princeton u. sp'02 cos 598B: algorithms and complexity

Lecture 3-4: Embedding metrics into ℓ_1 and applications to sparsest cut

Lecturer: *Sanjeev Arora* Scribe:*Elad Hazan*

1 Bourgain's theorem (ℓ_1 version)

The main goal of these lectures is to prove the following theorem, which is a special case of Bourgain's theorem from the mid-eighties.

Theorem 1 (Bourgain, 1985)
Every metric space embeds into ℓ_1 with distortion $O(\log n)$.

Bourgain's theorem is actually more general, and holds for any l_p. (We present a proof for the ℓ_1 case due to Fakcharoenphol, Rao and Talwar (2003) since it has been useful in subsequent developments, as we will see.) Furthermore, examining the proof of the theorem one can derive an efficient algorithm to produce such a low-distortion embedding. (Aside, this is not always the case with mathematical proofs. Later in the course we shall encounter proofs of existence for combinatorial objects that do not entail efficient algorithms to construct these objects.)

We start with some notation we will use throughout this scribe. Denote the original metric by (X, d), where X is a set of $|X| = n$ elements and $d : X \times X \mapsto \Re^+$ a distance function. Denote by $Ball(x, R) \subseteq X$ the set of all elements $y \in X$ such that the distance to x is at most R.

We now describe a (fairly efficient) procedure to partition the elements of X. This procedure lies at the heart of the embedding.

Procedure Partition(A, B)

1. Pick uniformly at random a number $R \in [A, B]$.
2. Pick uniformly at random an order σ on the elements of X.
3. Partition the items of X into at most $n = |X|$ blocks as follows.
    (a) Proceed with the elements of X according to the order σ.
    (b) For each element $x \in X$, pick all non-assigned elements within distance R from it, and form a new block.

We call $P_{\sigma,R}$ the partition created above, and denote by $P_{\sigma,R}(x)$ the block in which x was placed.

remarks: (i) Some blocks may be empty. (ii) $P_{\sigma,R}(x)$ may not be the same as the block created using x in part 3(b) of the above procedure. The reason is that x may already be assigned to some other block.

The crucial property of this partitioning procedure is the following:

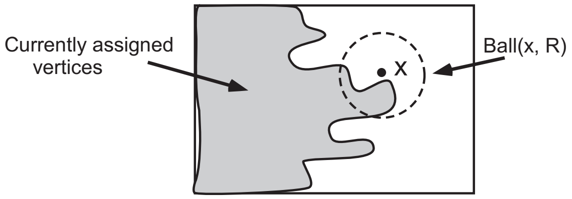

Figure 1: At each step, **Partition** takes the next element in order σ, say x, and creates a block consisting of all nonassigned elements in $Ball(x,R)$.

Theorem 2 (Padded decomposition property)
For every $\tau > 0, x \in X$ we have:

$$\Pr_{\sigma,R}[Ball(x,\tau) \nsubseteq P_{\sigma,R}(x)] \le \frac{4\tau}{B-A}\log\frac{|Ball(x,B+\tau)|}{|Ball(x,A-\tau)|}$$

Proof: Denote by E_z the event that z is the first element by the order σ such that $d(x,z) \le R+\tau$. Obviously,

$$\Pr[E_z] = \frac{1}{|Ball(x,R+\tau)|}.$$

Notice, $\bigvee_{z\in Ball(x,R+\tau)} E_z$ always happens.

We claim if $Ball(x,\tau) \nsubseteq P_{\sigma,R}(x)$, then the event $\bigvee_{z:d(x,z)\in[R-\tau,R+\tau]} E_z$ must have happened. (Indeed, consider the z which is the first element in $Ball(x,R+\tau)$ in the order σ. If $d(x,z)$ were $\le R-\tau$ then the block created using z would swallow all of $Ball(x,\tau)$.) We therefore bound the probability that $Ball(x,\tau) \nsubseteq P_{\sigma,R}(x)$ by:

$$\begin{aligned}\Pr_{R,\sigma}[Ball(x,\tau) \nsubseteq P_{\sigma,R}(x)] &\le \textstyle\sum_z \Pr_{R,\sigma}\left[E_z \bigwedge R \in [d(x,z)\pm\tau]\right] \\ &= \textstyle\sum_z \Pr\left[R \in [d(x,z)\pm\tau]\right]\cdot \Pr[E_z \mid R \in [d(x,z)\pm\tau]] \\ &\le \tfrac{2\tau}{B-A}\textstyle\sum_z \Pr\left[E_z \mid R \in [d(x,z)\pm\tau]\right] \qquad \text{R is chosen uniformly} \\ &\le \tfrac{2\tau}{B-A}\int_{r=|Ball(x,A-\tau)|}^{|Ball(x,B+\tau)|}\tfrac{1}{r}dr \\ &\le \tfrac{4\tau}{B-A}\log\tfrac{|Ball(x,B+\tau)|}{|Ball(x,A-\tau)|}\end{aligned}$$

□

Using the above procedure, we can now define the embedding into ℓ_1. We assume w.l.o.g (by scaling) that the given metric (X,d) has shortest distance 4, and largest distance Δ. The embedding we describe is probabilistic.

