COMS W6998-5: Algorithms through Geometric Lens (Fall'17) Dec 4, 2017

Lecture 24 – Review

Instructors: *Alex Andoni, Ilya Razenshteyn* Scribes: *Daniel Speyer*

1 Presentations

Will send sign-up right after lecture (already sent).

Wednesday: 5 presentations

Monday: 7 presentation (very tight)

If more than 7 request Monday, there will be a lottery. Wednesday will have lower standards, and leave more time for write-up, which is where most of the grade is.

Presentations last about 10 minutes. Which isn't much.

Making things shorter is much harder than making them long. Being concise is an art, and it takes effort.

Focus on a paper or a couple of papers.

Include:

- Problem Statement
- Context/Background (quick) Is this a new method? Was there nothing before? Was the old way slower?
- Main New Idea in the **simplest** possible setting. For example, simple graphs have fixed degree. Simple LSH is on the sphere (if they're not, reducing them to the sphere is already in the literature, so that's not what a new paper is about). A paper usually has one big idea.
- What is achieved (new bound?)
- What is done that is new (new way of looking at eigenvalues?)

Report should be about ten pages. You will find this is short. You will need to throw things out (but still say something meaningful!).

Will send more recommendations on final write-up.

2 Sketching/Streaming

Objects arrive in a stream; we have low space.

Frequency vector $x \in \mathbb{R}^d$ (d types of objects). If we had x, life would be easy, but we don't have $O(d)$ storage to put it in.

Linear random map $f : \mathbb{R}^d \to \mathbb{R}^k$ with $k << d$.

$||f(x)||_2 \underset{(1\pm\epsilon)}{\approx} ||x||_2$ with $p > 0.9$ using $k = O(1/\epsilon^2)$

Linear, so $f(x) = Gx$ where $G = \{\pm 1/\sqrt{k}\}^{n \times k}$ random.

Can also get ℓ_1 norm, and other things.

2.1 Heavy Hitters

Defined as $|x_j| \geq \phi ||x||_1$: Use "Count Sketch"

2.2 ℓ_0 sampling

Report j s.t. $x_j \neq 0$ with equal probability for all acceptable js.

Applies to Dynamic Graph Connectivity: Get a stream of updates about a graph (add edge, remove edge). Store something (not the whole graph) that allows a good estimate of whether two points are connected.

3 Dimension Reduction

JL (stands for "Johnson-Lindenstrauss"):

$f(x) = Gx$ where $G \sim \frac{1}{\sqrt{k}} \mathcal{N}(0,1)^{d \times k}$ such that $||f(x)||_2 \underset{(1 \pm \epsilon)}{\approx} ||x||_2$ with $p > 1 - \delta$ if $k = \Theta(-\log(\delta)/\epsilon^2)$.

But cubic time.

Fast JL: $G = PHD$ where H is Hadamard, D is diagonal of ± 1 and P is selection. $k = O(-\log(\delta)\log(d/\delta)/\epsilon^2)$ in nlogn time.

Application: quick numeric optimization:

$\arg\min_{x \in \mathbb{R}^d} ||Ax - b||_2$ where A is $n \times d$ and $n \gg d$.

Reduce to $\arg\min_{x \in \mathbb{R}^d} ||GAx - Gb||_2$ where G is $k \times n$ and $k = O(d/\epsilon^2)$ (for a $(1 \pm \epsilon)$ solution).

Lots of similar applications

4 Near Neighbor

Take points $P \subset \mathbb{R}^d$ with ℓ_2 metric (or any set and metric we like) and preprocess into a data structure of our choice.

Get a query point q s.t. $\exists p \in P ||q - p|| < r$, find $p^* \in P$ s.t. $||q - p^*|| < cr$. Where r and c are constants.

You could dimension reduce and use k-d trees, but we'd have to reduce a lot before those got good expected performance, and they have no theoretical guarentees. A better option is LSH.

$f : \mathbb{R}^d \to U$ where U is discrete (small integers, binary strings, whatever).

$$\begin{aligned} ||x - y|| < r &\implies Pr(f(p) = f(q)) > p_1 \\ ||x - y|| > cr &\implies Pr(f(p) = f(q)) < p_2 \end{aligned}$$

$|U| < 1/p_2$ will suffice.

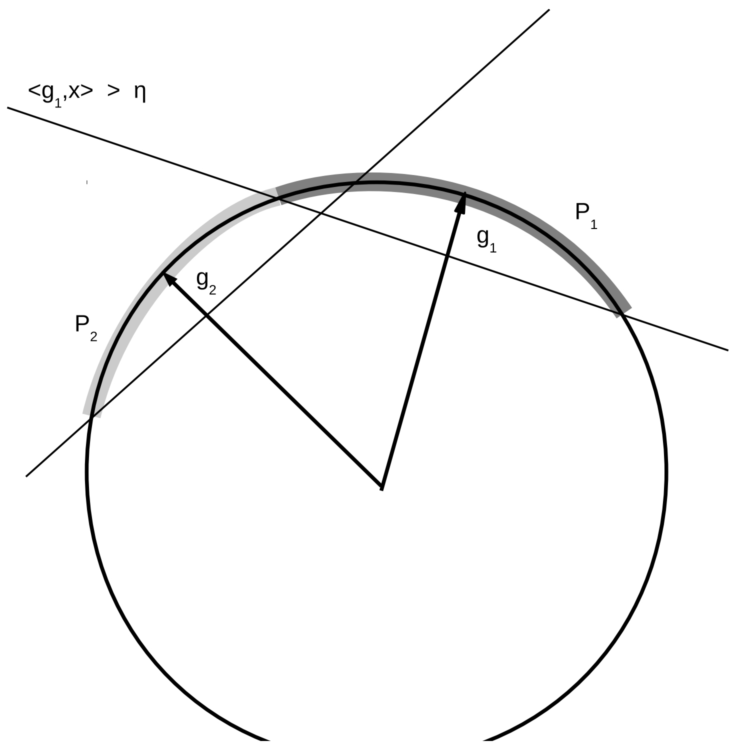
You can think of it as dimension reduction, to a discrete set instead of a space.

