ADFOCS22 Summer 2021

Graph and Matrix Approximation via Sampling

R. Kyng *Exercise Set 1 — Thursday, August 5th*

Exercise 1.

Let $\boldsymbol{L}$ be the Laplacian of a connected, weighted, undirected graph G, and let $\boldsymbol{B} \in \mathbb{R}^E$ be the associated edge-vertex incidence matrix. Let $\boldsymbol{d} \in \mathbb{R}^V$, $\boldsymbol{d} \perp \boldsymbol{1}$ be a demand vector.

The goal of this exercise is to prove that

$$\max_{\boldsymbol{x} \in \mathbb{R}^V} \boldsymbol{x}^\top \boldsymbol{d} - \frac{1}{2}\boldsymbol{x}^\top \boldsymbol{L}\boldsymbol{x} = \min_{\boldsymbol{f} \in \mathbb{R}^E} \frac{1}{2}\sum_e \boldsymbol{r}(e)\boldsymbol{f}(e)^2 \quad \text{s.t. } \boldsymbol{B}\boldsymbol{f} = \boldsymbol{d}.$$

We'll break that down into a few steps.

Let $\boldsymbol{f} \in \mathbb{R}^E$ be an arbitrary flow that satisfies $\boldsymbol{B}\boldsymbol{f} = \boldsymbol{d}$, i.e. it routes the demand $\boldsymbol{d}$. Let $\boldsymbol{x} \in \mathbb{R}^V$ be arbitrary voltages, i.e. not necessarily electrical voltages associated with the demand.

(i) Prove that

$$\frac{1}{2}\sum_e \boldsymbol{r}(e)\boldsymbol{f}(e)^2 = \boldsymbol{x}^\top \boldsymbol{d} - \left(\sum_{(u,v)\in E} (\boldsymbol{x}(u) - \boldsymbol{x}(v))(\boldsymbol{f}(u,v)) - \frac{1}{2}\boldsymbol{r}(u,v)\boldsymbol{f}(u,v)^2\right)$$

Hint: use that $\boldsymbol{x}^\top(\boldsymbol{B}\boldsymbol{f} - \boldsymbol{d}) = 0$.

(ii) Prove that

$$(\boldsymbol{x}(u) - \boldsymbol{x}(v))(\boldsymbol{f}(u,v)) - \frac{1}{2}\boldsymbol{r}(u,v)\boldsymbol{f}(u,v)^2 \leq \frac{1}{2}\frac{(\boldsymbol{x}(u) - \boldsymbol{x}(v))^2}{\boldsymbol{r}(u,v)}.$$

(iii) Conclude that $\frac{1}{2}\boldsymbol{f}^\top \boldsymbol{R}\boldsymbol{f} \geq \boldsymbol{x}^\top \boldsymbol{d} - \frac{1}{2}\boldsymbol{x}^\top \boldsymbol{L}\boldsymbol{x}$.

(iv) Assume we are given $\tilde{\boldsymbol{x}}$ and $\tilde{\boldsymbol{f}}$ such that

$$\boldsymbol{L}\tilde{\boldsymbol{x}} = \boldsymbol{d} \text{ and } \tilde{\boldsymbol{f}} = \boldsymbol{R}^{-1}\boldsymbol{B}^\top \tilde{\boldsymbol{x}}$$

Prove that $\boldsymbol{B}\tilde{\boldsymbol{f}} = \boldsymbol{d}$ and

$$\tilde{\boldsymbol{x}}^\top \boldsymbol{d} - \frac{1}{2}\tilde{\boldsymbol{x}}^\top \boldsymbol{L}\tilde{\boldsymbol{x}} = \frac{1}{2}\tilde{\boldsymbol{f}}^\top \boldsymbol{R}\tilde{\boldsymbol{f}}.$$

(v) Show

$$\tilde{\boldsymbol{x}} \in \arg\max_{\boldsymbol{x} \in \mathbb{R}^V} \boldsymbol{x}^\top \boldsymbol{d} - \frac{1}{2}\boldsymbol{x}^\top \boldsymbol{L}\boldsymbol{x}$$

and

$$\tilde{\boldsymbol{f}} \in \arg\min_{\boldsymbol{f} \in \mathbb{R}^E} \frac{1}{2}\sum_e \boldsymbol{r}(e)\boldsymbol{f}(e)^2 \quad \text{s.t. } \boldsymbol{B}\boldsymbol{f} = \boldsymbol{d}.$$

Exercise 2.

