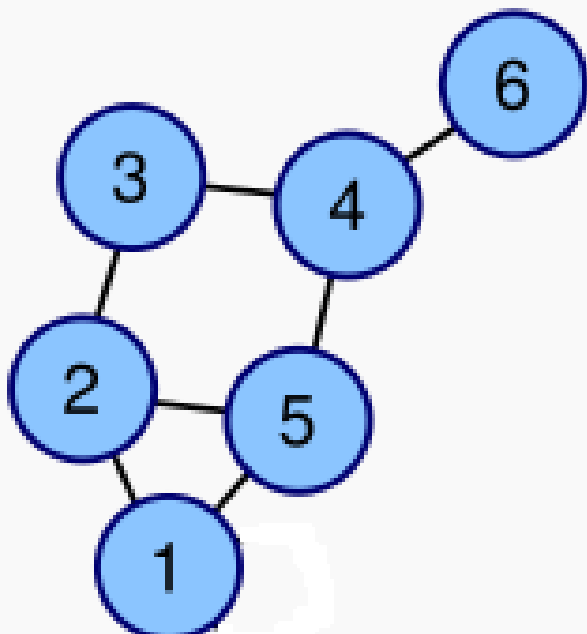


# Normalized-Cut

**NP-Hard**



# 拉普拉斯矩阵(Laplacian Matrix)

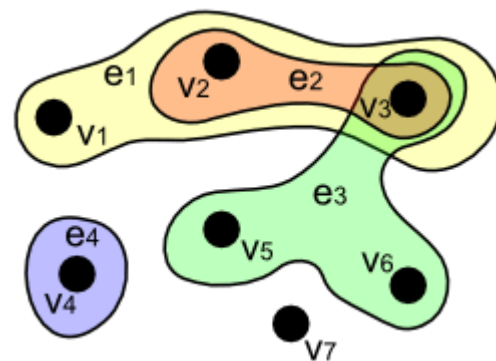
Labeled graph	Laplacian matrix
	$\begin{pmatrix} 2 & -1 & 0 & 0 & -1 & 0 \\ -1 & 3 & -1 & 0 & -1 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & -1 & 3 & -1 & -1 \\ -1 & -1 & 0 & -1 & 3 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 \end{pmatrix}$

# 谱(Spectrum)方法

- NP-Hard
- 谱方法逼近解
- $\min_z (Z^T L Z / Z^T Z)$  其中  $Z_i \in \{r_+, r_-\}$ ;
- $r_+ = \sqrt{|\{i: z_i < 0\}| / |\{i: z_i > 0\}|}$
- $r_- = \sqrt{|\{i: z_i > 0\}| / |\{i: z_i < 0\}|}$
- 不变式:  $Z^T Z = n$ ;  $Z^T \mathbf{1} = 0$ ;
- 含义:  $L$  是拉普拉斯矩阵  $L = B - A$

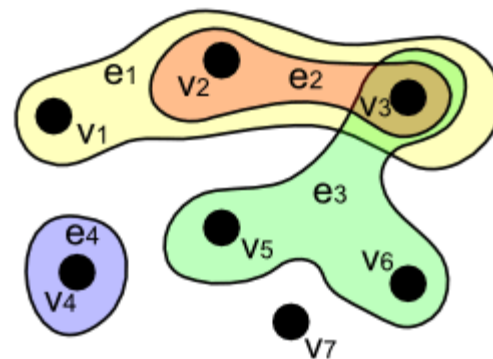
# 超图理论的目标

将简单图的表达泛化为超图表达，将简单图分割算法推广到超图分割之上，并证明超图分割和简单图分割的**内在标准(criteria)**是一致的

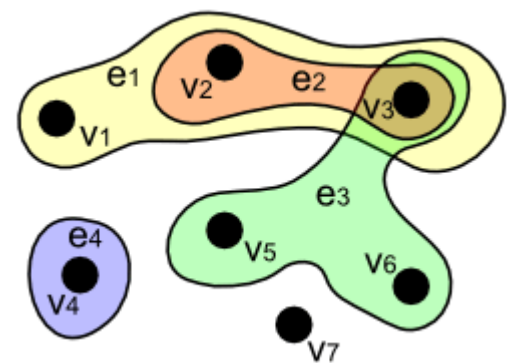


# 超图的表示

- 关键是超边如何表示： 用一个点集来表示。
- 令 $V$ 是一个顶点集合 $V=\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ ;
- 则每一条超边都是 $V$ 的一个子集
- $E = \{e_1, e_2, e_3, e_4\} = \{\{v_1, v_2, v_3\}, \{v_2, v_3\}, \{v_3, v_5, v_6\}, \{v_4\}\}$



