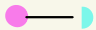
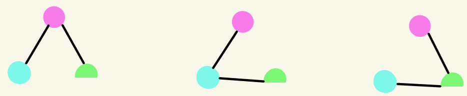
5. Cayley's Formula

Counting (labeled) trees: how many different trees can be formed from n distinct vertices? $\{V_1, V_2, \cdots, V_n\}$

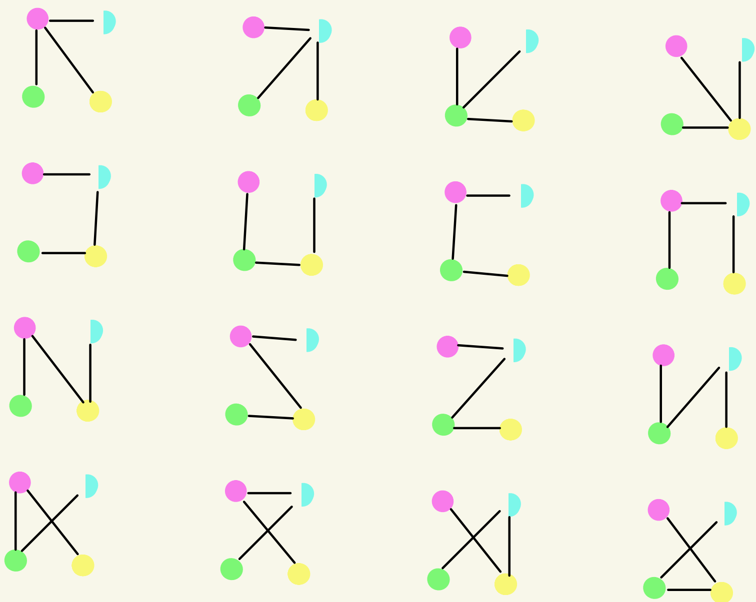
n=2



n=3



n=4



Cayley's formula: there are n^{n-2} trees on n distinct vertices

三个证明: ① Prüfer Code

② Double Counting

③ Kirchhoff's Matrix-Tree Theorem.

Proof of THE BOOK

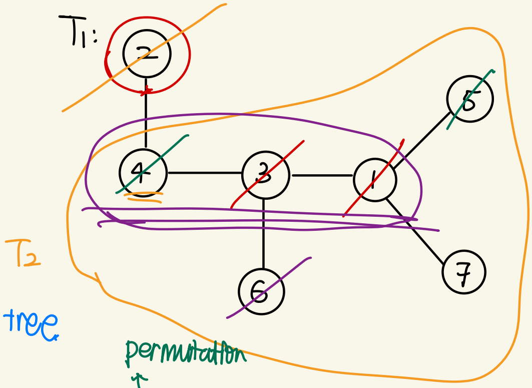
Prüfer Code

tree ⟷ tuple

→ bijection 双射 $f: S \to T$ $\quad |S| = |T|$

leaf: vertex of degree 1

⇒ removing a leaf from T stills gives a tree



编码(encode)

$T_1 = T$

for $i = 1$ to $n-1$:

$\quad u_i$: smallest leaf in T_i

$\quad (u_i, v_i)$: edge in T_i

$\quad T_{i+1}$ = delete u_i from T_i

→ Prüfer Code: $(v_1, v_2, \cdots, v_{n-2})$

permutation

叶 u_i:	2	4	5	6	3	1
叶邻 v_i:	4	3	1	3	1	7

n−2个点

n 一定保留

tuple: $(4, 3, 1, 3, 1)$

双射 {
- injection 单射 (1-1): 给定一个 Prüfer code, 都唯一对应一个树 ⇒ problem-dependent
- surjection 满射 (onto): 任给一个 Prüfer code 都唯一对应一个树 ⇒ problem-independent

树 → 无cycle

解码(decode)

u_i:	~~2~~	4	5	6	3	1
v_i:	~~4~~	3	1	3	1	7

n−2个数

Claim: $v_{n-1} = n$

Proof: u_i is smallest leaf in T_i; a tree has ≥ 2 leaves ⇒ n is never deleted.

