CS 229r Fall 2020: Course Overview

Wednesday, August 19, 2020 8:04 PM

Salil Vadhan

OH: today 4:30-6
tomorrow 9-12, 2-5

TF: Chi-Ning Chou
OH: 8/25 4:30-6

Agenda

1) Subject Matter
2) Syllabus
3) Q & A

Subject Matter: Spectral Graph Theory in CS

eigenvalues + eigenvectors of graphs
+ related linear algebra

↓

understanding graph structure, eg
- how well connected
- clustering + colorability
- mixing of random walks

- solving linear systems
- derandomization
- sampling via MCMC
- web search
- max flow and other graph algorithms

Matrices Associated with a graph G

Adjacency matrix $M_G(i,j) = \#$ edges from v_j to v_i

Random-walk matrix $W_G(i,j) = \dfrac{\#\text{ edges from } v_j \text{ to } v_i}{\deg(v_j)}$

$$W_G = M_G D_G^{-1} \qquad D_G = \begin{pmatrix} \deg(v_1) & & 0 \\ & \deg(v_2) & \\ 0 & & \ddots \ \deg(v_n) \end{pmatrix}$$

Laplacian $L_G = D_G - M_G$

Normalized Laplacian $N_G = I - D_G^{-1/2} M_G D_G^{-1/2}$

Random-Walk Laplacian $I - W_G = D_G^{1/2} N_G D_G^{-1/2}$

Q: why might we sometimes prefer N_G over $I - W_G$?

Eigenvalues + Eigenvectors

Assume G undirected and d-regular (so $D_G = dI$).

Then $N_G = I - M_G/d = I - W_G$ is symmetric

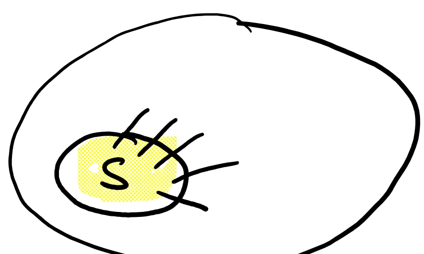
$\rightarrow$ orthogonal basis of eigenvectors $v_1, \dots, v_n \in \mathbb{R}^n$ (Spectral Theorem)

with real eigenvalues $\lambda_1 \le \lambda_2 \le \lambda_3 \le \dots \le \lambda_n$

One eigenvector: uniform distribution $u = (1/n, 1/n, \dots, 1/n) = \frac{\vec{1}}{n}$

- $W_G u = u$ (because G regular)
- $N_G u = (I - W_G) u = 0$

Facts:
- $\lambda_1 = 0$ is smallest eigenvalue of N_G
- λ_2 tells us a lot about G
  - $\rightarrow$ random walks converge to u in $O(\log n)/\lambda_2$ steps
  - $\rightarrow$ every set S of at most $n/2$ vertices has $\gtrsim \left(\frac{\lambda_2}{2}\right) \cdot |S|$ edges leaving it



i.e. G has conductance at least $\lambda_2/2$

Approximate converse (Cheeger's Inequality):

j λ_2 small $\Rightarrow$ can partition G cutting few edges

- Higher-order Cheeger (2011)

$\lambda_1, \dots, \lambda_k$ small $\Rightarrow$ partition G into k pieces cutting few edges

---

Other Course Topics

Markov Chains

λ_2 large $\Rightarrow$ random walks mix fast

$\Rightarrow$ fast sampling algorithm ("MCMC") for stationary distribution (even for exponentially large G)

$\Rightarrow$ fast approximate counting algorithms

Pagerank (Google ranking of webpages)

= stationary distribution of (directed) web-link graph + complete graph

Expander Graphs

graphs that are very well-connected (λ_2 large) but have few edges (e.g. $d = O(1)$)

Many applications in theoretical CS,
including derandomization

Intuition: random edge in an expander is like picking two independent vertices, but only uses $\approx$ half as many random bits.

Spectral Sparsification [2004+]

Expanders = sparse approx of complete graph

Now: sparse approx of any graph

$\rightarrow$ faster algorithms by using sparse approx.

Effective Resistances

Fact: If G is a network of resistors and we feed in external current $\vec{i}$
Then vector of voltages $\vec{v} = L_G \vec{i}$.

Kirchhoff's Matrix-Tree Theorem:

Coefficient of x in $\det(xI - L_G) = (x-\lambda_1)(x-\lambda_2)\cdots(x-\lambda_n)$

equals $\pm$ (# spanning trees in G).

Q: algorithmic application?

Solving Laplacian Systems

Given G and b, find x s.t. $L_G x = b$.

We'll see:

- randomized algs w/ time $\tilde{O}(m)$ [2004+]
  (m = # edges in G)
- deterministic algs w/ space $\tilde{O}(\log n)$ [2017]

cf. best known algorithms for general linear systems: $O(n^{2.37})$ time, $O(\log^2 n)$ space

Application: fastest known algs for max-flow
(idea: reduce max-flow to several electrical flow problems, eg $\vec{v} = L_G \vec{i}$)