# 超图的矩阵表达



$$H(v,e) = \begin{cases} 1; & v \in e \\ 0; & \text{otherwise} \end{cases}$$

$$G = (V, E, w)$$

顶点的度  $d(v)$

$$d(v) = \sum_{\{e \in E | v \in e\}} w(e)$$

超边的度

$$\delta(e) = |e|$$

超图的矩阵表达

$$H(v,e) = \begin{cases} 1; & v \in e \\ 0; & \text{otherwise} \end{cases}$$

	e1	e2	e3	e4
v1	1	0	0	0
v2	1	1	0	0
v3	1	1	1	0
v4	0	0	0	1
v5	0	0	1	0
v6	0	0	1	0
v7	0	0	0	0

# 超图的邻接矩阵

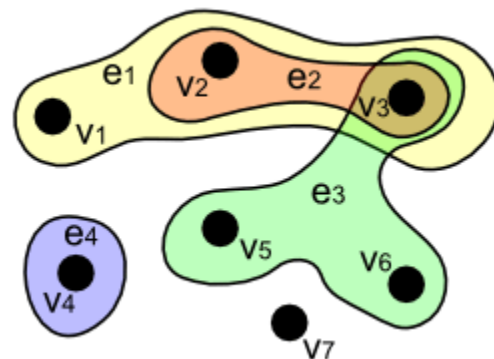
$$H(v,e)=\begin{cases} 1; v \in e \\ 0; otherwise \end{cases}$$

$$A=HWH^T-D_v$$

其中 $W$ 是一对角阵，对角线元素为各超边的权值。 $A$ 是超图的邻接矩阵

按右边方法表示的 $A$ (超图的邻接矩阵)， $A(i,i)$ 为0， $A(i,j)$ 为 $v_i$ 和 $v_j$ 共享的所有超边的权值和。

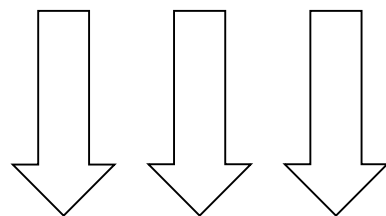
$D_v$ 为一对角阵，对角线元素为各顶点的度 $d(v)$ 。



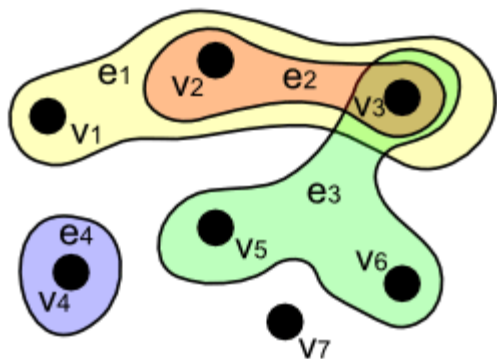
# 超图的分割(cut)

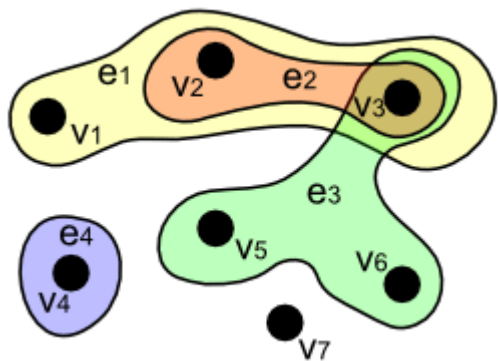
$$cut(G^+, G^-) = \sum_{u \in G^+, v \in G^-} w(u, v)$$

