CLUSTERING AND VISUALIZATION OF HIGH-DIMENSIONAL BIOLOGICAL DATASETS USING A FAST HMA APPROXIMATION

GUNJAN K. GUPTA ALEXANDER Y. LIU JOYDEEP GHOSH
Department of Electrical and Computer Engineering
The University of Texas at Austin,
Austin, TX 78712-1084, USA

ABSTRACT

In this paper, we reintroduce Hierarchical Mode Analysis(HMA), which was first proposed in 1968, as a powerful clustering algorithm for bioinformatics. The ability of HMA to find a compact hierarchy of a small number of dense clusters is very important in many bioinformatics problems (for example, when clustering genes in a set of gene-expression microarrays, where only a small number of genes related to the experimental context cluster well, while the rest need to be pruned). We also present two major improvements on HMA: a faster approximation algorithm, and a novel 2-D visualization scheme for high-dimensional datasets. These two improvements make HMA a powerful and promising new tool for many large, high-dimensional clustering problems in bioinformatics. We present empirical results on the Gasch dataset showing the effectiveness of our framework.

INTRODUCTION

In many real-world clustering problems, only a subset of the data actually needs to be clustered. This could be due to the fact that only a subset of our data actually clusters well while the rest can be treated as a "don't care" set. In particular, many types of large, high-dimensional bioinformatics datasets used for clustering genes exhibit the above property. From this data, biologists are interested in recovering clusters formed from small subsets of highly correlated genes.

An alternative to exhaustive clustering techniques are a class of non-parametric clustering algorithms that cluster the densest subset of data points in the entire dataset (e.g., [2, 1]). The first such approach was perhaps [6] that proposed an algorithm called *Hierarchical Mode Analysis*(HMA). In particular, the ability of HMA to *automatically* identify a compact hierarchy of clusters of varying density is highly desirable for many biological datasets. For example, gene-expression microarray datasets such as Gasch [3] are often created in a very well-defined and narrow context such as stress, and only the stress-related genes cluster well while the rest need to be pruned. Furthermore, for clustering genes on such data, there is usually no labeled data available, making model selection for clustering difficult. For such a setting, traditional clustering methods (such as K-Means and Agglomerative clustering) are difficult to apply because they cluster all the data, and/or require the number of clusters to be known.

In this short [1] paper, we build upon and improve the HMA algorithm in a way that makes it suitable for large, high dimensional, biological data. These improvements include: (1) creating a faster version of HMA appropriate for larger datasets; (2) the ability to use a variety of distance metrics including Pearson Distance [4], a biologically relevant distance measure; and (3) a novel visualization of the resulting cluster hierarchy. Our empirical results on Gasch dataset show that on high-dimensional biological data, one can obtain a very compact hierarchy of pure clusters with interesting sibling relationships.

SPEEDING UP HMA

Let us first describe the HMA algorithm as defined in [6]. Let $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^{n} \subseteq \mathbb{R}^d$ be a set of data points that need to be clustered. We assume that a relevant symmetric distance measure $d_S(\mathbf{x}_i, \mathbf{x}_j)$ is defined for all pairs of points $\mathbf{x}_i$ and $\mathbf{x}_j$ in $\mathcal{X}$. Let $\mathbf{M}_S$ represent the corresponding $n \times n$ symmetric distance matrix such that $\mathbf{M}_S(i, j) = d_S(\mathbf{x}_i, \mathbf{x}_j)$.

In addition, HMA uses the following notion of density. Given some $r_\epsilon \in \mathbb{R} : min(\mathbf{M}_S) \leq r_\epsilon \leq max(\mathbf{M}_S)$ as an input, the density $\rho_{r_\epsilon}(\mathbf{x})$ at any given point $\mathbf{x}$ is proportional to the number of points in $\mathcal{X}$ that are within r_ϵ of $\mathbf{x}$ [2]: $\rho_{r_\epsilon}(\mathbf{x}) \propto |\{\mathbf{y} \in \mathcal{X} : d_S(\mathbf{y}, \mathbf{x}) \leq r_\epsilon\}$

The HMA algorithm is as follows [3]:

1. Select the density threshold as integer $n_\epsilon < n$, compute the inter-point distance matrix $\mathbf{M}_S$ and the distances $\mathbf{d}^{n_\epsilon}$ from each point to its n_ϵ^{th} nearest point.

