Lecture notes

We count graphs with a labeled set of vertices, usually, \([n]\).

**Ex.** There are \(2^3 = 8\) distinct graphs on \([3]\), and 3 of these graphs are trees.

Here is a slight extension of the famous Cayley’s Formula (proved by Borchardt in 1860).

**Theorem 1** (Th. 6.1.18 in the book). For all \(1 \leq k \leq n\), the number \(b_{n,k}\) of forests of rooted trees with vertex set \([n]\) that have \(k\) components and a given set of \(k\) roots is \(kn^{n-k-1}\). In particular, there are \(n^{n-2}\) trees with vertex set \(n\).

**Proof.** Induction on \(n\). If \(n = 1\) or \(n = k\), then \(b_{n,k} = 1\).

Suppose \(n > k \geq 1\) and the theorem holds for all smaller \(n' \geq k'\). Consider an \(n\)-vertex \(k\)-component forest \(F\) with the set \(K\) of \(k\) roots and the set \(R\) of the neighbors of these roots. By deleting \(K\) from \(F\), we get an \((n-k)\)-vertex \(r\)-component forest \(F'\) with with the set \(R\) of \(r\) roots. By definition, the number of such forests with the set of roots \(R\) is \(b_{n-k,r}\). Each such \(F'\) can be extended to an \(n\)-vertex \(k\)-component forest \(F\) with the set \(K\) of roots in \(k\) ways. So by induction,

\[
b_{n,k} = \sum_{r=1}^{n-k} \binom{n-k}{r} k^r b_{n-k,r} = \sum_{r=1}^{n-k} \binom{n-k}{r} k^r (n-k)^{n-k-r-1}
\]

\[
= k \sum_{r=1}^{n-k} \binom{n-k-1}{r-1} k^{r-1} (n-k)^{n-k-1-(r-1)} = k(k+n-k)^{n-k-1}. \quad \Box
\]

Among ways to code a graph are adjacency and incidence matrices. For labeled trees, there are nicer and shorter ways to code. Consider the following procedure for a tree \(T\) with vertex set \(\{1, \ldots, n\}\):

**Prüfer algorithm.** Let \(T_0 = T\). For \(i = 1, \ldots, n-1\),
(a) let \(b_i\) be the smallest leaf in \(T_{i-1}\),
(b) denote by \(a_i\) the neighbor of \(b_i\) in \(T_{i-1}\), and
(c) let \(T_i = T_{i-1} - b_i\).

The **Prüfer code** of \(T\) is the vector \((a_1, \ldots, a_{n-2})\).

**EXAMPLE.**

**Properties of Prüfer algorithm**

(P1) \(a_{n-1} = n\).

(P2) Any vertex of degree \(s\) in \(T\) appears in \((a_1, \ldots, a_{n-2})\) exactly \(s - 1\) times.

(P3) \(b_i = \min \{k : k \notin \{b_1, \ldots, b_{i-1}\} \cup \{a_i, a_{i+1}, \ldots, a_{n-2}\}\}\) for each \(i\).

**Proofs.** (P1) follows from the fact that we always have a leaf distinct from \(n\).

(P2) follows from the facts that at the moment some \(k\) appears in \((a_1, \ldots, a_{n-2})\), its degree decreases by 1 and for \(s \geq 3\) the neighbors of leaves in \(s\)-vertex trees are not leaves.

(P3) follows from the algorithm and (P2). \(\Box\)

**Theorem 2** (Prüfer, 1918). Every vector \((a_1, \ldots, a_{n-2})\) with \(a_i \in \{1, \ldots, n\}\) for each \(i\) is the Prüfer code of exactly one labeled \(n\)-vertex tree.
Lemma 5 (Binet-Cauchy Formula). Let \( A = (a_{ij})_{i,j=1}^n \) be an \( n \times n \) matrix, \( B = (b_{ji}) \) be an \( m \times m \) matrix, \( C = AB \). For \( S \subset [m] \) with \( |S| = n \), let \( A_S \) (respectively, \( B_S \)) denote the \( n \times n \) submatrix of \( A \) (respectively, of \( B \)) formed by the columns (respectively, rows) indexed by \( S \). Then
\[
\det A = \sum_{S \subset [m]: |S| = n} \det A_S \det B_S.
\]

This is a HOMEWORK PROBLEM.

