Kruskal算法

设计思想

输入：图$G=(V,E,W)$, $V=\{1,2,\ldots,n\}$

输出：G的最小生成树 T

设计思想：

(1) 按照长度从小到大对边排序.

(2) 依次考察当前最短边 e，如果 e与 T 的边不构成回路，则把 e 加入树 T，否则跳过 e. 直到选择了n-1条边为止.

伪码

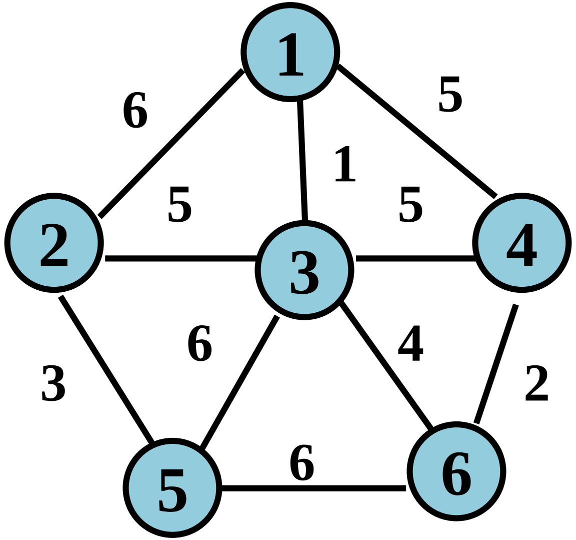
算法 Kruskal

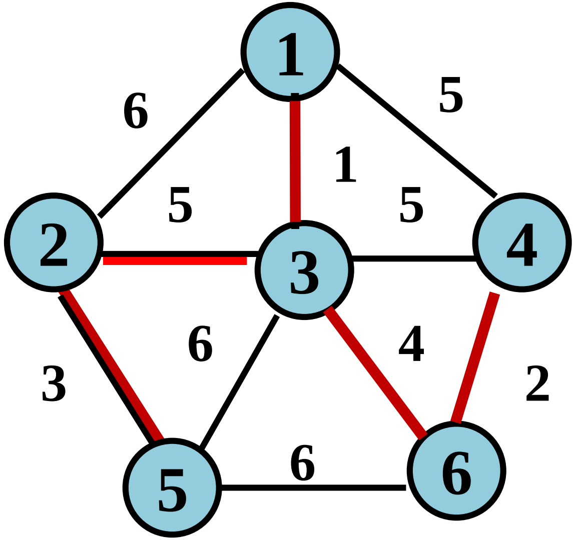
输入：连通图G　　// 顶点数n，边数m

输出：G 的最小生成树

1. 权从小到大排序E的边, $E=\{e_1,e_2,\ldots,e_m\}$
2. $T \leftarrow \varnothing$
3. **repeat**
4. $\quad e \leftarrow E$中的最短边
5. $\quad$**if** e 的两端点不在同一连通分支
6. $\quad$**then** $T \leftarrow T \cup \{e\}$
7. $\quad E \leftarrow E - \{e\}$
8. **until** T 包含了$n-1$条边

实例





正确性证明思路

命题：对于任意 n, 算法对 n 阶图找到一棵最小生成树.

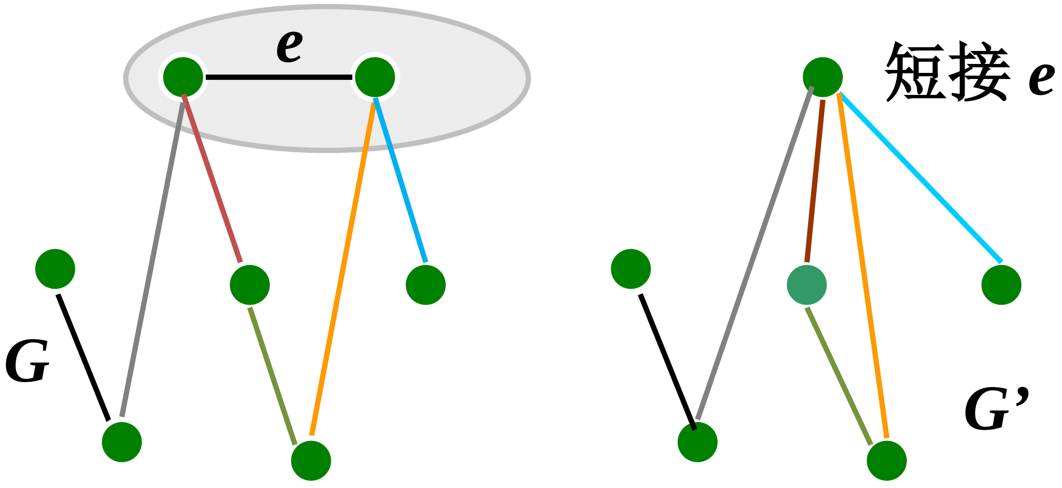
证明思路：

归纳基础　证明：$n = 2$, 算法正确.
G只有一条边，最小生成树就是 G.

