Spectral Graph Theory Lecture 21

Sparsification by Random Sampling

Daniel A. Spielman November 12, 2018

21.1 Overview

Two weeks ago, we learned that expander graphs are sparse approximations of the complete graph. This week we will learn that every graph can be approximated by a sparse graph. Today, we will see how a sparse approximation can be obtained by careful random sampling: every graph on n vertices has an ϵ-approximation with only $O(\epsilon^{-2} n \log n)$ edges (a result of myself and Srivastava [SS11]). We will prove this using a matrix Chernoff bound due to Tropp [Tro12].

We originally proved this theorem using a concentration bound of Rudelson [Rud99]. This required an argument that used sampling with replacement. When I taught this result in 2012, I asked if one could avoid sampling with replacement. Nick Harvey pointed out to me the argument that avoids replacement that I am presenting today.

In the next lecture, we will see that the $\log n$ term is unnecessary. In fact, almost every graph can be approximated by a sparse graph almost as well as the Ramanujan graphs approximate complete graphs.

21.2 Sparsification

For this lecture, I define a graph H to be an ϵ-approximation of a graph G if

$$(1-\epsilon)\boldsymbol{L}_G \preccurlyeq \boldsymbol{L}_H \preccurlyeq (1+\epsilon)\boldsymbol{L}_G.$$

We will show that every graph G has a good approximation by a sparse graph. This is a very strong statement, as graphs that approximate each other have a lot in common. For example,

1. the effective resistance between all pairs of vertices are similar in the two graphs,
2. the eigenvalues of the graphs are similar,
3. the boundaries of all sets are similar, as these are given by $\boldsymbol{\chi}_S^T \boldsymbol{L}_G \boldsymbol{\chi}_S$, and
4. the solutions of linear equations in the two matrices are similar.

We will prove this by using a very simple random construction. We first carefully[1] choose a probability $p_{a,b}$ for each edge (a,b). We then include each edge (a,b) with probabilty $p_{a,b}$, independently.

[1] For those who can't stand the suspense, we reveal that we will choose the probabilities to be proportional to leverage scores of the edges.

If we do include edge (a, b), we give it weight $w_{a,b}/p_{a,b}$. We will show that our choice of probabilities ensures that the resulting graph H has at most $4n \ln n/\epsilon^2$ edges and is an ϵ approximation of G with high probability.

The reason we employ this sort of sampling–blowing up the weight of an edge by dividing by the probability that we choose it—is that it preserves the matrix in expectation. Let $\boldsymbol{L}_{a,b}$ denote the elementary Laplacian on edge (a, b) with weight 1, so that

$$\boldsymbol{L}_G = \sum_{(a,b) \in E} w_{a,b} \boldsymbol{L}_{a,b}.$$

We then have that

$$\mathbb{E} \boldsymbol{L}_H = \sum_{(a,b) \in E} p_{a,b} (w_{a,b}/p_{a,b}) \boldsymbol{L}_{a,b} = \boldsymbol{L}_G.$$

21.3 Matrix Chernoff Bounds

The main tool that we will use in our analysis is a theorem about the concentration of random matrices. These may be viewed as matrix analogs of the Chernoff bound that we saw in Lecture 5. These are a surprisingly recent development, with the first ones appearing in the work of Rudelson and Vershynin [Rud99, RV07] and Ahlswede and Winter [AW02]. The best present source for these bounds is Tropp [Tro12], in which the following result appears as Corollary 5.2.

Theorem 21.3.1. *Let $\boldsymbol{X}_1, \dots, \boldsymbol{X}_m$ be independent random n-dimensional symmetric positive semidefinite matrices so that $\|\boldsymbol{X}_i\| \leq R$ almost surely. Let $\boldsymbol{X} = \sum_i \boldsymbol{X}_i$ and let μ_{min} and μ_{max} be the minimum and maximum eigenvalues of*

$$\mathbb{E}\left[\boldsymbol{X}\right] = \sum_i \mathbb{E}\left[\boldsymbol{X}_i\right].$$

Then,

$$Pr\left[\lambda_{min}(\sum_i \boldsymbol{X}_i) \leq (1-\epsilon)\mu_{min}\right] \leq n \left(\frac{e^{-\epsilon}}{(1-\epsilon)^{1-\epsilon}}\right)^{\mu_{min}/R}, \qquad \textit{for } 0 < \epsilon < 1, \textit{ and}$$

$$Pr\left[\lambda_{max}(\sum_i \boldsymbol{X}_i) \geq (1+\epsilon)\mu_{max}\right] \leq n \left(\frac{e^{\epsilon}}{(1+\epsilon)^{1+\epsilon}}\right)^{\mu_{max}/R}, \qquad \textit{for } 0 < \epsilon.$$