4.1 LSH Functions

Hamming space $(0, 1^d, \ell_1)$, f is sample a random dimension. That's a dimension reduction.

Euclidean space $(\{x \in \mathbb{R}^d s.t. ||x||_2 = 1\}, \ell_2)$, (could drop the sphere restriction, but easier to keep it) two options:

- Random Hyperplane: $\text{sign}(\langle x, g \rangle)$ where $g \sim \mathcal{N}(0, I)$. Slash the entire space with a hyperplane (perpendicular to g) and check which side the points fall on. Simple, but not optimal.
- Cap Carving, a.k.a. Ball Carving in cosine metric. min i s.t. $\langle x, g_i \rangle > \eta$ where all gs drawn from $N(0, I)$. Optimal, for the sphere, using the correct η (nontrivial). Becomes data-dependant for more.



It takes a lot more gs in high dimension.

With high probability, this algorithm will finish (even though we defined an infinite series of gs).

5 Graph Algorithms

5.1 SDP=Semi-Definite Programming

Unknown vectors $x_1, ..., x_n \in \mathbb{R}^n$.

Take the matrix such that cell i, j is $\langle x_i, x_j \rangle$.

Constraints and Target: linear on the cells of the matrix.

Can optimize in polynomial time.

Techniques left for another course

5.2 Max-Cut

Separate a graph into two pieces such that as many links are cut as possible.

Assign a value $x = \pm 1$ (a.k.a. $x^2 = 1$) to each node such that nodes in the same side of the cut get the same x. Now maximize $x_i \neq x_j$ (a.k.a. $1 - x_i x_j$) where i, j is an edge.

Equivalently, $\max_x \sum_{(i,j) \in E} (1 - x_i x_j)/2$ s.t. $||x_i||^2 = 1$.

Relax: let x be a vector. We will get a sum better than the optimal coloring.

Round: turn these vectors back to ± 1, so that we can cut the graph apart. Just use $\text{sign}(\langle x, g \rangle)$ where $g \sim \mathcal{N}(0, I)$.

Introducing the vectors is counter-intuitive, but they give a continuous domain to optimize on.

5.3 Coloring

We know the graph is k colorable. Color in k_2 colors in poly time where $k_2 > k$ but hopefully not $k_2 \gg k$ ($k_2 = k$ would be NP hard).

Again, turn into SDP. Round using ball-carving.

6 Spectral Graph Algorithms

$A =$ Adjanceny
$D =$ Diagnonal matrix of degrees
$L = D - A$ ("Laplacian")

$\hat{A}$ and $\hat{L}$ are normalized by dividing by $\sqrt{\delta_i \delta_j}$ (the diagonal becomes 1).

Eigenvalues of $\hat{L}$: $\mu_1 = 0$. $\mu_2 = 0$ iff disconnected. μ_2 small means low conductance (i.e. almost disconnected).

Conductance: $\phi(G) = min_{S \neq \emptyset, V} \frac{|E_{S,\bar{S}}|}{vol(s) vol(V \backslash s)}$

$\mu_2 \leq \phi(G) \leq \sqrt{2\mu_2}$

We can find the cut by looking at the second eigenvector. Recall that the vector assigns a number to each graph node. Cut the line somewhere. Find that somewhere by drawing from a relevant distribution, check if it works (high probabiltiy right away.

Similarly, can partition into k parts with few cuts ($\phi_k(G)$). Take eigenvectors $2 \ldots k$. Transpose the matrix to get n vectors of dimension $k - 1$. Use ball carving to partition those vectors into k parts.

7 Metric Embeddings

Map $S, dist$ (where S is some set and $dist$ is some valid distance metric) to $\mathbb{R}^d, \ell_p$ with low distortion D (i.e. $dist(x, y) \leq ||f(x) - f(y)||_p \leq D dist(x, y)$).

Note that dimension reduction is a special case where $S = \mathbb{R}^n$ with $n \gg d$ and $dist = \ell_p$.

If you're working in some inconvenient metric, try embedding in a convenient one like ℓ_2 with acceptable distortion. This is the best known nearest-neighbor approach for string edit or earth-mover distance.

Any n points with a metric can be put in ℓ_1 with $D = O(\log(n))$.

7.1 Padded Decomposition

Assume we start with a finite set of points $X \subset \mathbb{R}^d$ and wish to randomly partition $\mathbb{R}^d$ such that:

- maximum diameter of a part is constant Δ
- for any point $x \in (X \cap P)$ (where P is a part), $Pr[B(x, \Delta/8) \subset P] \geq p$ where p is high enough.

Can do this with ball carving.

Can be used for distance oracles: given a graph, build a data structure in reasonable space such that distance between any points can be estimated quickly.

Do PDs at varying Δs. Space $O(n^{1+1/k})$ for distortion $O(k)$ and query time $O(1)$. Or $O(2k-1)$ distortion and $O(k)$ time. Can do better.

8 Discrepency

Given sets $S_1, S_2, ..., S_n \subset [n]$ assign $x_1, ...x_n \in \pm 1$ (note that n is both the number of points and number of sets, because that's simpler) such that $|\sum_{i \in S_j} x_i| \leq O(n)$.

9 General Lessons to Take Home

You can have a much better time in a nice metric than an arbitrary one.

But ball carving often works anyway.

10 Courseworks Evaluation

Please fill out coursework evaluation! Leave blank redundant questions.