Consider the edge samples as defined in Lecture 1.

- We have a graph $G = (V, E, \boldsymbol{w})$ with $|V| = n$ and $|E| = m$, and with Laplacian $\boldsymbol{L} = \sum_e \boldsymbol{w}(e)\boldsymbol{b}_e\boldsymbol{b}_e^\top$.
- We define a matrix function $\Phi : \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times n}$ by

$$\Phi(\boldsymbol{M}) = \boldsymbol{L}^{+/2}\boldsymbol{M}\boldsymbol{L}^{+/2}.$$

- We introduce a set of independent random matrices $\boldsymbol{Y}_e$, one for each edge e, with a probability $p_e = \min(1, \alpha \left\|\Phi(\boldsymbol{w}(e)\boldsymbol{b}_e\boldsymbol{b}_e^\top)\right\|)$ associated with the edge. We let

$$\boldsymbol{Y}_e = \begin{cases} \frac{\boldsymbol{w}(e)}{p_e}\boldsymbol{b}_e\boldsymbol{b}_e^\top & \text{with probability } p_e \\ \boldsymbol{0} & \text{otherwise.} \end{cases}$$

- This way, $\tilde{\boldsymbol{L}} = \sum_e \boldsymbol{Y}_e$ is our random, hopefully sparse, approximation of $\boldsymbol{L}$. Let $\tilde{G}$ be the graph associated with $\tilde{\boldsymbol{L}}$.
- Let us define

$$\boldsymbol{X}_e = \Phi(\boldsymbol{Y}_e) - \mathbb{E}\left[\Phi(\boldsymbol{Y}_e)\right] \text{ and } \boldsymbol{X} = \sum_e \boldsymbol{X}_e$$

1. Prove that for any two matrices $\boldsymbol{A}, \boldsymbol{B} \succeq \boldsymbol{0}$, $\|\boldsymbol{A} - \boldsymbol{B}\| \le \max(\|\boldsymbol{A}\|, \|\boldsymbol{B}\|)$. (We skipped this step when proving $\|\boldsymbol{X}_e\| \le \frac{1}{\alpha}$).
2. Prove that $\left\|\sum_e \mathbb{E}\left[\boldsymbol{X}_e^2\right]\right\| \le \frac{1}{\alpha}$.
3. Conclude that there is an $\alpha = O(\log(n/\delta))$ s.t. $\tilde{G}$ is a spectral sparsifier of G with probability at least $1 - \delta/2$.
4. Explain how we can use a scalar Chernoff bound to prove that $\left|\tilde{E}\right| \le O(\epsilon^{-2}\log(n/\delta)n)$ with probability at least $1 - \delta/2$. You may pick any constant that suits you to establish the $O(\cdot)$ bound.

Exercise 3.

Recall the Matrix Bernstein theorem from class:

Theorem (A Bernstein Matrix Concentration Bound (Tropp 2011))**.** *Suppose $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_k \in \mathbb{R}^{n\times n}$ are independent, symmetric matrix-valued random variables. Assume each $\boldsymbol{X}_i$ is zero-mean, i.e. $\mathbb{E}[\boldsymbol{X}_i] = \boldsymbol{0}_{n\times n}$, and that $\|\boldsymbol{X}_i\| \le R$ always. Let $\boldsymbol{X} = \sum_i \boldsymbol{X}_i$, and $\sigma^2 = Var[\boldsymbol{X}] = \sum_i \mathbb{E}\left[\boldsymbol{X}_i^2\right]$, then for $\epsilon > 0$*

$$\Pr[\|\boldsymbol{X}\| \ge t] \le 2n \exp\left(\frac{-t^2}{2Rt + 4\sigma^2}\right).$$