It is important to note that the matrices $\boldsymbol{X}_1, \dots, \boldsymbol{X}_m$ can have different distributions. Also note that as the norms of these matrices get bigger, the bounds above become weaker. As the expressions above are not particularly easy to work with, we often use the following approximations.

$$\left(\frac{e^{-\epsilon}}{(1-\epsilon)^{1-\epsilon}}\right) \leq e^{-\epsilon^2/2}, \qquad \text{for } 0 < \epsilon < 1, \text{ and}$$

$$\left(\frac{e^{\epsilon}}{(1+\epsilon)^{1+\epsilon}}\right) \leq e^{-\epsilon^2/3}, \qquad \text{for } 0 < \epsilon < 1.$$

Chernoff (and Hoeffding and Bernstein) bounds rarely come in exactly the form you want. Sometimes you can massage them into the needed form. Sometimes you need to prove your own. For this reason, you may some day want to spend a lot of time reading how these are proved.

21.4 The key transformation

Before applying the matrix Chernoff bound, we make a transformation that will cause $\mu_{min} = \mu_{\max} = 1$.

For positive definite matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, we have

$$\boldsymbol{A} \preccurlyeq (1+\epsilon)\boldsymbol{B} \quad \Longleftrightarrow \quad \boldsymbol{B}^{-1/2}\boldsymbol{A}\boldsymbol{B}^{-1/2} \preccurlyeq (1+\epsilon)\boldsymbol{I}.$$

The same things holds for singular semidefinte matrices that have the same nullspace:

$$\boldsymbol{L}_H \preccurlyeq (1+\epsilon)\boldsymbol{L}_G \quad \Longleftrightarrow \quad \boldsymbol{L}_G^{+/2}\boldsymbol{L}_H\boldsymbol{L}_G^{+/2} \preccurlyeq (1+\epsilon)\boldsymbol{L}_G^{+/2}\boldsymbol{L}_G\boldsymbol{L}_G^{+/2},$$

where $\boldsymbol{L}_G^{+/2}$ is the square root of the pseudo-inverse of $\boldsymbol{L}_G$. Let

$$\boldsymbol{\Pi} = \boldsymbol{L}_G^{+/2}\boldsymbol{L}_G\boldsymbol{L}_G^{+/2},$$

which is the projection onto the range of $\boldsymbol{L}_G$. We now know that $\boldsymbol{L}_G$ is an ϵ-approximation of $\boldsymbol{L}_H$ if and only if $\boldsymbol{L}_G^{+/2}\boldsymbol{L}_H\boldsymbol{L}_G^{+/2}$ is an ϵ-approximation of $\boldsymbol{\Pi}$.

As multiplication by a fixed matrix is a linear operation and expectation commutes with linear operations,

$$\mathbb{E}\boldsymbol{L}_G^{+/2}\boldsymbol{L}_H\boldsymbol{L}_G^{+/2} = \boldsymbol{L}_G^{+/2}\left(\mathbb{E}\boldsymbol{L}_H\right)\boldsymbol{L}_G^{+/2} = \mathbb{E}\boldsymbol{L}_G^{+/2}\boldsymbol{L}_G\boldsymbol{L}_G^{+/2} = \boldsymbol{\Pi}.$$

So, we really just need to show that this random matrix is probably close to its expectation, $\boldsymbol{\Pi}$. It would probably help to pretend that $\boldsymbol{\Pi}$ is in fact the identity, as it will make it easier to understand the analysis. In fact, you don't have to pretend: you could project all the vectors and matrices onto the span of $\boldsymbol{\Pi}$ and carry out the analysis there.