如何将简单  
图的分割标  
准推广到超  
图上面？



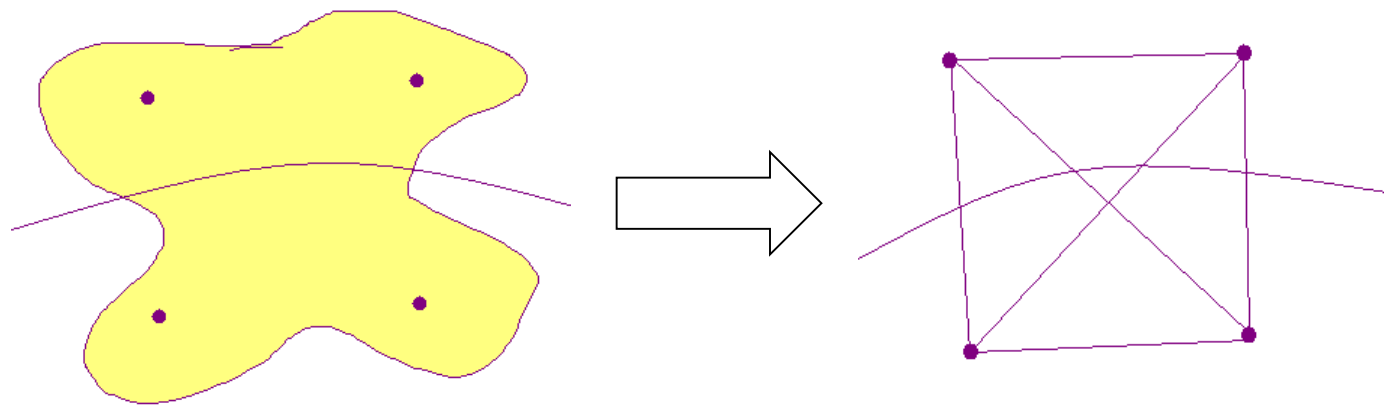
$$cut(G^+, G^-) = \sum_{e \in \partial G} w(e) \frac{|e \cap G^+| |e \cap G^-|}{\delta(e)}$$





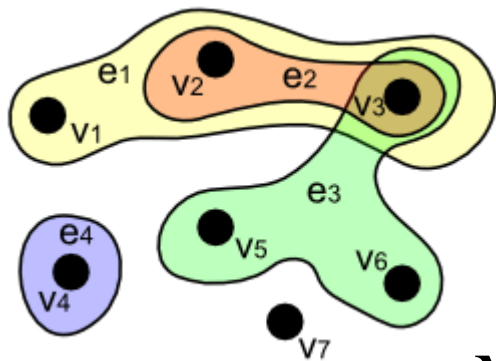
# 超图cut的含义

$$cut(G^+, G^-) = \sum_{e \in \mathcal{E}} w(e) \frac{|e \cap G^+| |e \cap G^-|}{\delta(e)}$$



将被切割的**每一条超边**看作一个子图，其中每两个顶点都是两两相连的，**连接的权值**皆为 **$w(e)/(e \text{ 的度})$** 。该子图被切割为 **$e \cap G^+$** 和 **$e \cap G^-$** 个顶点，因此被切断的边一共有 **$|e \cap G^+| |e \cap G^-|$** 个。

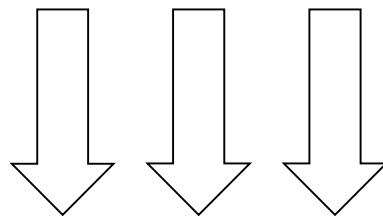




# 超图的Normalized-Cut

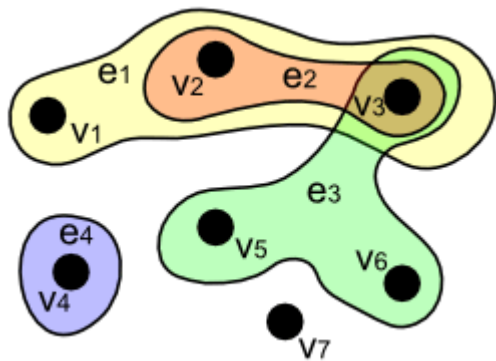
$$Ncut(G^+, G^-) = \frac{cut(G^+, G^-)}{asso(G^+, G)} + \frac{cut(G^+, G^-)}{asso(G^-, G)}$$

