MS&E 337: Spectral Graph Theory and Algorithmic Applications **Spring 2015**

Lecture 2-3: 4/7/2015

Lecturer: Prof. Amin Saberi *Scribe: Simon Anastasiadis and Nolan Skochdopole*

2.1 Laplacian Variants

Let L be the Laplacian of $G(V, E)$ with eigenvalues λ_i and corresponding eigenvectors v_i. We may add the following definitions:

$$\text{Laplacian} \quad L = \sum_{i=1}^{n} \lambda_i v_i v_i^T$$
$$\text{Pseudo-inverse} \quad L^\dagger = \sum_{i=2}^{n} \frac{1}{\lambda_i} v_i v_i^T$$
$$\text{Square-root} \quad L^{1/2} = \sum_{i=1}^{n} \sqrt{\lambda_i} v_i v_i^T$$

We also define the following matrix:

$$\tilde{L}_G = L_G + \frac{1}{n} J = L_G + \frac{1}{n} \mathbb{1}\mathbb{1}^T$$

Where $\mathbb{1}$ is the vector of all ones, so J is the matrix of all ones.

Lemma 2.1. $\det(\tilde{L}_G) = n \times$ *number of spanning trees in* G.

Proof. Recall that for $n \times n$ matrices A and B, $\det(A + B) = \sum_S \det([A_S B_{\bar{S}}])$ where S iterates over subsets of $[n]$. We apply this inequality on $\tilde{L}_G = L_G + \frac{1}{n} J$.
If we take zero columns from J the determinant is zero. If we take two or more columns from J the determinant is also zero. Hence the only cases where the summand is non-zero are when we take exactly one column from J. This is equivalent to taking the determinant of L_{ii} which gives you the number of spanning trees (see Lecture 1). □

2.2 Sampling spanning trees

Suppose we want to select a spanning tree uniformly at random from the set of all possible spanning trees of G. There may be exponentially many spanning trees so we need techniques that are faster than complete enumeration. There are both algebraic and combinatorial approaches to uniform sampling of spanning trees. In this lecture, we focus on algebraic approaches.

Since we can count spanning trees, we can sample from this distribution efficiently. First, of all we can compute the probability that each edge e is in a random spanning tree. Say μ is a uniform distribution on all spanning trees of G. Then,

$$\mathbb{P}_{T\sim\mu}\left[e \in T\right] = 1 - \mathbb{P}_{T\sim\mu}\left[e \notin T\right] = 1 - \frac{\det(\tilde{L}_{G\setminus\{e\}})}{\det(\tilde{L}_G)}.$$

Once we can calculate the probability of an edge being in a random spanning tree we can sample a tree from μ efficiently. The following algorithm runs in roughly $O(n^5)$ and samples a uniform spanning tree in a given graph.

Algorithm 1 Sampling a Uniform Spanning Tree

```
Let G_0 = G, and T = {}.
Choose an arbitrary ordering of the edges, e_1, ..., e_m.
for i = 1 → m do
    Let p be the probability that e_i is in a random spanning tree of G_{i-1}.
    With probability p, add e_i to T and let G_i = G_{i-1}/{e_i} (i.e., contract e_i).
    With the remaining probability, let G_i = G_{i-1} \ {e_i} (i.e., delete e_i).
end for
```

2.3 Computing marginal probabilities

Let μ be a uniform distribution of all spanning trees of G. Let $\mu_e := \mathbb{P}_{T\sim\mu}\left[e \in T\right]$, be the marginal probability of edge e. In this section we write analytical expression for μ_e.

Lemma 2.2. *For any edge e,*

$$\mu_e = b_e^{\intercal} L_G^{\dagger} b_e.$$

Next, we use the determinant rank one update formula to prove the above lemma.