The final embedding of (X,d) into ℓ_1 is a composition of several ℓ_1 pseudo-metrics, one for each distance scale.

Procedure Embed(X, d)
For every $t \geq 1$ such that $2^t < \Delta$ **do:**

Invoke **Partition**$(2^t, 2^{t+1})$ to create a partition $P_{\sigma,R}$. Then define an embedding $\rho_t : X \to \Re^K$ where K is the number of blocks in $P_{\sigma,R}$ and

$$|\rho_t(x) - \rho_t(y)|_1 = \begin{cases} 2^t & \text{if } x, y \text{ are in different blocks of } P_{\sigma,R} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

(Notice that embedding ρ_t consists of placing each block of $P_{\sigma,R}$ on a coordinate axis in $\Re^K$ at a distance 2^{s-1} from the origin.)
The final embedding f is the trivial composition of ρ_t for all scales. (Namely, use fresh coordinates to accomodate each ρ_t and never reuse them for any other scale.)

Now we prove that the expected distortion of the embedding f is $O(\log n)$. We give a trivial (deterministic) lowerbound on $|f(x) - f(y)|_1$, and a probabilistic (expectation) upperound.

Lemma 3
For any $x, y \in X$ it holds that $\|f(x) - f(y)\|_1 \geq \frac{1}{4} d(x, y)$.

Proof: Let $P_{\sigma,R}$ be the partition created at scale t. Notice that if t is such that $2^{t+2} < d(x, y)$, then since $R \leq 2^{t+1}$, any ball of radius R cannot contain both x and y, and it must be that $P_{\sigma,R}(x) \neq P_{\sigma,R}(y)$ and therefore $\rho_t(x, y) = 2^t$. Hence,

$$\|f(x) - f(y)\|_1 = \sum_{t=0}^{\lfloor \log \Delta \rfloor} \rho_t(x, y) \geq \sum_{t | 2^{t+2} < d(x,y)} 2^t \geq \frac{1}{4} d(x, y)$$

□

Lemma 4
For any $x, y \in X$ it holds that $E_{\overline{\sigma},\overline{R}}[\|f(x) - f(y)\|_1] \leq O(d(x, y) \cdot \log n)$.

Proof: Using the definitions:

$$\begin{aligned} E[\|f(x) - f(y)\|_1] &= \textstyle\sum_{t=0}^{\lfloor \log \Delta \rfloor} E_{\sigma,R}[\rho_t(x, y)] \\ &\leq \textstyle\sum_{t=0}^{\lceil \log d(x,y) \rceil} E_{\sigma,R}[\rho_t(x, y)] + \sum_{t > \lceil \log d(x,y) \rceil}^{\lfloor \log \Delta \rfloor} E_{\sigma,R}[\rho_t(x, y)] \\ &\leq \textstyle\sum_{t=0}^{\lceil \log d(x,y) \rceil} 2^t + \sum_{t > \lceil \log d(x,y) \rceil}^{\lfloor \log \Delta \rfloor} E_{\sigma,R}[\rho_t(x, y)] \\ &\leq 4d(x, y) + \textstyle\sum_{t > \lceil \log d(x,y) \rceil}^{\lfloor \log \Delta \rfloor} E_{\sigma,R}[\rho_t(x, y)] \end{aligned}$$

For the larger values of t we have:

$$\begin{aligned}
\sum_{t>\lceil\log d(x,y)\rceil}^{\lfloor\log\Delta\rfloor} E_{\sigma,R}[\rho_t(x,y)] &= \textstyle\sum_{t>\lceil\log d(x,y)\rceil}^{\lfloor\log\Delta\rfloor} 2^t\cdot\Pr[P_{\sigma,R}(x)\neq P_{\sigma,R}(y)] \\
&\le \textstyle\sum_{t>\lceil\log d(x,y)\rceil}^{\lfloor\log\Delta\rfloor} 2^t\cdot\Pr[Ball(x,d(x,y))\not\subseteq P_{\sigma,R}(x)] \\
&\le \textstyle\sum_{t>\lceil\log d(x,y)\rceil}^{\lfloor\log\Delta\rfloor} 2^t\cdot\frac{2d(x,y)}{2^t}\log\frac{|Ball(x,2^{t+1}+d(x,y))|}{|Ball(x,2^t-d(x,y))|} \qquad \text{by theorem 2} \\
&\le 2d(x,y)\textstyle\sum_{t>\lceil\log d(x,y)\rceil}^{\lfloor\log\Delta\rfloor}\log\frac{|Ball(x,2^{t+2})|}{|Ball(x,2^{t-1})|} \\
&\le 2d(x,y)\cdot 3\log|Ball(x,\Delta)| = 6d(x,y)\log n
\end{aligned}$$