超图和简单图  
的  
Normalized-  
cut是形式一  
致的



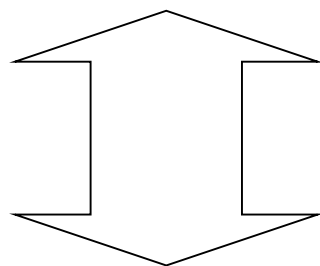
$$Ncut(G^+, G^-) = \frac{cut(G^+, G^-)}{asso(G^+, G)} + \frac{cut(G^+, G^-)}{asso(G^-, G)}$$

$$asso(G^+, G) = \sum_{v \in G^+} d(v)$$



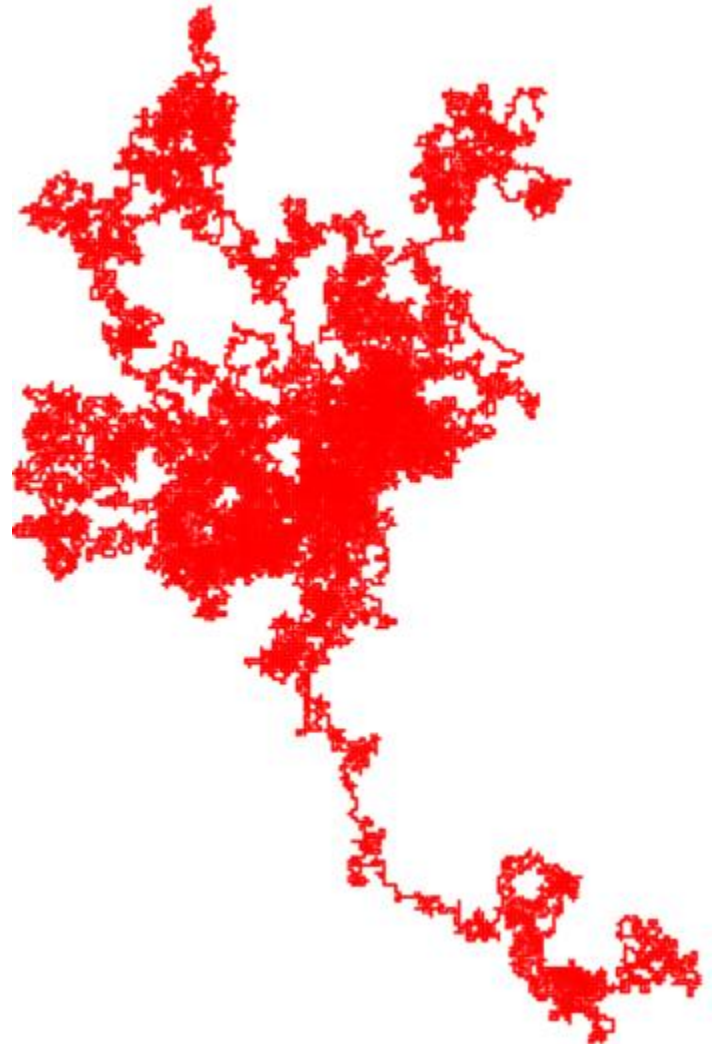
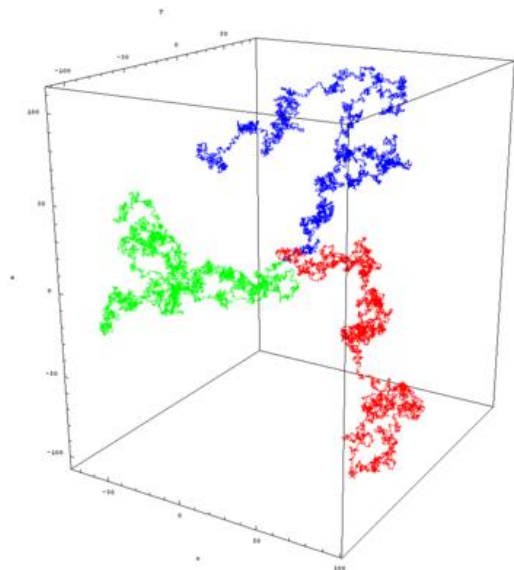
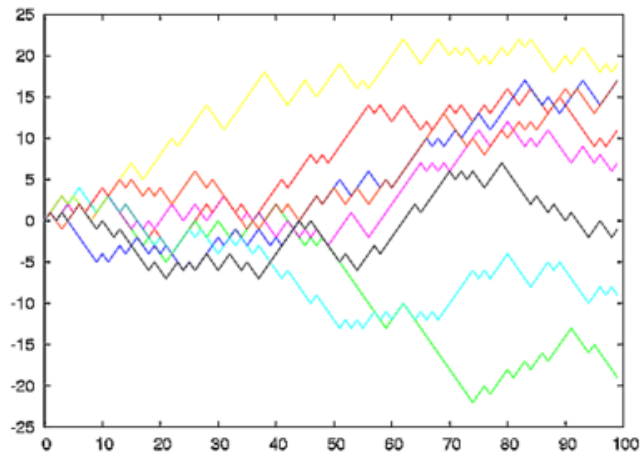
# 超图的Normalized-Cut