Proposition 2.3. *For a PD matrix $A \in \mathbb{R}^{n\times n}$ and any vectors $x, y\mathbb{R}^n$,*

$$\det(A + xx^{\intercal}) = \det(A)(1 + x^{\intercal}A^{-1}x).$$

Proof. First,

$$\det(A + x^{\intercal}x) = \det(A^{1/2}(I + A^{-1/2}x^{\intercal}A^{-1/2})A^{1/2}) = \det(A)\det(I + A^{-1/2}xx^{\intercal}A^{-1/2}).$$

It is easy to see $I + A^{-1/2}xx^{\intercal}A^{-1/2}$ has $n-1$ eigenvalues that are 1 and 1 eigenvalue that is $1 + x^{\intercal}A^{-1}x$. In words for any vector y, a rank 1 update, $I + yy^T$, only shifts one of the eigenvalues of I by $\langle y, y\rangle$. Therefore,

$$\det(I + A^{-1/2}xx^{\intercal}A^{-1/2}) = 1 + x^{\intercal}A^{-1}x,$$

that completes the proof. □

Proof of Lemma 2.2.

$$\mu_e = 1 - \mathbb{P}\left[e \notin T\right] = 1 - \frac{\det(\tilde{L}_G - b_e b_e^{\intercal})}{\det(\tilde{L}_G)}$$

By Proposition 2.3, we can write

$$\mu_e = 1 - \frac{\det(\tilde{L}_G)(1 - b_e^\intercal \tilde{L}_G^{-1} b_e)}{\det(\tilde{L}_G)} = b_e^\intercal L_G^\dagger b_e,$$

where we used that $\tilde{L}_G b_e = L_G b_e$. □

The quantity $\text{Reff}(e) := b_e^\intercal L_G^\dagger b_e$ is also known as the *effective resistance* of edge e. We will talk about effective resistance in more details in the next lecture.

2.4 Isotropic vectors and a natural normalization for edge vectors

A set of vectors $y_1, \ldots, y_m \in \mathbb{R}^n$ are in isotropic position if for any unit vector x,

$$\begin{aligned}\sum_{i=1}^m \langle x, y_i \rangle^2 &= \sum_{i=1}^m x^\intercal y_i y_i^\intercal x \\ &= x^\intercal \left(\sum_{i=1}^m y_i y_i^\intercal \right) x = 1.\end{aligned}$$

Put it in another way, a set of vectors are in isotropic position if every direction in the space looks like every other direction, that is for every direction x if we project all of the vectors onto x the sum of the squares of the projections is invariant over the choices of x.

There is a natural linear transformation that can make any given set of vectors $\{b_1, \ldots, b_m\}$ isotropic.

$$y_i = \left(\sum_{i=1}^m b_i b_i^\intercal \right)^{\dagger/2} b_i.$$

Then,

$$\begin{aligned}\sum_{i=1}^m y_i y_i^\intercal &= \sum_{i=1}^m \left(\sum_{i=1}^m b_i b_i^\intercal \right)^{\dagger/2} b_i b_i^\intercal \left(\sum_{i=1}^m b_i b_i^\intercal \right)^{\dagger/2} \\ &= \left(\sum_{i=1}^m b_i b_i^\intercal \right)^{\dagger/2} \left(\sum_{i=1}^m b_i b_i^\intercal \right) \left(\sum_{i=1}^m b_i b_i^\intercal \right)^{\dagger/2} = I.\end{aligned}$$

Geometrically, this transformation equalizes the eigenvalues of the matrix $\sum_{i=1}^m b_i b_i^\intercal$. For us, this is a very useful transformation, when dealing with vectors $\{b_e\}_{e \in E}$. Let us define

$$y_e := L_G^{\dagger/2} b_e. \tag{2.1}$$

If G is weighted, then we define

$$y_e := \sqrt{w(e)} \cdot L_G^{\dagger/2} b_e.$$

It follows from the discussion in the previous section that $\{y_e\}_{e \in E}$ are isotropic.

In the last lecture, we saw that for any set $F \subseteq E$ that is a spanning tree, $\det(B_F B_F^\intercal) = n - 1$. By **??** $\det(B_F B_F^\intercal)$ is the square of the volume of the parallelepiped

$$\left\{ \sum_{e \in F} \alpha_e b_e : 0 \leq \alpha_e \leq 1, \forall e \in F \right\}.$$

Now, let us calculate the volume of a basis after the isotropic transformation of a tree $F = \{b_{e_1}, \ldots, b_{e_{n-1}}\}$. Let

$$\mathcal{B}_F = L_G^{\dagger/2} B_F = \begin{pmatrix} L_G^{\dagger/2} b_{e_1} \\ L_G^{\dagger/2} b_{e_2} \\ \vdots \\ L_G^{\dagger/2} b_{e_{n-1}} \end{pmatrix}.$$

