Pivot Algorithms for Clustering (extended abstract)

Jaime Cohen* Martin Farach†
Rutgers University

October 20, 1997

1 The Numerical Taxonomy Problem

The numerical taxonomy problem is usually cast in the following terms. Let S be a set of elements under consideration.

The Numerical Taxonomy Problem

Input: $D : S^2 \rightarrow \Re_{\geq 0}$, a distance matrix.

Output: A *tree metric* T which spans S such that the distance from T to D is minimized.

This definition does not specify a distance function between T and D. Since T induces a matrix of distances for S, we can take any standard distance function between matrices, such as L_1, L_2, or L_∞. That is, for some $k \in \{1, 2, \ldots, \infty\}$, we want to find a tree metric T minimizing $L_k(T - D)$. We can similarly seek to minimize the *relative error*, which is defined in terms of percent change, rather than absolute change. Another type of optimization criterion used in the biology literature is the so-called *Minimum Evolution Criterion (ME)* [SN87]. Here we seek the tree T, such that $T[i,j] \geq D[i,j]$, for all i and j, and such that the sum of the weight of the edges in T is minimized. This is the Steiner Tree problem for tree metrics.

The numerical taxonomy problems associated with most of the optimization criteria described above are NP-hard [War93, Day87, FKW93, ABF+96].

2 The Pivot Algorithm

The Single Pivot algorithm proposed in [ABF+96] is based on the transformation of the general tree metric problem to that of ultrametrics with a loss of a factor of 3 on the approximation ratio. An ultrametric is a tree metric in which the tree can be rooted so that the root is equidistant from all point of the metric. In [FKW93], it was shown that under the L_∞ norm an optimal ultrametric is polynomially computable, in fact in linear time.

Briefly we describe the Single Pivot heuristic as follows. A distance matrix C is called a *centroid* if $\exists l_1, \ldots, l_n$ such that $\forall i \neq j$, $D[i,j] = l_i + l_j$. Any tree metric D can be transformed

*Department of Computer Science, Rutgers University, Piscataway, NJ 08855, USA. (*jaimecoh@paul.rutgers.edu, http://paul.rutgers.edu/~jaimecoh*) Supported by a CAPES-Brazil Fellowship.

†Department of Computer Science, Rutgers University, Piscataway, NJ 08855, USA. (*farach@cs.rutgers.edu, http://www.cs.rutgers.edu/~farach*) Supported by NSF Career Development Award CCR-9501942, an Alfred P. Sloan Research Fellowship and NATO Grant 960215.

to an ultrametric by the addition of a centroid defined as follows. Choose an arbitrary $a \in S$, the *pivot* and let $m_a = \max_i\{T[a,i]\}$. Let C_T^a be the centroid metric with $l_i = m_a - T[a,i]$, i.e., $C_T^a[i,j] = l_i + l_j = 2m_a - T[a,i] - T[a,j]$. Notice that C_T^a is a matrix defined only in terms of T and a, and so it is well defined whether T is a tree metric or not. The following lemma relates the structure of T and $T + C_T^a$.

Lemma 2.1 ([BG91, Th.3.2]) *For all a, D is a tree metric if and only if $D + C^a$ is ultrametric.*

Thus, if some arbitrary matrix D is not a tree metric, $D + C_D^a$ is not an ultrametric. The insight in [ABF+96] is that $D + C_D^a$ can be mapped to its closest ultrametric via the algorithm in [FKW93]. Let $U(D + C_D^a)$ be the ultrametric so derived. Then we would like to set $T = U(D + C_D^a) - C_D^a$. Unfortunately, T need not be additive. In particular, we are subtracting C_D^a from $U(D + C_D^a)$, and when we do so, some of the stems leading to leaves might become negative. In [ABF+96], it was shown how to fix this problem, and we refer readers to their solution. Their modification, which changes the function $U()$ so as to guarantee that $T = U(D + C_D^a) - C_D^a$ satisfies the triangle inequality, yields a three approximation for the L_∞ criterion, no matter which a is chosen. Each choice of pivot costs $O(n^2)$ to evaluate, so the SP method costs up to $O(n^3)$ time, depending on how many pivots we wish to evaluate.

The Double Pivot Heuristic (DP) is similar to the algorithm described above except that it uses a centroid defined for a pair of points instead of just one point. In this way the possible candidates for pivots is increased to n^2. See [CF97] for datails.

The above strategy for solving the problem involves running the heuristic with different choices of pivots and selecting the best tree with respect to one of the distance functions that is more suitable for the application at hand. One important aspect to be considered on this selection is that the trees produced by the heuristic do not have edge lengths that are optimal for any criteria. In other words, it is possible to improve the distance from the given tree to the input matrix by changing the edge lengths without changing the topology of the tree. The best approach to find a good solution for the problem is to find the optimal edge lengths before comparing the candidate trees. Although this optimization can be done in polynomial time by solving a mathematical programming problem, it has a high cost in terms of running time. This problem can be seen as an obstacle for applying the pivot method to large dataset.

3 Conclusions

Experimental results showed that the DP heuristic outperforms the well known Neighbor Joining heuristic in randomly generated data and biological data [CF97]. However, many practical questions are still to be answered.

1. How to efficiently optimize the edge lengths for a fixed tree topology that minimizes its distance to the input matrix ?

2. Is there an heuristic to restrict the choices of pivots so to decrease the running time of the method without compromising the quality of the solutions ?

3. Is there other alternative ways to define the pivots so to obtain a better heuristic ?

References

[ABF$^+$96] R. Agarwala, V. Bafna, M. Farach, B. Narayanan, M. Paterson, and M. Thorup. On the approximability of numerical taxonomy: Fitting distances by tree metrics. *Proc. of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1996.

[BG91] J-P. Barthélemy and A. Guénoche. *Trees and Proximity Representations*. Wiley, New York, 1991.

[CF97] J. Cohen and M. Farach. Numerical taxonomy on data: Experimental results. *Proc. of the 8th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1997.

[Day87] W.H.E. Day. Computational complexity of inferring phylogenies from dissimilarity matrices. *Bulletin of Mathematical Biology*, 49(4):461–467, 1987.

[FKW93] M. Farach, S. Kannan, and T. Warnow. A robust model for finding optimal evolutionary trees. *Algorithmica*, 13:155–179, 1993.

[SN87] N. Saitou and M. Nei. The neighbor-joining method: a new method for reconstructing phylogentic trees. *Mol. Biol. Evol.*, 4:406–424, 1987.

[War93] H. T. Wareham. On the computational complexity of inferring evolutionary trees. Master's thesis, Department of Computer Science, Memorial University of Newfoundland, May 1993. Technical Report no. 9301.