A simple proof of the matrix-tree theorem, upward routes, and the matrix-tree-cycle theorem

Jules Jacobs

December 12, 2020

Abstract

The matrix-tree theorem states that the number of spanning trees in a graph is $\det L'$, where L' is the Laplacian matrix L of the graph, with any row and column deleted. We give a direct proof of the fact that $\det(xI + L)$ is the generating function of spanning forests with k roots. Our proof does not rely on the Cauchy-Binet formula, yet is arguably simpler than the standard proof, and applies to weighted & directed graphs as well.

The proof is based on a lemma that if A is the adjacency matrix of a graph where each vertex has at most one outgoing edge, then $\det(I - A) = 1$ if A is a forest and $\det(I - A) = 0$ otherwise. We also generalize this lemma to any graph, in which case $\det(I - xA)^{-1}$ is shown to be the generating function of *upward routes*.

Lastly, we generalize the matrix-tree theorem to a theorem about $\det(A + L)$ where A is the adjacency matrix of a second graph, which reduces to counting spanning forests when $A = xI$ and to the fact that $\det(A)$ counts signed cycle covers when $L = 0$. The all-minors matrix-tree theorem also follows as a corollary. For instance, the fact that $\det(L')$ counts spanning forests, where L' is L with any column i and row j deleted, follows by picking A to be the matrix where $A_{ij} = 1$ and zero elsewhere.

1 Introduction

Given finite sets of numbers $S_1, \ldots, S_n \subset \mathbb{R}$, we have the identity:

$$\left(\sum_{x_1 \in S_1} x_1\right) \cdots \left(\sum_{x_n \in S_n} x_n\right) = \sum_{x_1 \in S_1} \cdots \sum_{x_n \in S_n} x_1 \cdots x_n$$

On the right hand side, we get one term for every way of choosing $(x_1, \ldots, x_n) \in S_1 \times \cdots \times S_n$. Similarly, given finite sets of vectors $S_1, \ldots, S_n \subset \mathbb{R}^k$, we have the following identity, by multilinearity of the determinant:

$$\det\left(\sum_{v_1 \in S_1} v_1 \,\middle|\, \cdots \,\middle|\, \sum_{v_n \in S_n} v_n\right) = \sum_{v_1 \in S_1} \cdots \sum_{v_n \in S_n} \det(v_1 | \cdots | v_n)$$

Where $\det(v_1 | \ldots | v_n)$ is the determinant of a matrix with those columns. We shall see that Kirchoff's theorem can be obtained from this identity, but we first need some definitions and a lemma.

Definition 1.1. We define the concepts *1-graph*, *01-graph*, *forest*, *root*, and *tree*:

- A *1-graph* is a directed graph where each vertex has 1 outgoing edge.
- A *01-graph* is a directed graph where each vertex has 0 or 1 outgoing edges.
- A *forest* is a 01-graph with no cycles.
- The *roots* are vertices with 0 outgoing edges.

- A *tree* is a forest with one root.

If we start at a vertex v in a 01-graph and keep following the unique outgoing edge, then we either loop in a cycle, or we end up at a vertex with no outgoing edges. Therefore a general 01-graph consists of roots and cycles, plus trees converging onto the roots and cycles. A 1-graph is a 01-graph with no roots, so regardless of where we start, we always end up in a cycle.

2 The matrix-tree theorem

We need a lemma that give us an indicator function for forests.

Lemma 2.1. *Let A_G be the adjacency matrix of a 1-graph G, then*

$$\det(I - A_G) = \begin{cases} 1 & \text{if } G \text{ is empty} \\ 0 & \text{if } G \text{ is not empty} \end{cases}$$

Proof. If G is empty, we have a 0×0 matrix, which has determinant 1. If G is not empty, then each column of $I - A_G$ has one $+1$ diagonal entry and one -1 entry from A_G, so the sum of the rows is zero, so $\det(I - A_G) = 0$. This remains true in the presence of self loops. □

Lemma 2.2. *Let A_G be the adjacency matrix of a 01-graph G, then*

$$\det(I - A_G) = \begin{cases} 1 & \text{if } G \text{ is a forest} \\ 0 & \text{if } G \text{ has a cycle} \end{cases}$$

Proof. We calculate the determinant by repeatedly performing Laplace expansion on a column i that corresponds to a root. The column of a root has a single $+1$ entry on the diagonal, so performing Laplace expansion along this column deletes column i and row i. Row i contains all the incoming edges of the root. Therefore, this operation corresponds to deleting root i and all its incoming edges from the graph. Deleting a root may create new roots. Repeating this process of deleting roots, the remaining 1-graph will be empty iff the original graph was a forest. Applying the previous lemma gives the desired result. □