Then,

$$\begin{aligned} \det(\mathcal{B}_F \mathcal{B}_F^{\intercal}) &= \det(\tilde{L}_G^{\dagger/2} B_F B_F^{\intercal} \tilde{L}_G^{\dagger/2}) \\ &= \det(\tilde{L}_G^{\dagger}) \det(B_F B_F^{\intercal}) = \frac{1}{|\mathcal{T}|}. \end{aligned}$$

The last equality uses the matrix tree theorem. So, the square of the volume of each basis in the isotropic position is exactly the probability that the basis is chosen in a uniform distribution of bases.

On the other hand, for any edge e, the probability that e is in a random spanning tree is the same as the 1 dimensional volume of y_e.

$$\det(y_e y_e^{\intercal}) = b_e L_G^{\dagger} b_e = \mathbb{P}_{T\sim\mu}\left[e \in T\right].$$

So, one might suspect a generalization. That is to show that for any arbitrary set F of edges, the probability of edges of F appearing in a random spanning tree is equal to the square of the volume of their corresponding parallelepiped in the isotropic mapping. This is indeed true and it is proved by Burton and Pemantle as we will see in the next section.

2.5 The Burton-Pemantle Theorem

We discuss a beautiful theorem of Burton and Pemantle [**BP93**] that shows that random spanning tree distributions are families of determinantal measures.

Theorem 2.4 (Burton and Pemantle [**BP93**])**.** *Given a graph G, let $Y \in \mathbb{R}^{E\times E}$ where for each pair of edges $e, f \in F$,*

$$Y(e, f) = \langle y_e, y_f\rangle.$$

Then, for any set of edges $F \subseteq E$,

$$\mathbb{P}_{T\sim\mu}\left[F \subseteq T\right] = \det(\mathcal{B}_F \mathcal{B}_F^{\intercal}) = \det(Y_F).$$

Before proving the above theorem, let us discuss some of its implications. An immediate consequence of Theorem 2.4 is that each pair of edges are negatively correlated.

Fact 2.5 (Pairwise Negative Correlation)**.** *For any pair of edges $e, f \in E$,*

$$\mathbb{P}_{T\sim\mu}\left[e, f \in T\right] \le \mathbb{P}\left[e \in T\right] \cdot \mathbb{P}\left[f \in T\right].$$

To see the proof, by Burton, Pemantle,

$$\begin{aligned} \mathbb{P}\left[e, f \in T\right] &= \det\begin{pmatrix} \langle y_e, y_e\rangle & \langle y_e, y_f\rangle \\ \langle y_e, y_f\rangle & \langle y_f, y_f\rangle \end{pmatrix} \\ &= \|y_e\|^2 \cdot \|y_f\|^2 - \langle y_e, y_f\rangle\langle y_f, y_e\rangle. \end{aligned}$$

Using Lemma 2.2,

$$\mathbb{P}[e, f \in T] - \mathbb{P}[e \in T] \cdot \mathbb{P}[f \in T] = -\langle y_e, y_f \rangle^2 \leq 0.$$

So, e, f are negatively correlated. But, we can even analytically write down the correlation between each pair of edges. For example, if $\langle y_e, y_f \rangle = 0$ then we can say edge e is independent of edge f.

More generally, we can define negative correlation of a subset of edges.

Definition 2.6 (Negative Correlation)**.** *For a distribution $\mu : 2^E \to \mathbb{R}_+$, we say a set of edges $F \subseteq E$, are negatively correlated if*

$$\mathbb{P}_{T\sim\mu}[F \subseteq T] \leq \prod_{e \in F} \mathbb{P}[e \in T].$$

A similar argument shows that in any random spanning tree distribution any set of edges are negatively correlated. In particular, the LHS of the above equation is the square of the volume of the $|F|$-parallelepiped defined on $\{y_e\}_{e\in F}$ and the RHS is just $\prod_{e\in F} \|y_e\|^2$.