通过 $(v_1, v_2, \cdots, v_{n-2})$ 恢复出每一个 u_i ⇒ $\boxed{\{(u_i, v_i)\}_{i=1}^{n-1}}$ → 树T的边集

v_i中的点一定不是T的叶子结点

⇒ 叶子结点不在 v_i 中

叶子结点

⇒ 第一步: 找到最小的不在 V_i 中的结点 (2)

Proposition: u_i is the smallest number NOT in $\{u_1, \cdots, u_{i-1}\} \cup \{v_i, \cdots, v_{n-1}\}$

已经删过点集 非叶子结点 (T_i)

Proof: (以证明第一个点为例)

# of occurrences of v in $u_1, \cdots, u_{n-1}, v_{n-1}$: 1.

# of occurrences of v in edges (u_i, v_i), $i = 1, \cdots, n-1$: $\deg_T(v)$

⇒ # of occurrences of v in Prüfer code $(v_1, \cdots, v_{n-2})$: $\deg_T(v) - 1 > 0$

⇒ $\deg_T(v) > 1$

⇒ v is NOT a leaf □

Decoding:

T = empty graph

$v_{n-1} = n$

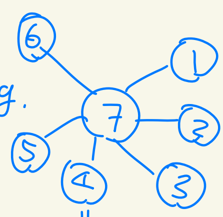
for $i = 1$ to $n-1$:

  u_i: smallest number NOT in $\{u_1, \cdots, u_{i-1}\} \cup \{v_i, v_{i+1}, \cdots, v_{n-1}\}$

  add edge (u_i, v_i) to T.

⇒ Prüfer Code is reversible

e.g.



⇓

(7, 7, 7, 7, 7)

树 ⇔ Code $\in [n]^{n-2}$ $(v_1, v_2, \cdots, v_{n-2})$

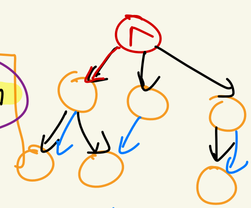
By Bijection rule, Cayley's formula: n^{n-2}

Double Counting 双重计数

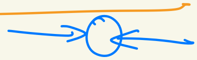
↳ 最关键: What do you want to doubly count?

Goal:

# of sequences of adding directed edges to an empty graph to form a rooted tree



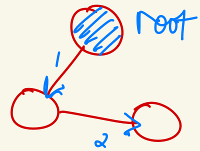
按照某种顺序添加有向边，构成一个有根树



: illegal

T_n : # of tree on n distint vertices

① from a tree : T_n
- pick a root $\quad n$
- pick an order of edges $\quad (n-1)!$



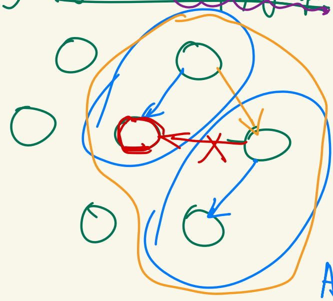
第n笔画了哪条边

$\Rightarrow T_n \cdot n \cdot (n-1)! = n! \cdot T_n$

② from an empty graph : add edges one by one.

Start from n ~~isolated vertices~~ rooted trees : 将每个点看成一个树根

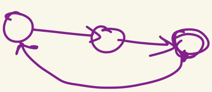
each step joins 2 trees



After adding k edges : $(n-k)$ rooted trees

避免:

终点



环!

起点: 所有点均可 $\quad n$

终点: 只能是树根 $\quad n-k-1$

非起点所在树的

$\Downarrow$

$$\prod_{k=0}^{n-2} n(n-k-1) = n^{n-1} \cdot (n-1)(n-2)\cdots 1 = n^{n-1} \cdot (n-1)! = n^{n-2} \cdot n!$$

$$\Rightarrow n^{n-2} \cdot n! = n! \cdot T_n \Rightarrow T_n = n^{n-2}$$; Cayley's formula.