Combining both previous equations we get $E_{\sigma,R}[\|f(x)-f(y)\|_1] = O(d(x,y)\cdot\log n)$. $\Box$

From lemmas 3 and 4 we derive that the embedding described into ℓ_1 has *expected* distortion $O(\log n)$. In order to prove theorem 1, we need an embedding that has a *worst case* distortion guaranty. Here, again, we use the fact that the set of n-point ℓ_1 metrics is a convex cone. The randomized embedding that we have described thus far can be viewed as a distribution over deterministic embeddings into ℓ_1. The convex combination of all these ℓ_1 metrics is itself a an ℓ_1 metric with *worst case* distortion which is precisely equal to the expected distortion of the randomized embedding, i.e. $O(\log n)$.

Efficiency issues: Note that even though we have described a polynomial time procedure for embedding into ℓ_1 with low expected distortion, the last argument does not directly imply an efficient method to produce an embedding with *worst case* distortion $O(\log n)$. A polynomial time procedure with a *worst case* distortion guaranty can be derive using a standard method for derandomization - repeat the randomized constructed many times and take the average ℓ_1 metric of all those produced. This requires some care, since so far we have only bounded the expected distortion, and did not bound the standard deviation or other moments. The trivial bound is $poly(\Delta, n)$ time, which will be good enough for the application below.

2 Application to Sparsest Cut

For the rest of this lecture we use the following definition of the SPARSEST CUT problem: Given a graph $G=(V,E)$, find the subset of vertices $S\subseteq V$ that minimizes:

$$\min_{S\subseteq V}\frac{|E(S,\bar{S})|}{|S||\bar{S}|}$$

Notice that this is equivalent to finding the cut metric d_S minimizes $\frac{\sum_{(i,j)\in E} d_S(i,j)}{\sum_{i<j} d_S(i,j)}$, and as ℓ_1 is exactly the cone of cut metrics, the objective to minimize is:

$$\min_{d_S\text{ cut metric}}\frac{\sum_{(i,j)\in E} d_S(i,j)}{\sum_{i<j} d_S(i,j)} = \min_{d\in\ell_1}\frac{\sum_{(i,j)\in E} d(i,j)}{\sum_{i<j} d(i,j)}.$$

(In going from optimizing over cut metrics to optimizing over the cone of cut metrics we used the fact that $\min_i\left\{\frac{a_i}{b_i}\right\}\le\frac{a_1+a_1+\cdots}{b_1+b_2+\cdots}$, and thus the optimum in the latter case is wlog achieved at a single cut metric.)

As the sparsest cut problem is NP-hard, we consider a relaxation of the objective to all metrics, $\min_{d \text{ is a metric}} \frac{\sum_{(i,j)\in E} d(i,j)}{\sum_{i<j} d(i,j)}$. The optimum of this relaxation can be found in polynomial time by linear-programming. The linear programming formulation is:

$$\begin{aligned} \min \quad & \sum_{(i,j)\in E} d(i,j) \\ \sum_{i<j} \quad & d(i,j) = 1 \\ \forall i,j \in V \quad & d(i,j) \geq 0 \\ \forall i,j,k \in V \quad & d(i,j) + d(i,k) \geq d(j,k) \end{aligned}$$

This relaxation was first formulated by Sharokhi and Matula in the mid 80's, and first analyzed by Leighton and Rao in 1988, who showed —using duality theory—that the objective value of this relaxation is within $O(\log n)$ of the sparsest cut optimum.

In 1994, Linial, London and Rabinovich, and independently Aumann and Rabani, proposed a metric embedding viewpoint and showed how a $O(\log n)$ approximation can be derived using Bourgain's theorem. We now briefly describe how this is done.