$$\operatorname{argmin}_{\emptyset \neq G^+ \subset G} c(G^+) := \operatorname{cut}(G^+, G^-) \left( \frac{1}{\operatorname{asso}(G^+, G)} + \frac{1}{\operatorname{asso}(G^-, G)} \right)$$



$$\operatorname{argmin}_{\emptyset \neq S \subset V} c(S) := \operatorname{vol} \partial S \left( \frac{1}{\operatorname{vol} S} + \frac{1}{\operatorname{vol} S^c} \right)$$

# 随机游走(Random Walk)



# 超图分割的随机游走解释

- **意义**：证明超图分割的确是简单图分割的一个妥善的推广，这对超图分割算法的有效性至关重要。
- **图分割的随机游走解释**：一个最优分割须使得随机游走落在同一个子图中的概率最大，同时随机游走跨越分割边界的几率最小。
- **目标**：证明超图分割也满足同样的随机游走性质。

# 什么是随机游走(Random Walk) Google Pagerank算法



# Google Pagerank算法



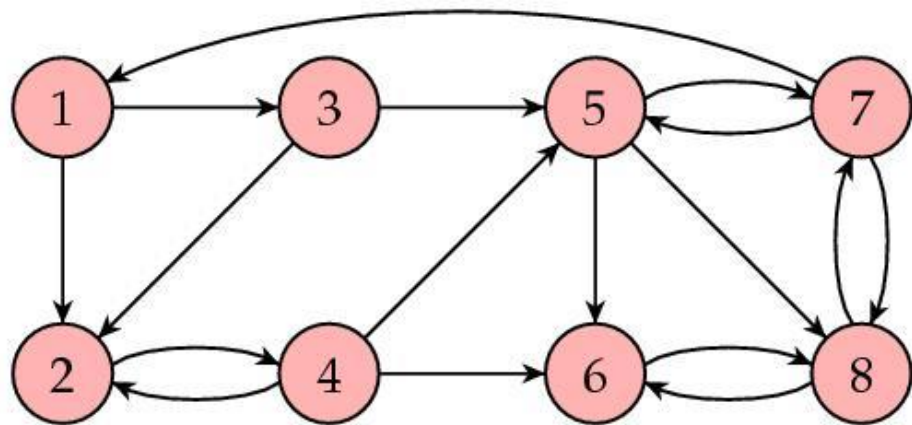
这么多页面，  
它们互相之间  
都有一堆链接，  
我怎么知道一个  
特定的页面的  
重要性是多少  
呢？

- 基本模型：用一个向量 $I$ 来代表所有页面的重要性， $I$ 的第 $i$ 个分量 $I_i$ 就是第 $i$ 个页面的重要性；另，假设一个页面有 $l_j$ 个向其它页面的链接，那么每个被指向的页面都得到该页面的 $1/l_j$ 的重要性；同时假设一个页面的重要性完全来自指向它的页面的贡献

- 数学表达：
$$I_i = \sum_{P_j \in B_i} \frac{I_j}{l_j}$$

- 其中 $P_j$ 表示第 $j$ 个页面。 $l_j$ 表示第 $j$ 个页面上的链接数， $P_j \in B_i$ 表示第 $j$ 个页面指向 $P_i$ 。