Matrix-Tree Theorem

矩阵树定理 $\qquad$ from spectral graph theory 谱图论

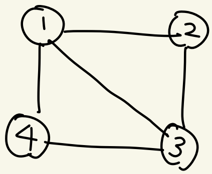
$G=(V,E)$

adjacency matrix A: $A(i,j) = \begin{cases} 1, & \text{if } (i,j) \in E \\ 0, & \text{if } (i,j) \notin E \end{cases}$

diagonal matrix D: $D(i,j) = \begin{cases} \deg(i) & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases}$

$\Rightarrow$ graph Laplacian: $\boxed{L = D - A}$

e.g.



$$L = \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ -1 & 0 & -1 & 2 \end{bmatrix} \end{array} \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array}$$

$$L(i,j) = \begin{cases} \deg(i) & \text{if } i=j \\ -1 & \text{if } i\neq j,\ (i,j)\in E \\ 0 & \text{if } i\neq j,\ (i,j)\notin E \end{cases}$$

$\downarrow$

quadratic form: $xLx^T = \sum_i d_i x_i^2 - \sum_{(i,j)\in E} x_i x_j = \frac{1}{2}\sum_{(i,j)\in E}(x_i - x_j)^2$

x: row vector. $x \in \mathbb{R}^n$

邻接矩阵

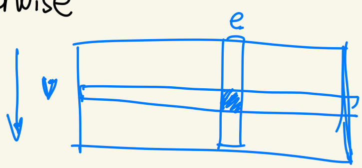
Incidence Matrix B: $n \times m$

n: vertex, m: edge

$\forall i \in V,\ j \in E$: $B(i,e) = \begin{cases} 1, & e=(i,j),\ i<j \\ -1, & e=(i,j),\ i>j \\ 0, & \text{otherwise} \end{cases}$

$\Downarrow$

$$\boxed{L = BB^T}$$



$$(BB^T)_{ij} = \sum_{k=1}^{m} b_{ik} b_{jk} = \begin{cases} \deg(i) & i=j \\ -1 & i\neq j,\ (i,j)\in E \\ 0 & i\neq j,\ (i,j)\notin E \end{cases}$$

剩余矩阵 $L_{i,i}$: submatrix of L by removing the ith row and ith column of L

$t(G)$: # of spanning trees in G

graph $\leftrightarrow$ spectral property of Matrix

Kirchhoff's Matrix-Tree Theorem: $\forall i$: $\boxed{t(G) = \det(L_{i,i})}$

Cayley's formula: # of trees on n distinct vertices n^{n-2}

$\hookrightarrow G = K_n$ n个顶点的完全图的生成树

$$L = \begin{pmatrix} n-1 & -1 & \cdots & -1 \\ -1 & n-1 & \cdots & -1 \\ \vdots & \vdots & & \vdots \\ -1 & -1 & \cdots & n-1 \end{pmatrix} \in \mathbb{R}^{n\times n} \Rightarrow L_{i,i} = \begin{pmatrix} n-1 & -1 & \cdots & -1 \\ -1 & n-1 & \cdots & -1 \\ \vdots & \vdots & & \vdots \\ -1 & -1 & \cdots & n-1 \end{pmatrix} \in \mathbb{R}^{(n-1)\times(n-1)}$$

$$\det(L_{i,i}) = n^{n-2} \Rightarrow T_n = t(K_n) = \det(L_{i,i}) = n^{n-2}$$

Proof of Kirchhoff's Matrix-Tree Theorem:

$L = BB^T \qquad L \in \mathbb{R}^{n\times n},\ B \in \mathbb{R}^{n\times m}$

$L_{i,i} = B_i B_i^T \qquad L_i \in \mathbb{R}^{(n-1)\times(n-1)},\ B_i \in \mathbb{R}^{(n-1)\times m}$

在B中删去第i行

$\det(L_{i,i}) = \det(B_i B_i^T)$

$\det(AB) = \det(A)\cdot\det(B)$

$\hookrightarrow A, B \in \mathbb{R}^{n\times n}$

$\det(A\cdot B) \qquad A \in \mathbb{R}^{n\times m},\ B \in \mathbb{R}^{m\times n}$

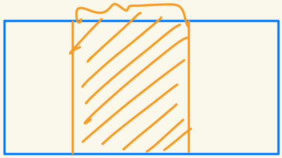
Cauchy-Binet Theorem: $\det(A\cdot B) = \sum_{S\in\binom{[m]}{n}} \det(A_{[n],S})\cdot\det(B_{S,[n]})$