Recall that in Lecture 1, we sketched a proof that for all $0 < \theta \leq 1/R$,

$$\Pr[\|\boldsymbol{X}\| \geq t] \leq 2\exp(-\theta t)\operatorname{Tr}\exp\left(\theta^2\sigma^2\boldsymbol{I}\right).$$

1. Prove that $\log(\exp(\theta\,\mathbb{E}[\boldsymbol{X}_i])) \preceq \theta^2\,\mathbb{E}\left[\boldsymbol{X}_i^2\right]$, when $0 < \theta \leq 1/R$.
2. Finish the proof of Matrix Bernstein by finding a suitable choice of θ.

Exercise 4

In Lecture 1, we discussed a lemma which allows us to prove that a cut sparsifier of an expander is also a spectral sparsifier of that expander.

Lemma (Cut Approximation Implies Spectral Approximation in Expanders)**.** *Suppose $G = (V, E, \boldsymbol{w})$ is a ϕ-expander. Let H be a K-factor cut approximator of G, i.e.*

$$\frac{1}{(K+1)}G \leq_{cut} H \leq_{cut} (K+1)G.$$

Then $H \approx_{\operatorname{poly}(K\phi^{-1})} G$, i.e. H is also a spectral approximation of G.

- Prove that when H and G have $\operatorname{wdeg}_G(u) = \operatorname{wdeg}_H(u)$ for all $u \in V$ then $H \preceq \operatorname{poly}(K\phi^{-1})G$. *Hint: Use expansion and Cheeger's inequality. Be careful about the kernels of the matrices involved.*
- Prove the full lemma. *Hint: you can add self-loops to enforce degrees being equal. Cheeger's inequality applies to graphs with self-loops.*

Exercise 5

Consider an unweighted graph $G = (V, E)$ with

1. Use the Expander Decomposition Theorem by Saranurak and Wang to decompose the graph into (overlapping) expanders with expansion $\phi = 1/\operatorname{polylog} n$.
2. Sparsify the expanders using Benczur-Karger cut sparsification. Note the output is a weighted graph.
3. Use the sparsified expanders to construct a (weighted) graph $H = (V, F, \boldsymbol{w}_H)$ s.t. $|F| = |V|\operatorname{polylog}(|V|)$ and $H \approx_{\operatorname{polylog} n} G$.
4. Bonus: Can you extend the approach to weighted graphs where the weights lie between 1 and $\operatorname{poly}(|V|)$?

Bonus Questions

For those of you who want more, here are some extra questions. I probably won't have time to discuss how to solve them.

Exercise 6.

The bound obtained in Cheeger's inequality is indeed tight. Prove that:

1. Let G be the graph consisting of two vertices connected by a single edge of unit weight. Prove that $\phi(G) = \lambda_2(\boldsymbol{N})/2$ and therefore that the lower bound of Cheeger's inequality is tight.
2. To show that the line graph proves that the upper bound of Cheeger's Inequality is asymptotically tight (i.e. up to constant factors).

Exercise 7.

Sparse Expanders: In random graph theory, the graph over n vertices where each edge between two endpoints is present independently with probability p is denoted $G(n, p)$.

Show that for $p = \Omega(\log n/n)$, that $G(n, p)$ is a $\Omega(1)$-expander with high probability (it is up to you to fix large constants). Take the following steps:

1. Prove that with high probability, $\deg(u) = \Theta(pn)$ for all vertices $u \in V(G(n, p))$.
2. For each set S of $k \leq n/2$ vertices, argue that

$$\mathbb{P}[|E(S, V \setminus S)| = \Theta(kpn)] > 1 - n^{-c \cdot k}$$

   for any large constant $c > 0$.
3. Observe that there are at most $\binom{n}{k}$ sets of vertices S of size k. Conclude that $G(n, p)$ is with high probability a $\Omega(1)$-expander.