# Google PageRank 算法



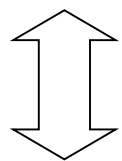
$$H_{ij} = \begin{cases} 1/l_j; & P_j \in B_i \\ 0; & \text{otherwise} \end{cases} \quad H = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1/3 & 0 \\ 1/2 & 0 & 1/2 & 1/3 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1/3 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 1 & 1/3 & 0 \end{bmatrix}$$

# Google Pagerank算法



- 如何计算  $I=HI$  中的  $I$ ? ( $I$  是  $H$  的一个特征向量, 对应特征值为 1)
- 迭代法:  $I^{k+1} = HI^k$

$$I_i = \sum_{j \in B_i} \frac{I_j}{l_j}$$



$$I = HI$$

$H =$

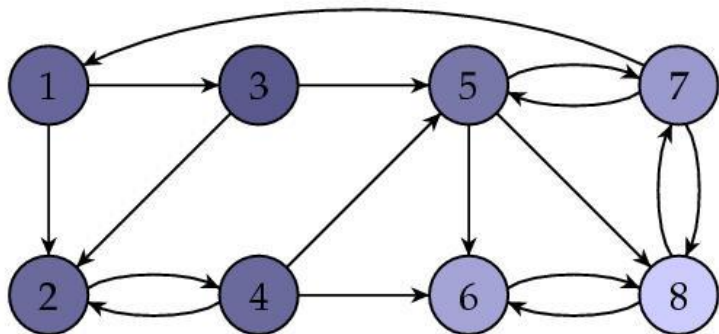
$$H = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1/3 & 0 \\ 1/2 & 0 & 1/2 & 1/3 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1/3 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 1 & 1/3 & 0 \end{bmatrix}$$

$I =$

$$I = \begin{bmatrix} 0.0600 \\ 0.0675 \\ 0.0300 \\ 0.0675 \\ 0.0975 \\ 0.2025 \\ 0.1800 \\ 0.2950 \end{bmatrix}$$



# Google PageRank算法



$$H = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1/3 & 0 \\ 1/2 & 0 & 1/2 & 1/3 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 1/3 & 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1/3 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 1 & 1/3 & 0 \end{bmatrix}$$

$$I = HI$$



$I^0$	$I^1$	$I^2$	$I^3$	$I^4$	...	$I^{60}$	$I^{61}$
1	0	0	0	0.0278	...	0.06	0.06
0	0.5	0.25	0.1667	0.0833	...	0.0675	0.0675
0	0.5	0	0	0	...	0.03	0.03
0	0	0.5	0.25	0.1667	...	0.0675	0.0675
0	0	0.25	0.1667	0.1111	...	0.0975	0.0975
0	0	0	0.25	0.1806	...	0.2025	0.2025
0	0	0	0.0833	0.0972	...	0.18	0.18
0	0	0	0.0833	0.3333	...	0.295	0.295

$$I = \begin{bmatrix} 0.0600 \\ 0.0675 \\ 0.0300 \\ 0.0675 \\ 0.0975 \\ 0.2025 \\ 0.1800 \\ 0.2950 \end{bmatrix}$$

# Google Pagerank算法



- 问题：链接黑洞(只进不出)



$$H = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$$

$I^0$	$I^1$	$I^2$	$I^3=I$
1	0	0	0
0	1	0	0

# Google Pagerank算法



- 解决：随机游走(Random Walk)理论
- 假设你是一个网络爬虫，在网络上跟着页面链接随机的游走。那么，当你发现自己停在一个页面 $P_j$ 上，而 $P_j$ 共有 $l_j$ 个链接，其中一个指向 $P_i$ ，那么你下一步游走到 $P_i$ 的几率就是 $1/l_j$ 。
- 在你随机游走的整个过程中，假设你停留在 $P_j$ 上的时间是 $T_j$ ，那么你停留在 $P_i$ 上的时间就是：