Another way to argue the above is to use the fact that effective resistances only decrease when you short circuit the path between two nodes in an elecrical network.

2.6 Proof of Theorem 2.4

Let G be an undirected, connected graph with Laplacian $L = BB^T = \sum_{e\in E} b_e b_e^T$. Recall the Cauchy-Binet theorem (for $A \in \mathbb{R}^{n\times m}$ and $B \in \mathbb{R}^{m\times n}$):

$$\det(AB) = \sum_{S \in \binom{m}{n}} \det((BA)_S) := \det_n(BA)$$

Also recall from the first lecture the following expansion of the determinant:

$$\det(tI - A) = \sum_{k=0}^{n} (-1)^k t^{n-k} \det_k(A)$$

Let $y_e = L^{\frac{\dagger}{2}} b_e$. Then the y_e vectors are in isotropic position, i.e. $\sum_e y_e y_e^T = L^{\frac{\dagger}{2}} \left(\sum_e b_e b_e^T\right) L^{\frac{\dagger}{2}} = I_{n-1}$. We saw last class $b_e^T L^\dagger b_e = \mathrm{R}_{\text{eff}}(e)$, the probability that an edge e is in a uniform random spanning tree. Now we state and prove the Burton, Pemantle theorem which we saw at the end of the previous lecture.

Theorem [Burton and Pemantle]: Given a graph G, let $Y \in \mathbb{R}^{m\times m}$ be the the matrix with $Y(e, f) = y_e^T y_f$. Let μ be the weighted uniform distribution of spanning trees on G, so $\mu(T) = \prod_{e\in T} w(e)$. Then for any set of edges $F \subseteq E$, $\Pr_{T\sim\mu}[F \subseteq T] = \det(Y_F)$.

We proceed by induction in the size of our subset F. Assume the claim is true for any set of size less than $|F|$. By the inclusion-exclusion principle, we have:

$$\begin{aligned}\Pr[F \cap T \neq \emptyset] &= \sum_{k=1}^{|F|}(-1)^{k-1} \sum_{S \in \binom{F}{k}} \Pr[S \subseteq T] \\ &= \sum_{k=1}^{|F|-1} (-1)^{k-1} \sum_{S \in \binom{F}{k}} \det(Y_S) + (-1)^{|F|-1}\Pr[F \subseteq T]\end{aligned}$$

By Cauchy-Binet, for any set $S \subseteq F$, $\det(Y_S) = \det_{|S|}\left(\sum_{e \in S} y_e y_e^T\right)$. Then, we have the following:

$$\begin{aligned}\Pr[F \cap T \neq \emptyset] &= \sum_{k=1}^{|F|-1} (-1)^{k-1} \sum_{S \in \binom{F}{k}} \det_k\left(\sum_{e \in S} y_e y_e^T\right) + (-1)^{|F|-1}\Pr[F \subseteq T] \\ &= \sum_{k=1}^{|F|-1} (-1)^{k-1}\det_k\left(\sum_{e \in F} y_e y_e^T\right) + (-1)^{|F|-1}\Pr[F \subseteq T]\end{aligned}$$

Notice that the coefficients the last equation above are the coefficients of the characteristic polynomial of $\sum_{e \in F} y_e y_e^T$. By our expansion of the characteristic polynomial above (with $t = 1$) we have:

$$\begin{aligned}\det\left(I - \sum_{e \in F} y_e y_e^T\right) &= 1 - \sum_{k=1}^{|F|}(-1)^{k-1}\det_k\left(\sum_{e \in F} y_e y_e^T\right) \\ &= 1 - \Pr[F \cap T \neq \emptyset] + (-1)^{|F|-1}\Pr[F \subseteq T] + (-1)^{|F|}\det_{|F|}\left(\sum_{e \in F} y_e y_e^T\right)\end{aligned}$$

But we also have:

$$\begin{aligned}1 - \Pr[F \cap T \neq \emptyset] &= \Pr[F \cap T = \emptyset] \\ &= \frac{\det\left(\tilde{L}_G - \sum_{e \in F} b_e b_e^T\right)}{\det(\tilde{L}_G)} \\ &= \det\left(I - \sum_{e \in F} y_e y_e^T\right)\end{aligned}$$