21.5 The probabilities

Let

$$\boldsymbol{X}_{a,b} = \begin{cases} (w_{a,b}/p_{a,b})\boldsymbol{L}_G^{+/2}\boldsymbol{L}_{(a,b)}\boldsymbol{L}_G^{+/2} & \text{with probability } p_{a,b} \\ 0 & \text{otherwise,} \end{cases}$$

so that

$$\boldsymbol{L}_G^{+/2}\boldsymbol{L}_H\boldsymbol{L}_G^{+/2} = \sum_{(a,b)\in E} \boldsymbol{X}_{a,b}.$$

We will choose the probabilities to be

$$p_{a,b} \stackrel{\text{def}}{=} \frac{1}{R} w_{a,b} \left\| \boldsymbol{L}_G^{+/2}\boldsymbol{L}_{(a,b)}\boldsymbol{L}_G^{+/2} \right\|,$$

for an R to be chosen later. Thus, when edge (a, b) is chosen, $\|X_{a,b}\| = R$. Making this value uniform for every edge optimizes one part of Theorem 21.3.1.

You may wonder what we should do if one of these probabilities $p_{a,b}$ exceeds one. There are many ways of addressing this issue. For now, pretend that it does not happen. We will then explain how to deal with this at the end of lecture.

Recall that the leverage score of edge (a, b) written $\ell_{a,b}$ was defined in Lecture 14 to be the weight of an edge times the effective resistance between its endpoints:

$$\ell_{a,b} = w_{a,b}(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \boldsymbol{L}_G^{+}(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b).$$

To see the relation between the leverage score and $p_{a,b}$, compute

$$\begin{aligned}\left\|\boldsymbol{L}_G^{+/2}\boldsymbol{L}_{(a,b)}\boldsymbol{L}_G^{+/2}\right\| &= \left\|\boldsymbol{L}_G^{+/2}(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \boldsymbol{L}_G^{+/2}\right\| \\ &= \left\|(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \boldsymbol{L}_G^{+/2}\boldsymbol{L}_G^{+/2}(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)\right\| \\ &= (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \boldsymbol{L}_G^{+}(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) \\ &= \mathrm{R_{eff}}(a, b).\end{aligned}$$

As we can quickly approximate the effective resistance of every edge, we can quickly compute sufficient probabilities.

Recall that the leverage score of an edge equals the probability that the edge appears in a random spanning tree. As every spanning tree has $n - 1$ edges, this means that the sum of the leverage scores is $n - 1$, and thus

$$\sum_{(a,b)\in E} p_{a,b} = \frac{n-1}{R} \leq \frac{n}{R}.$$

This is a very clean bound on the expected number of edges in H. One can use a Chernoff bound (on real variables rather than matrices) to prove that it is exponentially unlikely that the number of edges in H is more than any small multiple of this.

For your convenience, I recall another proof that the sum of the leverage scores is $n-1$:

$$\begin{aligned}
\sum_{(a,b)\in E} \ell_{a,b} &= \sum_{(a,b)\in E} w_{a,b} \mathrm{R}_{\mathrm{eff}}(a,b) \\
&= \sum_{(a,b)\in E} w_{a,b} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \boldsymbol{L}_G^{+} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b) \\
&= \sum_{(a,b)\in E} w_{a,b} \mathrm{Tr}\left(\boldsymbol{L}_G^{+} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \right) \\
&= \mathrm{Tr}\left(\sum_{(a,b)\in E} \boldsymbol{L}_G^{+} w_{a,b} (\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)(\boldsymbol{\delta}_a - \boldsymbol{\delta}_b)^T \right) \\
&= \mathrm{Tr}\left(\boldsymbol{L}_G^{+} \sum_{(a,b)\in E} w_{a,b} \boldsymbol{L}_{a,b} \right) \\
&= \mathrm{Tr}\left(\boldsymbol{L}_G^{+} \boldsymbol{L}_G \right) \\
&= \mathrm{Tr}\left(\boldsymbol{\Pi} \right) \\
&= n - 1.
\end{aligned}$$

21.6 The analysis

We will choose

$$R = \frac{\epsilon^2}{3.5 \ln n}.$$

Thus, the number of edges in H will be at most $4(\ln n)\epsilon^{-2}$ with high probability.

We have

$$\sum_{(a,b)\in E} \mathbb{E} \boldsymbol{X}_{a,b} = \boldsymbol{\Pi}.$$

It remains to show that it is unlikely to deviate from this by too much.