归纳步骤　证明：假设算法对于n 阶图是正确的，其中$n>1$，则对于任何$n+1$阶图算法也得到一棵最小生成树.

短接操作

任给 n+1个顶点的图 G，G中最小权边 e = {i, j}，从G 中短接i 和 j, 得到图 G'.



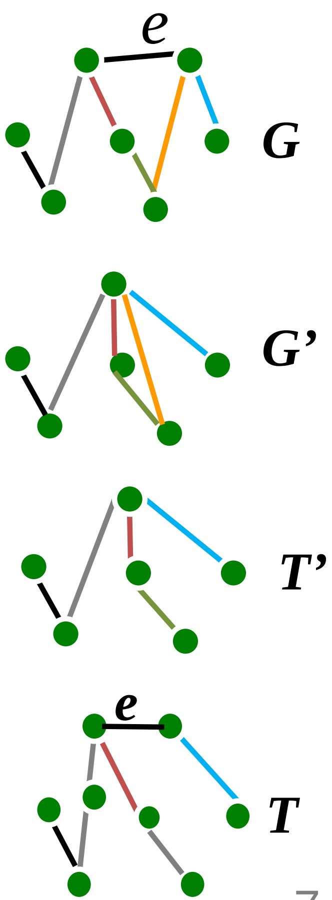
归纳步骤证明

对于任意 n+1阶图G短接最短边e，得到 n 阶图 G'

根据归纳假设算法得到G'的最小生成树 T'

将被短接的边 e “拉伸”回到原来长度，得到树 T

证明 T 是 G 的最小生成树



T是G的最小生成树

$T=T'\cup\{e\}$是关于G 的最小生成树.

否则存在G 的含边e 的最小生成树 $T^*, W(T^*)<W(T)$. (如果 $e\notin T^*$, 在T^* 中加边e，形成回路. 去掉回路中任意别的边所得生成树的权仍旧最小).

在T^*短接e得到G' 的生成树$T^*-\{e\}$,

$$\boxed{\begin{aligned} &W(T^*-\{e\}) = W(T^*) - w(e) \\ &< W(T) - w(e) = W(T') \end{aligned}}$$

与T' 的最优性矛盾.

算法实现与时间复杂度

建立FIND数组，**FIND[*i*]** 是结点 ***i*** 的连通分支标记.

(1) 初始 **FIND[*i*] = *i*.**

(2) 连通分支合并，较小分支标记更新为较大分支标记

每个结点至多更新 **log*n***次，
建立和更新 **FIND**数组: $O(n \log n)$

时间：$O(m \log m)+O(n\log n)+O(m)$
$= O(m \log n)$

应用：数据分组问题

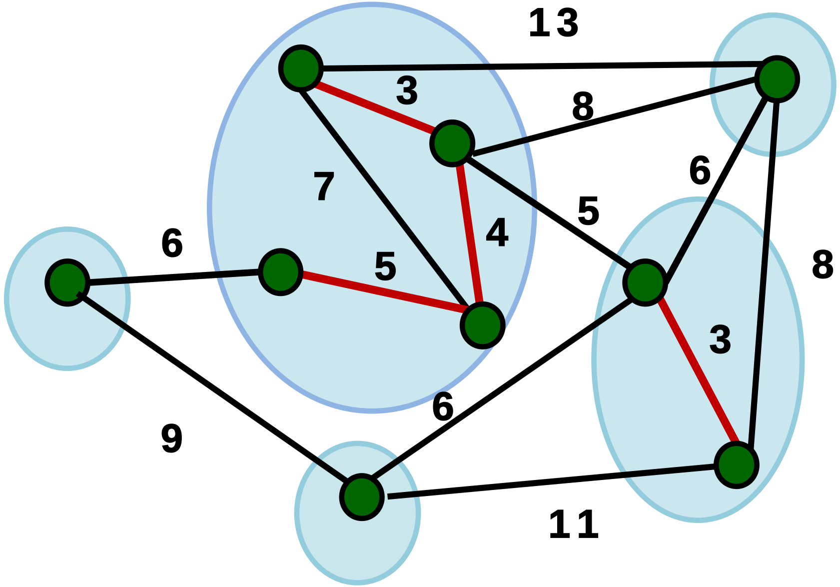
一组数据(照片, 文件, 生物标本)要把它们按照相关性进行分类.

用相似度函数或“距离”来描述个体之间的差距.

如果分成5类，使得每类内部的个体尽可能相近，不同类之间的个体尽可能地“远离”．如何划分?

10

应用：数据分组问题



单链聚类

类似于**Kruskal**算法

(1) 按照边长从小到大对边排序

(2) 依次考察当前最短边 ***e***， 如果 ***e*** 与 已经选中的边不构成回路，则把 ***e*** 加入集合，否则跳过 ***e***. 计数图的连通分支个数.

(3) 直到保留了k个连通分支为止.

小结

- **Kruskal算法的贪心策略：在不构成回路条件下选当前最短边**
- 正确性证明：对规模归纳
- 时间复杂度：$O(m\log n)$
- 应用：单链聚类