Hierarchical Clustering for Euclidean Data

August 9, 2018

The goal of this project is to extend Dasgupta's hierarchical clustering paradigm to Euclidean datasets. Recall that in the graphical case given a graph $G(V, E)$ and a tree $\mathcal{T}$ whose nodes correspond to vertices in the graph Dasgupta's hierarchical clustering cost is defined as follows:

$$f^-(\mathcal{T}) = \sum_{(i,j)\in E} w_{ij} |\{x \in T(i,j)\}|,$$

where $T(i,j)$ is the subtree of $\mathcal{T}$ rooted in the least common ancestor of i and j in $\mathcal{T}$. Here $w(i,j)$ corresponds to a given similarity measure between vertices i and j. For the general case the best known approximation is $O(\sqrt{\log |V|})$ using the sparsest cut algorithm. Furthermore, under SSE-conjecture no algorithm can get a constant factor approximation in polynomial time [CC17]

We also consider a complementary objective which one aims to maximize:

$$f^+(\mathcal{T}) = \sum_{(i,j)\in E} w_{ij}(|V| - |\{x \in T(i,j)\}|),$$

Consider a set of vectors $v_1, \dots, v_n \in \mathbb{R}^d$. In this case the similarity measure w only depends on the underlying vectors, i.e. $w_{ij} = f(v_i, v_j)$ for some function $f \colon \mathbb{R} \times \mathbb{R} \to [0, 1]$.

Definition 0.1 (Monotone distance-based similarity measure)**.** *A similarity measure $w_{ij} = f(v_i, v_j)$ is* distance-based *if $f(v_i, v_j) = g(\|v_i - v_j\|_2)$ for some function $g \colon \mathbb{R} \to [0, 1]$. A similarity measure w_{ij} is* monotone distance-based *if furthermore $g \colon \mathbb{R} \to [0, 1]$ is a monotone non-increasing function.*

As a specific example of a monotone distance-based similarity measure it is natural to consider the Gaussian kernel similarity, i.e.:

$$w_{ij} = (\sqrt{2\pi}\sigma)^{-n} e^{-\frac{\|v_i - v_j\|_2^2}{2\sigma^2}},$$

where σ is a normalization factor. Below we will ignore the multiplicative factor as it doesn't affect multiplicative approximations. We will also set $\sigma = 1/\sqrt{2}$ to simplify the presentation.

Question: Is it possible to get a better approximation and/or faster algorithm for the Hierarchical clustering problem in this setting? Or can hard instances of the general case be embedded into a into vectors with weights from the Gaussian kernel?

1 Tight cases for Average-Linkage for f^+

1.1 High-dimensional case

For $i \in [n^{2/3}]$ and $j \in [n^{1/3}]$ let $v_{i,j} = \Delta(e_i + (1+\epsilon)e_{k+j})$ where $k = n^{2/3}$. Then it is easy to see that for any fixed $i \in [n^{2/3}]$ and $j_1 \neq j_2 \in [n^{1/3}]$ it holds that:

$$\|v_{i,j_1} - v_{i,j_2}\|_2^2 = 2(1+\epsilon)^2\Delta^2$$

For any fixed $j \in [n^{1/3}]$ and $i_1 \neq i_2 \in [n^{2/3}]$ it holds that:

$$\|v_{i_1,j} - v_{i_2,j}\|_2^2 = 2\Delta^2.$$

Otherwise if $i_1 \neq i_2 \in [n^{2/3}]$ and $j_1 \neq j_2 \in [n^{1/3}]$ then:

$$\|v_{i_1,j_1} - v_{i_2,j_2}\|_2^2 = 2\Delta^2 + 2(1+\epsilon)^2\Delta^2 \geq 4\Delta^2.$$

By setting $\Delta^2 > c \log n$ for a sufficiently large constant c the contribution of pairs of vectors with $i_1 \neq i_2$ and $j_1 \neq j_2$ can be made negligible. The rest of the pairs correspond to an embedded hard instance from Charikar, Chatziafratis and Niazadeh (SODA'19 submission) for which average-linkage only achieves a $\frac{1}{3}$-approximation compared to the optimum.

Using JL-transform we can reduce the dimension required for the above reduction to $d = O(\log n)$.

1.2 Low-dimensional case

For $d = 1$ the hardest instance seems to be the following. Take four equally spaced points on a line, i.e. $0, \Delta, 2\Delta, 3\Delta$. Then the average-linkage clustering algorithm might first connect the two middle points and then connect the two other points in arbitrary order. We denote the cost of this solution as AVG. An alternative solution would be to create two groups $(0, \Delta)$ and $(2\Delta, 3\Delta)$ and then merge them together. We denote the cost of this solution as OPT. By making Δ sufficiently large the contribution of pairs at distance more than Δ from each other can be ignored. We thus have:

$$AVG \approx 3e^{-\Delta^2}$$
$$OPT \approx 4e^{-\Delta^2},$$

which gives the ratio of 4/3.

Question: Is the 4/3 ratio achievable by average-linkage clustering for $d = 1$?

2 Hierarchical Clustering in 1D

We first consider the case $d = 1$. In this case the input can be represented as points on a line, which we assume to be sorted without loss of generality, i.e. $x_1 \leq \cdots \leq x_n$.