Reading list

The following is a very haphazard of list of papers that I mentioned during class. It is very much against the advice of *none mentioned, none forgotten.*

- My course notes: `http://kyng.inf.ethz.ch/courses/AGAO21/agao21_script.pdf`
- Code: `https://github.com/danspielman/Laplacians.jl/`
- Solving Laplacian linear equations: [ST04, KS16, JS21] and many more.
- Graph sparsification: [BK96, SS11].
- Matrix concentration: [Tro12].
- Semi-supervised learning using graphs: [ZGL03].
- Structured linear equation solvers beyond Laplacians: [DS07, DS08, KLP+16, CKP+16b, CKP+17, CKK+18, KPSZ18, AJSS19].
- Scalar elliptic partial differential equations: [BHV08].
- Maximum flow: [DS08, CKM+10, Mad13, Mad16, KLS20].
- Minimum cost flow: [LS14, CMSV17, AMV20, vdBLL+21].
- Fine-grained complexity for spectral graph theory: [KZ20], `http://rasmuskyng.com/papers/LPto2CF.pdf`.
- Heuristic solvers for Laplacians: [Mv77, BD79, Bra00].
- Expander decomposition: [ST04, SW19].

References

[AJSS19] AmirMahdi Ahmadinejad, Arun Jambulapati, Amin Saberi, and Aaron Sidford. Perron-frobenius theory in nearly linear time: Positive eigenvectors, m-matrices, graph kernels, and other applications. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1387–1404. SIAM, 2019.

[AMV20] Kyriakos Axiotis, Aleksander Madry, and Adrian Vladu. Circulation Control for Faster Minimum Cost Flow in Unit-Capacity Graphs. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 93–104. IEEE Computer Society, November 2020.

[BD79] Achi Brandt and Nathan Dinar. Multigrid Solutions to Elliptic Flow Problems. In SEYMOUR V. Parter, editor, *Numerical Methods for Partial Differential Equations*, pages 53–147. Academic Press, January 1979.

[BHV08] Erik G. Boman, Bruce Hendrickson, and Stephen Vavasis. Solving Elliptic Finite Element Systems in Near-Linear Time with Support Preconditioners. *SIAM J. Numer. Anal.*, 46(6):3264–3284, October 2008.

[BK96] András A. Benczúr and David R. Karger. Approximating *s*-*t* minimum cuts in $\tilde{O}$ (n ^{2}) time. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing - STOC '96*, pages 47–55, Philadelphia, Pennsylvania, United States, 1996. ACM Press.

[Bra00] Achi Brandt. General highly accurate algebraic coarsening. *Electronic Transactions on Numerical Analysis*, 10(1):21, 2000.

[CKK$^+$18] Michael B. Cohen, Jonathan Kelner, Rasmus Kyng, John Peebles, Richard Peng, Anup B. Rao, and Aaron Sidford. Solving directed laplacian systems in nearly-linear time through sparse LU factorizations. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 898–909. IEEE, 2018.

[CKM$^+$10] Paul Christiano, Jonathan A. Kelner, Aleksander Madry, Daniel A. Spielman, and Shang-Hua Teng. Electrical Flows, Laplacian Systems, and Faster Approximation of Maximum Flow in Undirected Graphs. *arXiv:1010.2921 [cs]*, October 2010.

[CKP$^+$16a] Michael B. Cohen, Jonathan Kelner, John Peebles, Richard Peng, Anup Rao, Aaron Sidford, and Adrian Vladu. Almost-Linear-Time Algorithms for Markov Chains and New Spectral Primitives for Directed Graphs. *arXiv:1611.00755 [cs]*, November 2016.

[CKP$^+$16b] Michael B. Cohen, Jonathan Kelner, John Peebles, Richard Peng, Aaron Sidford, and Adrian Vladu. Faster algorithms for computing the stationary distribution, simulating random walks, and more. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 583–592. IEEE, 2016.