Denote by OPT the value of the objective for the sparsest cut in the graph. Let d be the metric obtained from solving the linear program above. Since every ℓ_1 metric is feasible for the LP relaxation above, we have

$$\frac{\sum_{(i,j)\in E} d(i,j)}{\sum_{i<j} d(i,j)} \leq OPT.$$

Using Bourgain, one can embed metric d into ℓ_1 with distortion $O(\log n)$. Let d' be the resulting metric. Naturally, this increases the objective by at most a factor $O(\log n)$. Hence:

$$\frac{\sum_{(i,j)\in E} d'(i,j)}{\sum_{i<j} d'(i,j)} \leq O(\log n) \cdot \frac{\sum_{(i,j)\in E} d(i,j)}{\sum_{i<j} d(i,j)} \leq O(\log n) \cdot OPT$$

As we've in previous lectures, the ℓ_1 metric d' can be expressed as a positive combination of a polynomial number of cut metrics $d' = \sum_S \alpha_S d_S$. Picking the cut amongst these to minimize the objective ratio yields $O(\log n)$-approximate solution.

Efficiency issues. Now we have to confront the issue of how to make the above algorithm —particularly the embedding part—run in polynomial time. The embedding algorithm runs in time poly(n, Δ), where Δ is the *aspect ratio*, ie the ratio of the maximum internode distance to the smallest internode distance. It is easy to see that Δ never needs to be more than OPT_f/n^3, where OPT_f is the fractional optimum. The reason is that we could merge all node pairs whose distance in the optimum solution is at most OPT_f/n^3. This has negligible effect of the numerator and the denominator (at most an additive error of OPT_f/n) and still gives a metric. This metric can be embedded into ℓ_1 in polynomial time since its aspect ratio is poly(n).

(Aside: the above analysis of the running time can be greatly improved. The best running time to date is $O(n^2)$. It is obtained by multicommodity flow computations, which correspond to solving the dual of the linear program.)

3 Lower Bounds

The $O(\log n)$ bound on the approximation ratio of the above algorithm is in fact tight. Leighton and Rao proved in 1988 that the integrality gap for the above linear program (the maximum ratio between OPT and the LP objective) can be as large as $\Omega(\log n)$.

THEOREM 5
The integrality gap for the LP in the previous section is $\Omega(\log n)$.

In order to prove Theorem 5, we need the following definition:

DEFINITION 1 *A 3-regular graph $G = (V, E)$ is called a β-**expander** if for any set of nodes $S \subseteq V, |S| \leq \frac{n}{2}$, it holds that:*

$$|E(S, \bar{S})| \geq \beta |S|$$

Expander graphs are useful combinatorial objects we shall probably encounter more later in the course. A simple fact is that there is a $\beta > 0$ such that for any large enough n, a β-expander exists. (One can prove this by the probalistic method: imagine picking a graph randomly, and show that the probability that it is not a β-expander is $< 1/2$.)

PROOF:[Theorem 5] The counterexample is any β-expander family. Let $G = (V, E)$ be a β-expander. Then

$$\min_{S \subseteq V} \frac{|E(S, \bar{S})|}{|S||\bar{S}|} \geq \frac{\beta\,|S|}{|S|\,n/2} = \frac{2\beta}{n}.$$

We show that the optimum value of the linear program is $O(\frac{1}{n \log n})$.

This is shown by considering a particular metric (i.e., feasible solution): the shortest-path metric d on G. Since G is a 3-regular graph, we have:

$$\sum_{(i,j) \in E} d(i,j) = \frac{3n}{2} = O(n).$$

Furthermore, the average distance between two nodes in the graph is $\Omega(\log n)$. To see that, consider any vertex $v \in V$. This vertex has at most 3 other nodes —its neighbors—at distance 1, at most 3^2 vertices at distance 2, and so forth. Thus there are $\Omega(n)$ vertices at distance $\Omega(\log_3 n) = \Omega(\log n)$. Therefore:

$$\sum_{i<j} d(i,j) \geq \binom{n}{2} \cdot \Omega(\log n) = \Omega(n^2 \log n).$$

Hence, the value obtained by this graph metric is:

$$\frac{O(n)}{\Omega(n^2 \log n)} = O(\frac{1}{n \log n}).$$

□

Next time we will see a recent new algorithm that gives an $O(\sqrt{\log n})$ approximation.

References

[1] J. Bourgain. On lipschitz embeddings of finite metric spaces in hilbert space. *Israel Journal of Mathematics*, 52(1-2):46–52, 1985.

[2] T. Leighton and S. Rao. An approximate max-flow min-cut theorem for uniform multicommodity flow problems with application to approximation algorithms. *In IEEE Symposium on Foundations of Computer Science*, pages 422–431, 1988.

[3] N. Linial, E. London, and Y. Rabinovich. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15, 1995.

[4] Jittat Fakcharoenphol , Satish Rao and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics, *STOC '03: Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, 2003, pages 448–455,