Proof of Matrix Tree Theorem. (1) Let \( D \) be any orientation of \( G \) and \( M \) be its incidence matrix. Then \( Q = MM^T \).

(2) Let \( B \) be any \( (n-1) \times (n-1) \) submatrix of \( M \). Then \( \det B = 0 \) if the corresponding \( n-1 \) edges in \( G \) form a subgraph with a cycle. Otherwise, \( \det B \in \{-1, 1\} \).

Let \( M^* \) be obtained from \( M \) by deleting row \( n \). Then \( Q^* = M^*(M^*)^T \).

(3) Calculate \( \det Q^* \) by Lemma 5: every term is 0 or 1, and 1 if the edges in \( S \) form a tree.
A branching or out-tree is an orientation of a tree that directs all edges from a given vertex (a root).

An arborescence is a digraph whose every component is a branching. An in-tree is a reversed branching.

For a digraph $G$ with incidence matrix $A$, let $D^+$ (resp. $D^-$) be the diagonal matrix of in-degrees (resp. out-degrees), $Q^+ = D^+ - A^T$ and $Q^- = D^- - A^T$.

Examples.

Theorem 6 (Directed Matrix Tree Theorem, Tutte, 1948, Th. 6.1.28 in the book). The number of spanning out-trees (in-trees) of $G$ rooted at $v_i$ is the value of the cofactor for any entry in $i$th row of $Q^-$ ($i$th column of $Q^+$).

Examples.

Instead of Theorem 6, we will prove a much more general theorem:

Theorem 7 (Matrix Arborescence Theorem, Chaiken–Kleitman, 1978, Th. 6.1.30 in the book). For real $a_{ij}$, variables $x_1, \ldots, x_n$ and an arborescence $A$ on $\{v_1, \ldots, v_n\}$, let $w_A = \prod_{v_iv_j \in E(A)} a_{ij}x_j$. For $S \subseteq [n]$, let $T(S)$ be the set of all arborescences on $\{v_1, \ldots, v_n\}$ whose set of roots is $\{v_i : i \in S\}$. Define $Q = (q_{ij})_{i,j=1}^n$ as follows:

$$q_{ij} = \begin{cases} -a_{ij}x_j, & i \neq j; \\ \sum_{\ell \neq i} a_{i\ell}x_\ell, & i = j. \end{cases}$$

If $Q_S$ is obtained from $Q$ by deleting all rows and columns indexed by $S$, then

$$\det Q_S = \sum_{A \in T(S)} w_A.$$ 

Observation. Theorem 6 is obtained from Theorem 7 by letting $a_{ij}$ be the number of edges from $v_j$ to $v_i$, letting all $x_j = 1$ and $S$ be a singleton.

EXAMPLES.

Proof of Theorem 7. By induction on $m = n - s$, where $s = |S|$. If $n = s$, then we get $1 = 1$. Suppose the theorem holds for $n - s \leq m - 1$. Consider any choice of $S \subseteq [n]$ with $|S| = s$ and any $a_{ij}$'s. We view $\det Q_S$ as a polynomial of degree $m$, $f_S(x_1, \ldots, x_n)$. For $i \in S$, call $x_i$ a root variable.

Two claims:

(1) In both, $\sum_{A \in T(S)} w_A$ and $f_S(x_1, \ldots, x_n)$ each term has degree 0 in some non-root variable

(2) For each non-root variable $x_i$, the terms in which $x_i$ is missing coincide in $\sum_{A \in T(S)} w_A$ and $f_S(x_1, \ldots, x_n)$.

Together, the claims imply the theorem, so let us prove them.