Combining the two equations, we get:

$$\begin{aligned}\det\left(I-\sum_{e\in F}y_ey_e^T\right) &= 1-\Pr[F\cap T\neq\emptyset]+(-1)^{|F|-1}\Pr[F\subseteq T]+(-1)^{|F|}\det_{|F|}\left(\sum_{e\in F}y_ey_e^T\right)\\ &= \det\left(I-\sum_{e\in F}y_ey_e^T\right)+(-1)^{|F|-1}\Pr[F\subseteq T]+(-1)^{|F|}\det_{|F|}\left(\sum_{e\in F}y_ey_e^T\right)\\ &\Leftrightarrow (-1)^{|F|-1}\Pr[F\subseteq T]+(-1)^{|F|}\det_{|F|}\left(\sum_{e\in F}y_ey_e^T\right)=0\\ &\Leftrightarrow \Pr[F\subseteq T]=\det_{|F|}\left(\sum_{e\in F}y_ey_e^T\right)=\det(Y_F)\end{aligned}$$

□

2.7 Remarks

First, note that Y is a projection matrix, i.e. $Y^2 = B^TL^\dagger BB^TL^\dagger B = Y$, with $n-1$ eigenvalues of 1 and $m-n+1$ eigenvalues of 0. We also have $b_e^TL^\dagger b_e = \mathrm{R_{eff}}(u,v)$ where $e=(u,v)$.

The effective resistance is a metric, so it obeys the triangle inequality. Effective resistance is also convex in conductance of the edges. We may write $L = BB^T = BWB^T$ where W is the conductance matrix (keeping track of weighted parallel paths).

If we let $f_v = L^{\frac{\dagger}{2}}1_v$, where 1_v is the vector with 1 in the v position, then we have an embedding into $\mathbb{R}^2$ by the following: $||f_v - f_u||^2 = ||L^{\frac{\dagger}{2}}(1_u - 1_v)||^2 = (1_u-1_v)^TL^\dagger(1_u-1_v) = \mathrm{R_{eff}}(u,v)$.

We may consider random walks on graphs. For example the gambler's ruin problem is just a chain graph. Say the left end is 0 and the right end is n. If we start with a dollar and keep playing a fair game, then we can define the probability of success (reaching n dollars before going bankrupt) for each node by the following recurrence and boundary conditions: $p_0 = 0$, $p_1 = \frac{1}{2}p_2$, $p_i = \frac{1}{2}(p_{i-1}+p_{i+1})$, $p_n = 1$.

In a general graph, we may set up a similar set of equations to describe a random walk: $p_x = \frac{1}{\deg(x)}\sum_{y\sim x}p_y$. If we set up a node as an "escape" to end the walk we may use the effective conductance to describe an escape probability: $p_{\mathrm{escape}}(s\to t) = \frac{C_{\mathrm{eff}}(s,t)}{\deg(s)}$.

The commute time between two nodes i and j is $2m\mathrm{R_{eff}}(i,j)$ and we have the following bounds for the cover time of a graph $C(G)$: $\max_{i,j}2m\mathrm{R_{eff}}(i,j) \le C(G) \le \max_{i,j}\mathrm{R_{eff}}(i,j)2mO(\log n)$.

2.8 Other ways to sample spanning trees

- Feder-Mihail chain

1. Start with an arbitrary spanning tree
2. Add a random edge to this spanning tree
3. Look at the resulting cycle and remove one edge at random
4. The result is a spanning tree

The above process defines a Markov-chain on the set of all spanning trees. However this process has a slow mixing time, at least, $O(n^9)$.

- Broder and Aldons
  This result has been independently rediscovered several times.

  1. Pick an arbitrary starting node
  2. Consider a random walk that covers the graph
  3. For every node other than the starting node, identify the edge that was used to first visit the node.
  4. Add this edge to the tree

  This approach runs in the covering time of the graph, which is at worst $O(n^3)$.

- David Wilson This method is faster than the covering time approach.

  1. Start with a random walk between two nodes (remove cycles)
  2. This is the current tree
  3. While there is a node disconnected from the current tree
  4. Start a random walk from this node, stopping once you reach the current tree
  5. Remove cycles from the random walk, add the resulting path to the tree