2.1 Random Cut

Consider the following algorithm: given a range of indices $[l, r]$ pick a uniformly random index i between l and $r - 1$ and split the range into two: $[l, i]$ and $[i + 1, r]$, then continue recursively until the range contains a single point. We apply this algorithm to the initial range $[1, n]$ and call it Random Cut.

Theorem 2.1. *For $d = 1$ under any monotone distance-based similarity measure $w_{ij} = g(x_i, x_j)$ the algorithm* Random Cut *gives a $\frac{1}{2}$-approximation for the objective f^+ in expectation.*

Proof. The expected value of f^+ for the algorithm Random Cut can be written as:

$$ALG = \mathbb{E}\left[\sum_{i=1}^{n} \sum_{j=i+1}^{n} w_{ij}(n - |\{x \in T(i,j)\}|)\right].$$

The rest of the analysis relies on the following two lemmas. In the first lemma we give an expression for the expected number of leaves in the subtree rooted in the least common ancestor of i and j:

Lemma 2.2. *The random variable $T(i,j)$ corresponding to the number of leaves in the subtree rooted in the least common ancestor of i and j in the tree produced by the algorithm* Random Cut *has expectation:*

$$\mathbb{E}[T(i,j)] = (j-i+1) + \sum_{k=j+1}^{n} \frac{j-i}{k-i} + \sum_{k=1}^{i-1} \frac{j-i}{j-k}$$

Proof. We can express $\mathbb{E}[T(i,j)]$ as follows:

$$\mathbb{E}[T_{i,j}] = \sum_{k=1}^{n} \ell_k(i,j),$$

where ℓ_k is the probability that point k belongs to the subtree rooted in the least common ancestor of i and j. The proof of the lemma then follows from the fact that ℓ_k can be expressed as:

$$\ell_k = \begin{cases} \frac{j-i}{j-k} \text{ if } k \le i-1 \\ 1 \text{ if } i \le k \le j \\ \frac{j-i}{k-i} \text{ if } k \ge j+1. \end{cases}$$

Indeed, by construction all points between i and j inclusive always belong to the subtree. Consider the case $k \le i-1$ as the other case is symmetric. Consider the first time the segment $[k,j]$ is partitioned by the algorithm Random Cut. Conditioned on $[k,j]$ being cut for the first time with probability $\frac{j-i}{j-k}$ this cut lands between i and j and hence k belongs to the subtree and otherwise it doesn't.

In the second lemma we give an upper bound on the optimum value of the objective, which we denote as OPT.

Lemma 2.3. *For any monotone distance-based similarity measure w the reward OPT of the optimum solution satisfies the following inequality:*

$$OPT \le \sum_{i=1}^{n} \sum_{j=i+1}^{n} w_{ij}(n-(j-i+1)).$$

Proof. Consider any fixed triple of indices (i,j,k) where $i<j<k$. In the optimal tree $\mathcal{T}^*$ consider the least common ancestor p or these three points. Note that if we go one level below p in $\mathcal{T}^*$ at least one of the points in the triple will get separated from the rest. This implies that only one pair in this triple can contribute to the objective and hence:

$$OPT \le \sum_{i<j<k} \max(w_{ij}, w_{ik}, w_{jk}) \le \sum_{i<j<k} \max(w_{ij}, w_{jk}),$$

where in the second inequality we use the fact that since w is a monotone distance-based similarity measure it holds that $w_{ij} \ge w_{ik}$ and $w_{jk} \ge w_{ik}$.

We can now divide all (i,j,k) triples into two sets: $\mathcal{S}_1 = \{(i<j<k) | w_{ij} \ge w_{jk}\} \mathcal{S}_2 = \{(i<j<k) | w_{ij} < w_{jk}\}$, and hence we have:

$$OPT \le \sum_{(i,j,k)\in\mathcal{S}_1} w_{ij} + \sum_{(i,j,k)\in\mathcal{S}_2} w_{jk} = \sum_{(i,j,k)\in\mathcal{S}_1} w_{ij} + \sum_{(k,i,j)\in\mathcal{S}_2} w_{ij} \le \sum_{i<j<k} w_{ij} + \sum_{k<i<j} w_{ij} = \sum_{i<j} w_{ij}(n-(j-i+1)).$$

∎

We are now ready to give the proof of Theorem 2.1. We introduce the following notation for $i<j$:

$$n_{ij} = n-(j-i+1)$$

$$e_{ij} = n_{ij} - \sum_{k=j+1}^{n} \frac{j-i}{k-i} - \sum_{k=1}^{i-1} \frac{j-i}{j-k}$$

By using Lemma 2.2 and Lemma 2.3 we have $\mathbb{E}[f^+(\mathcal{T})] = \sum_{i=1}^n \sum_{j=i+1}^n w_{ij} e_{ij}$ and $OPT \leq \sum_{i=1}^n \sum_{j=i+1}^n w_{ij} n_{ij}$. Hence it suffices to show that:

$$\sum_{i=1}^n \sum_{j=i+1}^n w_{ij} e_{ij} \geq \frac{1}{2} \sum_{i=1}^n \sum_{j=i+1}^n w_{ij} n_{ij}.$$

References

[CC17] Moses Charikar and Vaggos Chatziafratis. Approximate hierarchical clustering via sparsest cut and spreading metrics. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 841–854, 2017.

[Das16] Sanjoy Dasgupta. A cost function for similarity-based hierarchical clustering. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 118–127, 2016.