Proof of (1). Since $k < n$, in $w_A$ there are non-root vertices. The outdegree of a non-root leaf $v_i$ is 0, and hence $x_i$ is not present.
Consider $\det Q_S$. Recall that the sum of columns of $Q$ is the zero vector by definition. When we delete rows and columns corresponding to $S$, this is not true because in the diagonal elements some terms with $x_j$ for $j \in S$ may remain. But when we set all these variables to 0, the property recovers. So $f_S |_{x_j=0, j \in S} = 0$. This means each term of $Q_S$ contains $x_j$ for some $j \in S$. Since the degree of each term is $m$, some of the $m$ non-root variables is missing. □

**Proof of (2).** Consider the terms with no non-root $x_t$ in both polynomials. In $\sum_{A \in T(S)} w_A$ they arise from the arborescences where $x_t$ is a leaf. Each such arborescence $A$ is obtained from an arborescence $A'$ with $n - 1$ vertices by adding an arc to $v_t$. So if $T'$ is the set of all arborescences on $V(G) - v_t$, then the sum of terms omitting $x_t$ is

$$
\left( \sum_{A' \in T'(S)} w_{A'} \right) \left( \sum_{j \neq t} a_{t,j} \cdot x_j \right).
$$

In $f_S$ the terms omitting $x_t$ form $f_S(x_1, \ldots, x_{t-1}, 0, x_{t+1}, \ldots, x_n)$. The only non-zero entry in the $ts$ column of this determinant is $\sum_{j \neq t} a_{t,j} x_j$ in row $t$. Expand the determinant w.r.t. this column: By the IH, the remaining determinant equals $\left( \sum_{A' \in T'(S)} w_{A'} \right)$. □

Together, the claims prove the theorem. □

**AN EXAMPLE.**

**Eulerian circuits versus trees in digraphs**

**Lemma 8** (Lem. 6.1.33 in the book). For each Eulerian circuit in a digraph $G$ that begins from vertex $v$ along edge $e$, the set $T$ of edges last leaving each vertex apart from $v$ forms an in-tree with root $v$.

**Proof.** The outdegree in $T$ of each vertex apart from $v$ is 1, the outdegree of $v$ is 0, and there are no directed cycles. □

**Algorithm.**

**Input.** An Eulerian digraph $D$ and a spanning in-tree $T$.

**Step 1.** For each $u \in V(D)$, give an order of exiting edges s.t. (*) for each $u \neq v$, the edge of $T$ is the last.

**Step 2.** Starting from $v$ always go along the non-used edges smallest in the order.

**Lemma 9** (Lem. 6.1.35 in the book). The algorithm above always produces an Eulerian circuit in $D$.

**Proof.** We check that by (*) the our trail $L$ can stop only at $v$. Hence $L$ uses all edges entering $v$. Then for each in-neighbor $w$ of $v$, $L$ also uses all edges entering $w$. Continuing, we conclude that $L$ uses all edges at each vertex. □

**Theorem 10** (BEST Theorem, de Bruijn–van Aardenne-Ehrenfest, 1951, Smith–Tutte, 1941, Th. 6.1.36 in the book). Let $D$ be an Eulerian digraph with $V(D) = \{v_1, \ldots, v_n\}$,
where $d^+(v_i) = d^-(v_i) = d_i$ for all $1 \leq i \leq n$. Let $M = M_j$ be the number of spanning in-trees in $D$ with root $v_j$. Then the number of Eulerian circuits in $D$ is

$$M \prod_{i=1}^{n} (d_i - 1)!.$$ 

**Proof.** For each in-tree, the algorithm produces $\prod_{i=1}^{n} (d_i - 1)!$ distinct Eulerian circuits, and by Lemma 8, each Eulerian circuit is obtained this way. □

**Corollary 11.** In each Eulerian digraph, the number of spanning in-trees with root $v_i$ is equal for all $v_i$ (and equal to the number of spanning out-trees with root $v_i$).

**Corollary 12.** In each Eulerian digraph, the number of Eulerian circuits can be computed in polynomial time.

Note that for undirected graphs it is NP-complete to calculate the number of Eulerian circuits.

------------------- Here Lecture 4 ended.-------------------