$$T_i = \sum_{P_j \in B_i} \frac{T_j}{l_j}$$

随机游走模型跟页面重要性模型是一致的

$$T_i = \sum_{P_j \in B_i} \frac{T_j}{l_j}$$

# Google Pagerank算法



- 随机游走到页面2(一个链接黑洞)的时候, 尽管没有链接, 但我们可以假设下一步游走等概率游走到任意一个其它页面, 即

$$S = \begin{bmatrix} 0 & 1/2 \\ 1 & 1/2 \end{bmatrix}$$

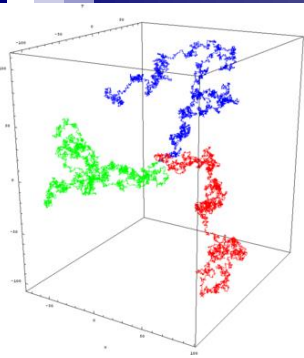


$$H = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$$

$I^0$	$I^1$	$I^2$	$I^3=I$
1	0	0	0
0	1	0	0

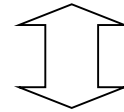
- 于是 
$$I = \begin{bmatrix} 1/3 \\ 2/3 \end{bmatrix}$$

# 超图分割de随机游走解释



$p(u,v)$ 表示从顶点 $u$ 随机游走到顶点 $v$ 的概率。

$$p(u,v) = \sum_{e \in E | u \in e, v \in e} \frac{w(e)}{\delta(e)} \frac{1}{d(u)}$$



$$p(u,v) = \sum_{e \in E} w(e) \frac{H(u,e)}{d(u)} \frac{H(v,e)}{\delta(e)}$$

$\pi(v)$ 表示随机游走停留在 $v$ 上的概率。

$$\pi(v) = \frac{d(v)}{\text{vol}V}$$

$$\begin{aligned} \sum_{u \in V} \pi(u) p(u,v) &= \sum_{u \in V} \frac{d(u)}{\text{vol}V} \sum_{e \in E} w(e) \frac{h(u,e)}{d(u)} \frac{h(v,e)}{\delta(e)} = \frac{1}{\text{vol}V} \sum_{u \in V} \sum_{e \in E} w(e) h(u,e) \frac{h(v,e)}{\delta(e)} \\ &= \frac{1}{\text{vol}V} \sum_{e \in E} w(e) h(v,e) \sum_{u \in V} \frac{h(u,e)}{\delta(e)} = \frac{1}{\text{vol}V} \sum_{e \in E} w(e) h(v,e) = \frac{d(v)}{\text{vol}V} = \pi(v) \end{aligned}$$

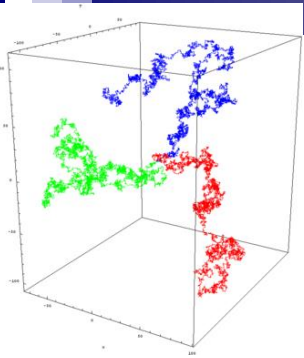
# 超图分割de随机游走解释

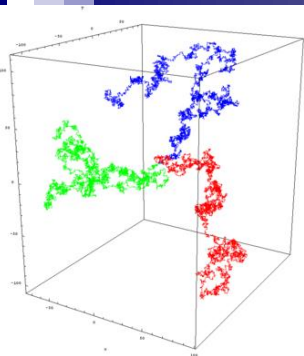
$$\operatorname{argmin}_{\emptyset \neq S \subset V} c(S) := \operatorname{vol} \partial S \left( \frac{1}{\operatorname{vol} S} + \frac{1}{\operatorname{vol} S^c} \right)$$

$$c(S) = \frac{\operatorname{vol} \partial S}{\operatorname{vol} V} \left( \frac{1}{\operatorname{vol} S / \operatorname{vol} V} + \frac{1}{\operatorname{vol} S^c / \operatorname{vol} V} \right)$$

$$\operatorname{vol} S / \operatorname{vol} V = \sum_{v \in S} \frac{d(v)}{\operatorname{vol} V} = \sum_{v \in S} \pi(v)$$