An alternative shorter proof using eigenvalues:

Proof. If G is a forest then A_G is nilpotent, so all its eigenvalues are 0, so all the eigenvalues of $I - A_G$ are 1, so $\det(I - A_G) = 1$. If G has a cycle, then A_G has 1 as an eigenvalue (take the eigenvector that is 1 on the cycle and 0 elsewhere), so $I - A_G$ has 0 as an eigenvalue, so $\det(I - A_G) = 0$. □

We now fix $n \times n$ matrices A and D over some ring:

- An arbitrary matrix A of edge weights (with A_{ij} being the weight of edge $i \to j$)
- A diagonal matrix D of vertex weights (with D_{ii} being the weight of vertex i).

Definition 2.1. The Laplacian matrix L is defined as having columns

$$L_i = \sum_j A_{ij}(e_i - e_j)$$

Definition 2.2. The weight of a forest G is

$$w(G) = \prod_{i \in \text{roots}(G)} D_{ii} \prod_{(i \to j) \in \text{edges}(G)} A_{ij}$$

We're ready to state Kirchoff's theorem for multiple-root forests in weighted directed graphs.

Theorem 2.3. *(Kirchoff, Tutte) The determinant* $\det(D+L)$ *is the weight-sum of all forests on n-vertices:*

$$\det(D+L) = \sum_{\textit{forest } G} w(G)$$

Proof. The i-th column of the matrix is $(D+L)_i = D_{ii}e_i + \sum_j A_{ij}(e_i - e_j)$, so

$$\det(D+L) = \det \left(\begin{array}{c|c|c|c} D_{11}e_1 & D_{22}e_2 & & D_{nn}e_n \\ + & + & & + \\ A_{11}(e_1-e_1) & A_{21}(e_2-e_1) & & A_{n1}(e_n-e_1) \\ + & + & \cdots & + \\ A_{12}(e_1-e_2) & A_{22}(e_2-e_2) & & A_{n2}(e_n-e_2) \\ \vdots & \vdots & & \vdots \\ A_{1n}(e_1-e_n) & A_{2n}(e_2-e_n) & & A_{nn}(e_n-e_n) \end{array} \right) = \sum_{\text{01-graph } G} w(G)\det(I-A_G) = \sum_{\text{forest } G} w(G)$$

The first step is by the definition of L. In the middle step we expand the determinant by multilinearity: we sum over all possible ways to pick one term of each column. To choose a 01-graph on n vertices, is to choose for each vertex i (column i) whether to make i a root (term $D_{ii}e_i$) or to give i an outgoing edge $i \to j$ (term $A_{ij}(e_i - e_j)$). Then we take the weights D_{ii} and A_{ij} out of the determinant, and we're left with $w(G)\det(I - A_G)$, where A_G is the adjacency matrix of the chosen 01-graph. The final step is applying the lemma. □

3 Upward routes

We now know that when A is the adjacency matrix of a 01-graph, then $\det(I-A) = 1$ if G is a forest and $\det(I-A) = 0$ if G has a cycle. One naturally wonders about the value of $\det(I-A)$ when A is an arbitrary adjacency matrix. The goal of this section is a combinatorial interpretation of $\det(I-xA)$ and $\det(I-xA)^{-1}$ for an arbitrary adjacency matrix, which generalizes the lemma to arbitrary graphs.

Definition 3.1. Given a directed graph G with an order on the vertices, we define *(strictly) upward loops* and *(strictly) upward routes*:

- An *upward loop* at vertex i is a path from i to i that does not visit vertices lower than i.
- A *strictly upward loop* at vertex i is a path from i to i that only visits vertices higher than i (except at the start/endpoint of the path, where it does visit i itself).
- A *(strictly) upward route* is a choice of (strictly) upward loop at each vertex.

Let $f_i(x)$ be the generating function of strictly upward loops of length k at vertex i. Then

$$f_i^*(x) = (1 - f_i(x))^{-1}$$

is the generating function of upward loops of length k at vertex i, because an upward loop splits uniquely into a sequence of strictly upward loops. Furthermore, the generating functions $f(x)$ and $f^*(x)$ of (strictly) upward routes of k edges are given by:

$$f(x) = \prod_{i=1}^{n} f_i(x) \qquad\qquad f^*(x) = \prod_{i=1}^{n} f_i^*(x)$$

Recall Cramer's rule:

Theorem 3.1. *(Cramer's rule) Let A be an invertible matrix and let $A_{[i,j]}$ be the same with column i and row j deleted, then:*

$$A^{-1}_{ij} = \frac{\det(A_{[i,j]})}{\det(A)}$$

From this, we get the following lemma that allows us to calculate $\det(A)^{-1}$ in terms of entries of inverses of submatrices of A:

Lemma 3.2. *Given a matrix A, provided both sides are defined,*

$$\det(A)^{-1} = \prod_{i=0}^{n-1} (A_{[1\ldots i,1\ldots i]})^{-1}_{11}$$

Where $A_{[1\ldots i,1\ldots i]}$ is the matrix A with the first i rows and columns deleted.