[CKP$^+$17] Michael B. Cohen, Jonathan Kelner, John Peebles, Richard Peng, Anup B. Rao, Aaron Sidford, and Adrian Vladu. Almost-linear-time algorithms for markov chains and new spectral primitives for directed graphs. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 410–419, 2017.

[CMSV17] Michael B. Cohen, Aleksander Madry, Piotr Sankowski, and Adrian Vladu. Negative-weight shortest paths and unit capacity minimum cost flow in õ (m 10/7 log w) time*. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 752–771. SIAM, 2017.

[DS07] Samuel I. Daitch and Daniel A. Spielman. Support-Graph Preconditioners for 2-Dimensional Trusses. *arXiv:cs/0703119*, March 2007.

[DS08] Samuel I. Daitch and Daniel A. Spielman. Faster approximate lossy generalized flow via interior point algorithms. In *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*, STOC '08, pages 451–460, New York, NY, USA, May 2008. Association for Computing Machinery.

[JS21] Arun Jambulapati and Aaron Sidford. Ultrasparse Ultrasparsifiers and Faster Laplacian System Solvers. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 540–559. SIAM, 2021.

[KLP$^+$16] Rasmus Kyng, Yin Tat Lee, Richard Peng, Sushant Sachdeva, and Daniel A. Spielman. Sparsified cholesky and multigrid solvers for connection laplacians. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*, pages 842–850, 2016.

[KLS20] Tarun Kathuria, Yang P. Liu, and Aaron Sidford. Unit Capacity Maxflow in Almost $ O (m$̂{$4/3$}$) $ Time. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 119–130. IEEE, 2020.

[KPSZ18] Rasmus Kyng, Richard Peng, Robert Schwieterman, and Peng Zhang. Incomplete nested dissection. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 404–417, 2018.

[KS16] Rasmus Kyng and Sushant Sachdeva. Approximate gaussian elimination for laplacians-fast, sparse, and simple. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 573–582. IEEE, 2016.

[KZ20] Rasmus Kyng and Peng Zhang. Hardness results for structured linear systems. *SIAM Journal on Computing*, 49(4):FOCS17–280, 2020.

[LS14] Yin Tat Lee and Aaron Sidford. Path finding methods for linear programming: Solving linear programs in o (vrank) iterations and faster algorithms for maximum flow. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 424–433. IEEE, 2014.

[Mad13] Aleksander Madry. Navigating central path with electrical flows: From flows to matchings, and back. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 253–262. IEEE, 2013.

[Mad16] Aleksander Madry. Computing maximum flow with augmenting electrical flows. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 593–602. IEEE, 2016.

[Mv77] J. A. Meijerink and H. A. van der Vorst. An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix is a Symmetric M-Matrix. *Mathematics of Computation*, 31(137):148–162, 1977.

[RS87] John W. Ruge and Klaus Stüben. Algebraic multigrid. In *Multigrid Methods*, pages 73–130. SIAM, 1987.

[SS11] Daniel A. Spielman and Nikhil Srivastava. Graph sparsification by effective resistances. *SIAM Journal on Computing*, 40(6):1913–1926, 2011.

[ST04] Daniel A. Spielman and Shang-Hua Teng. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '04, pages 81–90, New York, NY, USA, June 2004. Association for Computing Machinery.

[SW19] Thatchaphol Saranurak and Di Wang. Expander Decomposition and Pruning: Faster, Stronger, and Simpler. *arXiv:1812.08958 [cs]*, February 2019.

[Tro12] Joel A. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, August 2012.

[vdBLL+21] Jan van den Brand, Yin Tat Lee, Yang P. Liu, Thatchaphol Saranurak, Aaron Sidford, Zhao Song, and Di Wang. Minimum Cost Flows, MDPs, and $$\backslash$ell_1 $-Regression in Nearly Linear Time for Dense Instances. *arXiv preprint arXiv:2101.05719*, 2021.

[ZGL03] Xiaojin Zhu, Zoubin Ghahramani, and John D. Lafferty. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 912–919, 2003.