穷举所有的方子矩阵

机械降神

$S: |S| = n$

$A: n\times m$

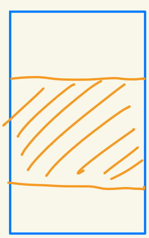


$A_{[n],S}$

m列

$B: m\times n$

S



$B_{S,[n]}$

$$\det(L_{i,i}) = \det(B_i B_i^T)$$

$$= \sum_{S\in\binom{[m]}{n-1}} \det\left(B_{[n]\setminus\{i\},S}\right)\cdot \det\left(B^T_{S,[n]\setminus\{i\}}\right)$$

$$= \sum_{S\in\binom{[m]}{n-1}} \det\left(B_{\underbrace{[n]\setminus\{i\}}_{j},\underbrace{S}_{e}}\right)^2 = \#\text{ of spanning trees of } G = t(G)$$

□

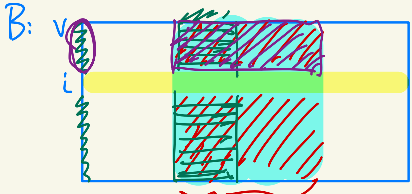
$\forall\ j\in[n]\setminus\{i\},\ e\in S$:

$$B_{[n]\setminus\{i\},S}(j,e) = \begin{cases} 1, & e=(j,k),\ j<k \\ -1, & e=(j,k),\ j>k \\ 0, & \text{otherwise} \end{cases}$$

Claim: $\det\left(B_{[n]\setminus\{i\},S}\right) = \begin{cases} \pm 1, & \text{if } S \text{ is a spanning tree of } G \\ 0, & \text{otherwise} \end{cases}$

(n vertices)

Proof:



$S\in\binom{[m]}{n-1}$

$B' = B_{[n]\setminus\{i\},S} \in \mathbb{R}^{(n-1)\times(n-1)}$

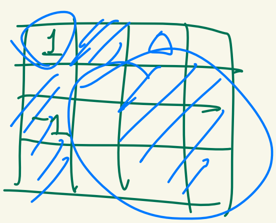
every column of B' contains at most one 1. at most one -1 and all other entries are 0.

$\Rightarrow \det(B') \in \{-1,0,1\}$

It suffices to show:

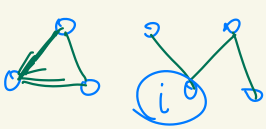
$\det(B') \neq 0$ iff S is a spanning tree of G.

(用数学归纳法)



① S 不是生成树

↳ $n-1$ 条边 $\Rightarrow$ 至少有两个连通分量



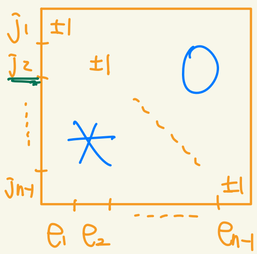
$\Downarrow$

一定存在某个连通分量 R s.t. $i\notin R$

⇒ B' 在该连通分量中的行和应为0

⇒ 这些行是 linearly dependent! ⇒ $\boxed{\det(B') = 0}$

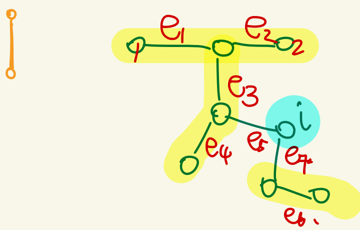
② S 是生成树;



∃ a leaf $j_1 \neq i$ with incident edge e_1 ⇒ delete e_1 — delete j_1

∃ a leaf $j_2 \neq i$ with incident edge e_2 ⇒ delete e_2 — delete j_2

⋮



vertices: $j_1, j_2, \cdots, j_{n-1}$

edges: $e_1, e_2, \cdots, e_{n-1}$

⇒ 构建一个上三角矩阵

⇒ $\det(B') = \pm 1$

$$\Rightarrow \det(B') = \begin{cases} \pm 1, & S \text{ 是生成树} \\ 0, & S \text{ 不是生成树} \end{cases}$$

□

⇒ $t(G) = \det(L_{i,i})$: Kirchhoff's Matrix-Tree Theorem.

① Prüfer Code

② Double Counting → Proofs of THE BOOK

③ Matrix-Tree Thm → Spanning tree

Spanning forest ?

①③: algorithmic implication

$P \neq NP$

probably approximately correct — PAC ⇒ #P — Leslie Valiant

$1-\delta$ $\quad |\cdot| \leq \varepsilon$