Proof. Cramer's rule implies:

$$\det(A)^{-1} = A^{-1}_{1,1} \cdot \det(A_{[1,1]})^{-1}$$

Continuing this by induction, we get:

$$\begin{aligned}\det(A)^{-1} &= A^{-1}_{1,1} \cdot \det(A_{[1,1]})^{-1} \\ &= A^{-1}_{1,1} \cdot (A_{[1,1]})^{-1}_{11} \cdot \det(A_{[1..2,1..2]})^{-1} \\ &= \cdots \\ &= A^{-1}_{1,1} \cdot (A_{[1,1]})^{-1}_{11} \cdot (A_{[1..2,1..2]})^{-1}_{1,1} \cdots (A_{[1..n-1,1..n-1]})^{-1}_{1,1} \cdot 1\end{aligned}$$

□

Lemma 3.3. *The generating function of upward loops is $((I - xA)_{[1\ldots i-1,1\ldots i-1]})^{-1}_{11}$.*

Proof. Given an adjacency matrix B, $(I - xB)^{-1} = I + xB + x^2B^2 + \cdots$ is the matrix generating function of paths, so $(I - xB)^{-1}_{11}$ is the generating function of loops from vertex 1 to 1. To obtain upward loops at vertex i in A we take $B = (I - xA)_{[1\ldots i-1,1\ldots i-1]}$ with the first $i - 1$ rows and columns deleted. □

We combine these lemmas to obtain:

Theorem 3.4. *The generating function of upward routes with k edges is* $\det(I - xA)^{-1}$.

Proof. Combine the preceding two lemmas with $f^*(x) = \prod_{i=1}^{n} f_i^*(x)$. □

Corollary 3.4.1. The number of upward routes of k edges does not depend on the order of the vertices.

Proof. If we permute the order of the vertices by a permutation P, the generating function

$$\det(I - xPAP^{-1}) = \det(P(I - xA)P^{-1}) = \det(I - xA)$$

stays the same. A bijective proof is left as an exercise. □

Corollary 3.4.2. For an arbitrary adjacency matrix A,

$$\det(I - xA) = \prod_{i=1}^{n} (1 - f_i(x))$$

Where $f_i(x)$ is the generating function of strictly upward loops at vertex i.

Proof. Use the relationship between the generating functions:

$$\det(I - xA) = \bar{f}(x)^{-1} = \prod_{i=1}^{n} \bar{f}_i(x)^{-1} = \prod_{i=1}^{n}(1 - f_i(x))$$

$\square$

Note that $\det(I - xA)$ is a polynomial, even though $f_i(x)$ is a power series. The main lemma follows as a corollary, which gives us a third proof of the lemma:

Corollary 3.4.3. Let A_G be the adjacency matrix of a 01-graph G, then

$$\det(I - A_G) = \begin{cases} 1 & \text{if } G \text{ is a forest} \\ 0 & \text{if } G \text{ has a cycle} \end{cases}$$

Proof. If G is a forest, then it has no strictly upward loops, so $f_i(x) = 0$, so $\det(I - xA_G) = 1$. If G has a cycle, let i be the lowest vertex on the cycle. Then $f_i(x) = x^k$ where k is the length of the cycle. Now substitute $x = 1$ to obtain $\det(I - A_G) = 0$. $\square$

4 The matrix-tree-cycle theorem

We shall now generalize the matrix-tree theorem about $\det(D + L)$ to the *matrix-tree-cycle theorem* by allowing D to be a general matrix. This theorem will express $\det(D + L)$ as a sum over 1-graphs with its edges labeled with T (for tree) or C (for cycle).

Definition 4.1. A TC-labeled 1-graph is called *tree-cyclic* if the T-edges form a forest, and each C-edge is part of a cycle.

Definition 4.2. The sign of a TC-labeled 1-graph G is

$$(-1)^G = (-1)^{\#\text{cycles}+\#\text{C-edges}}$$

where #cycles is the number of cycles of G and #C-edges is the total number of C-labeled edges in G.