$$\begin{aligned} \frac{\operatorname{vol} \partial S}{\operatorname{vol} V} &= \sum_{e \in \partial S} \frac{w(e)}{\operatorname{vol} V} \frac{|e \cap S| |e \cap S^c|}{\delta(e)} = \sum_{e \in \partial S} \sum_{u \in e \cap S} \sum_{v \in e \cap S^c} \frac{w(e)}{\operatorname{vol} V} \frac{h(u, e) h(v, e)}{\delta(e)} \\ &= \sum_{e \in \partial S} \sum_{u \in e \cap S} \sum_{v \in e \cap S^c} w(e) \frac{d(u)}{\operatorname{vol} V} \frac{h(u, e)}{d(u)} \frac{h(v, e)}{\delta(e)} \\ &= \sum_{u \in S} \sum_{v \in S^c} \frac{d(u)}{\operatorname{vol} V} \sum_{e \in S} w(e) \frac{h(u, e)}{d(u)} \frac{h(v, e)}{\delta(e)} = \sum_{u \in S} \sum_{v \in S^c} \pi(u) p(u, v) \end{aligned}$$





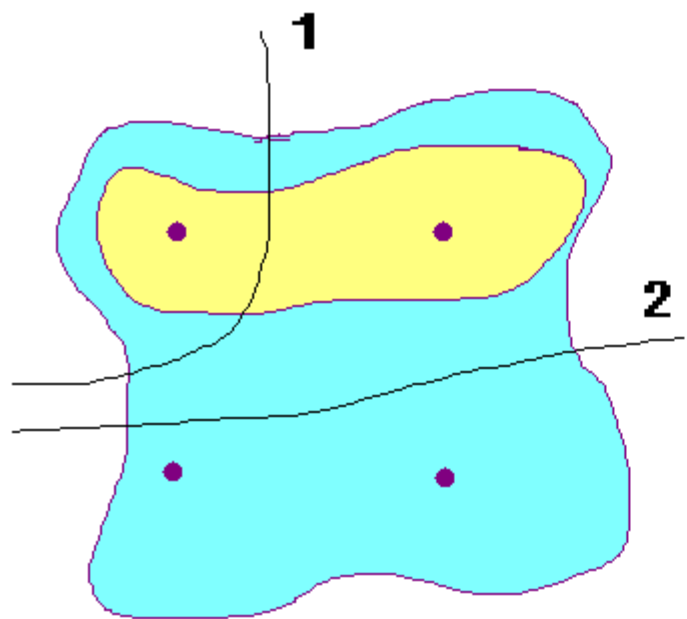
# 超图分割的随机游走解释

$$c(S) = \frac{\text{vol} \partial S}{\text{vol} V} \left( \frac{1}{\text{vol} S / \text{vol} V} + \frac{1}{\text{vol} S^c / \text{vol} V} \right)$$

$$c(S) = \sum_{u \in S} \sum_{v \in S^c} \pi(u) p(u, v) \left( \frac{1}{\sum_{v \in S} \pi(v)} + \frac{1}{\sum_{v \in S^c} \pi(v)} \right)$$

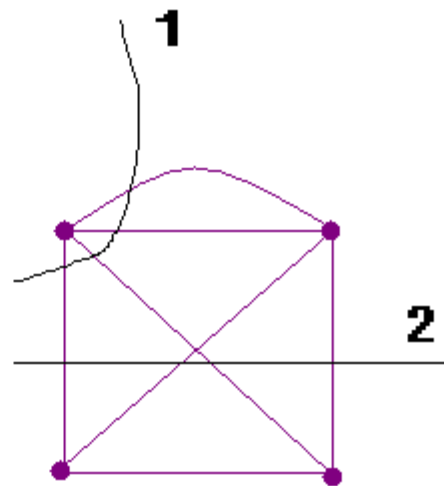
随机游走留在分割子图内的几率尽可能大，跨越分割边界的几率尽可能小

# 超图真的比简单图优越吗？



$$\text{vol}(\text{ds1}) = \frac{1*3}{4} + \frac{1*1}{2} = 5/4$$

$$\text{vol}(\text{ds2}) = \frac{2*2}{4} = 4/4$$



$$\text{vol}(\text{ds1}) = \text{vol}(\text{ds2}) = 4$$



# 如何将超图运用在共指消解中

- 跟把简单图运用在共指消解中一样，因为超图是简单图的推广。
- 半指导聚类：惩罚矩阵
- 问题讨论：在共指消解里面，超图的超边表示什么，超边的权值又如何确定(这篇论文里面也是一个未决问题)。