We first consider the case in which $p_{(a,b)} \leq 1$ for all edges (a,b). If this is the case, then Theorem 21.3.1 tells us that

$$\Pr\left[\sum_{a,b} \boldsymbol{X}_{a,b} \geq (1+\epsilon)\boldsymbol{\Pi} \right] \leq n \exp\left(-\epsilon^2/3R\right) = n \exp\left(-(3.5/3)\ln n\right) = n^{-1/6}.$$

For the lower bound, we need to remember that we can just work orthogonal to the all-1s vector, and so treat the smallest eigenvalue of $\boldsymbol{\Pi}$ as 1. We then find that

$$\Pr\left[\sum_{a,b} \boldsymbol{X}_{a,b} \leq (1-\epsilon)\boldsymbol{\Pi} \right] \leq n \exp\left(-\epsilon^2/2R\right) = n \exp\left(-(3.5/2)\ln n\right) = n^{-3/2},$$

We finally return to deal with the fact that there might be some edges for which $p_{a,b} \geq 1$ and so definitely appear in H. There are two natural ways to deal with these—one that is easiest algorithmically and one that simplifies the proof. The algorithmically natural way to handle these is to simply include these edges in H, and remove them from the analysis above. This requires a small adjustment to the application of the Matrix Chernoff bound, but it does go through.

From the perspective of the proof, the simplest way to deal with these is to split each such $\boldsymbol{X}_{a,b}$ into many independent random edges: $k = \lfloor \ell_{a,b}/R \rfloor$ that appear with probability exactly 1, and one more that appears with probability $\ell_{a,b}/R - k$. This does not change the expectation of their sum, or the expected number of edges once we remember to add together the weights of edges that appear multiple times. The rest of the proof remains unchanged.

21.7 Open Problem

If I have time in class, I will sketch a way to quickly approximate the effective resistances of every edge in the graph. The basic idea, which can be found in [SS11] and which is carried out better in [KLP12], is that we can compute the effective resistance of an edge (a, b) from the solution to a logarithmic number of systems of random linear equations in $\boldsymbol{L}_G$. That is, after solving a logarithmic number of systems of linear equations in $\boldsymbol{L}_G$, we have information from which we can estimates all of the effective resistances.

In order to sparsify graphs, we do not actually need estimates of effective resistances that are always accurate. We just need a way to identify many edges of low effective resistance, without listing any that have high effective resistance. I believe that better algorithms for doing this remain to be found. Current fast algorithms that make progress in this direction and that exploit such estimates may be found in [KLP12, Kou14, CLM$^+$15, LPS15]. These, however, rely on fast Laplacian equation solvers. It would be nice to be able to estimate effective resistances without these. A step in this direction was recently taken in the works [CGP$^+$18, LSY18], which quickly decompose graphs into the union of short cycles plus a few edges.

References

[AW02] R. Ahlswede and A. Winter. Strong converse for identification via quantum channels. *Information Theory, IEEE Transactions on*, 48(3):569–579, 2002.

[CGP$^+$18] Timothy Chu, Yu Gao, Richard Peng, Sushant Sachdeva, Saurabh Sawlani, and Junxing Wang. Graph sparsification, spectral sketches, and faster resistance computation, via short cycle decompositions. *arXiv preprint arXiv:1805.12051*, 2018.

[CLM$^+$15] Michael B Cohen, Yin Tat Lee, Cameron Musco, Christopher Musco, Richard Peng, and Aaron Sidford. Uniform sampling for matrix approximation. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*, pages 181–190. ACM, 2015.

[KLP12] Ioannis Koutis, Alex Levin, and Richard Peng. Improved spectral sparsification and numerical algorithms for sdd matrices. In *STACS'12 (29th Symposium on Theoretical Aspects of Computer Science)*, volume 14, pages 266–277. LIPIcs, 2012.

[Kou14] Ioannis Koutis. Simple parallel and distributed algorithms for spectral graph sparsification. In *Proceedings of the 26th ACM Symposium on Parallelism in Algorithms and Architectures*, SPAA '14, pages 61–66, New York, NY, USA, 2014. ACM.

[LPS15] Yin Tat Lee, Richard Peng, and Daniel A. Spielman. Sparsified cholesky solvers for SDD linear systems. *CoRR*, abs/1506.08204, 2015.

[LSY18] Yang P Liu, Sushant Sachdeva, and Zejun Yu. Short cycles via low-diameter decompositions. *arXiv preprint arXiv:1810.05143*, 2018.

[Rud99] M. Rudelson. Random vectors in the isotropic position,. *Journal of Functional Analysis*, 164(1):60 – 72, 1999.

[RV07] Mark Rudelson and Roman Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *J. ACM*, 54(4):21, 2007.

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

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