We associate a matrix M_G with a TC-labeled 1-graph G. This matrix will play the role that $I - A_G$ played in the matrix-tree theorem.

Definition 4.3. The matrix M_G is given by columns:

$$(M_G)_i = \begin{cases} e_j & \text{if the outgoing edge } i \to j \text{ has label C} \\ e_i - e_j & \text{if the outgoing edge } i \to j \text{ has label T} \end{cases}$$

The following lemma generalizes the main lemma of the matrix-tree theorem.

Lemma 4.1. *The determinant of M_G is given by:*

$$\det(M_G) = \begin{cases} (-1)^G & \textit{if } G \textit{ is tree-cyclic} \\ 0 & \textit{otherwise} \end{cases}$$

Proof. Calculate the determinant in steps:

1. Perform Laplace expansion on vertices with no predecessors. This makes the determinant zero if the successor is a C-edge, and deletes the corresponding vertex if the successor is a T-edge.

2. We are now left with a disjoint set of cycles. If there is a cycle consisting solely of T-edges, the determinant is zero because the corresponding rows sum to zero.

3. We are now left with a disjoint set of cycles where each cycle has at least one C-edge. Use the C-edges to turn the T-edges into C-edges by row operations. The determinant obtains obtains a -1 sign for each such switch.
4. We are now left with a disjoint set of cycles where each cycle consists solely of C-edges. In other words, a permutation matrix. The determinant of a permutation matrix is $(-1)^{\#\text{cycles}+\#\text{edges}}$

If there was a C-edge not part of a cycle we have obtained 0 in step 1, and if there was a cycle among T-edges we have obtained 0 in step 2. For the remaining graphs, we have obtained a -1 sign for each T-edge in the cycles, so together with $(-1)^{\#\text{cycles}+\#\text{edges}}$ we are left with $(-1)^{\#\text{cycles}+\#\text{C-edges}} = (-1)^G$. □

We now fix $n \times n$ matrices A and D over some ring:

- An arbitrary matrix A of weights for T-edges (with A_{ij} being the weight of edge $i \xrightarrow{T} j$)
- An arbitrary matrix D of weights for C-edges (with D_{ij} being the weight of edge $i \xrightarrow{C} j$).
- As before, the Laplacian L is given by $L_i = \sum_j A_{ij}(e_i - e_j)$.

Definition 4.4. The weight of a tree-cyclic 1-graph G is

$$w(G) = \prod_{\text{C-edge } (i \to j) \in G} D_{ij} \prod_{\text{T-edge } (i \to j) \in G} A_{ij}$$

We're ready to state the matrix-tree-cycle theorem.

Theorem 4.2. *The determinant* $\det(D + L)$ *is the signed weight-sum of tree-cyclic graphs:*

$$\det(D + L) = \sum_{\textit{tree-cyclic } G} (-1)^G w(G)$$

Proof. The i-th column of the matrix is $(D + L)_i = \sum_j D_{ij} e_j + \sum_j A_{ij}(e_i - e_j)$, so

$$\det(D+L) = \det\left(\begin{array}{c|c|c|c} D_{11}e_1 & D_{21}e_1 & & D_{n1}e_1 \\ + & + & & + \\ \vdots & \vdots & & \vdots \\ + & + & & + \\ D_{1n}e_n & D_{2n}e_n & & D_{nn}e_n \\ + & + & \cdots & + \\ A_{11}(e_1 - e_1) & A_{21}(e_2 - e_1) & & A_{n1}(e_n - e_1) \\ + & + & & + \\ \vdots & \vdots & & \vdots \\ + & + & & + \\ A_{1n}(e_1 - e_n) & A_{2n}(e_2 - e_n) & & A_{nn}(e_n - e_n) \end{array}\right) = \sum_{\text{TC-1-graph } G} w(G) \det(M_G) = \sum_{\text{tree-cyclic } G} (-1)^G w(G)$$

In the middle step we have again expanded the determinant by multilinearity: we sum over all possible ways to pick one term of each column. To choose a TC-labeled 1-graph on n vertices, is to choose for each vertex i (column i) an outgoing edge $i \to j$ and a label C (term $D_{ij} e_j$) or T (term $A_{ij}(e_i - e_j)$) for this edge. The final step is applying the lemma. □

Bunch of corollaries:

Corollary 4.2.1. $\det(A)$ is the number of signed cycle covers.

Corollary 4.2.2. $\det(I + L)$ is the number of spanning forests.

Corollary 4.2.3. $\det(L_{[i,j]})$ spanning trees, for all i, j.

Corollary 4.2.4. All-minor matrix tree theorem.

Corollary 4.2.5. Undirected matrix tree theorem.