2. Order the distances $\mathbf{d}^{n_\epsilon}$ so that the smallest is first using the array $\mathbf{a}^{n_\epsilon}$ as an index. Thus $\mathbf{a}^{n_\epsilon}$ defines the order in which the data points become dense: point $\mathbf{a}^{n_\epsilon}(1)$ has the smallest n_ϵ^{th} distance $\mathbf{d}^{n_\epsilon}(1)$ and is first to become dense when $r_\epsilon = \mathbf{d}^{n_\epsilon}(1)$, point $\mathbf{a}^{n_\epsilon}(2)$ is second at $\mathbf{d}^{n_\epsilon}(2)$, and so on.

3. Select distance thresholds r_ϵ from successive $\mathbf{d}^{n_\epsilon}$ values, initializing a new dense point at each cycle. As the second and each subsequent dense point is introduced, the method tests the new point to determine one of three possible fusion phases: either (i) the new point does not lie within r_ϵ of another dense point, in which case it initializes a new cluster mode, (ii) the point lies within r_ϵ of dense points from one cluster only, and therefore the point is directly fused to that cluster, or (iii) the point falls in the saddle region, lying within r_ϵ of dense points from separate clusters, and the clusters concerned are fused.

[1] Extended version at: http://www.lans.ece.utexas.edu/∼gunjan/annie06/readme.html

[2] The set of points within r_ϵ distance of $\mathbf{x}$ includes $\mathbf{x}$.

[3] Since [6] is not easily available, the four steps below are presented exactly as in [6] except with the substitution of notation used in this paper.

4. Finally, a note must be kept of the nearest-neighbor distance r_{min} between dense points of different clusters. When r_ϵ exceeds r_{min}, the direct fusion of the two clusters separated by r_{min} is indicated.

For the labeled points (i.e., the dense points) from the i^{th} iteration of HMA, it can be shown that two dense points $\mathbf{x}, \mathbf{y} \in \mathcal{G}$ (where $\mathcal{G}$ is the set of dense points), belong to the same dense cluster represented as $\mathcal{C}$ if $d(\mathbf{x}, \mathbf{y}) < r_\epsilon$. That is,

$$\forall \mathbf{x}, \mathbf{y} \in \mathcal{G} : d(\mathbf{x}, \mathbf{y}) < r_\epsilon \Rightarrow \mathbf{x}, \mathbf{y} \in \mathcal{C} \tag{1}$$

As a consequence of equation 1, for any two points $\mathbf{x}_1$ and $\mathbf{x}_m \in \mathcal{G}$, if there exists a chain of points $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{m-1}, \mathbf{x}_m \in \mathcal{G}$ such that $\{d_S(\mathbf{x}_i, \mathbf{x}_{i-1}) < r_\epsilon\}_{i=2}^m$, then $\mathbf{x}_1$ and $\mathbf{x}_m$ also belong to the same cluster in a given iteration of HMA.

This leads to an algorithm that can compute the cluster labels in the i^{th} iteration of HMA directly without the iterative process required in HMA. We call this algorithm *Density Shaving* (DS). DS essentially takes two parameters as inputs: (1) f_{shave}, the fraction of least dense points to *shave* or exclude from consideration, and (2) n_ϵ, the number of points that must be within a distance r_ϵ of a given point x_i in order for x_i to be considered dense. The DS algorithm computes the corresponding r_ϵ using the same approach as HMA using $r_\epsilon = \mathbf{d}^{n_\epsilon}(i)$, where $i = (\lceil n f_{shave} \rceil)$. DS then applies a graph traversal process to discover the clusters composed of the dense points, where, as stated in equation 1, two dense points are in the same cluster if the distance between them is less than r_ϵ. The output of the algorithm consists of k clusters labeled 1 to k formed by the set $\mathcal{G}$ of n_c densest points and a "don't care" set $\mathcal{O}$ containing the remaining points that are labeled 0.

Unfortunately, constructing the full HMA hierarchy as described above has a time complexity of at least $O(n^3)$. While DS can be applied as a clustering algorithm, DS can also be used to construct a faster approximation of the HMA hierarchy because of the following observation:

Proposition 1: The cluster labels in each of the n iterations of the HMA hierarchy can be computed independently of one another.

This proposition follows as a direct consequence of the DS algorithm that can compute the i^{th} iteration of HMA directly without using the iterative procedure originally proposed by [6]. In addition, since HMA iteration cluster labels are nested (which follows from HMA directly), and because of Proposition 1, any subset of DS executions corresponding to some subset of the n HMA iteration clusterings also forms a hierarchical clustering. Thus, instead of computing all the n levels of HMA, one can skip levels of the full HMA hierarchy in order to approximate it. The time complexity of the approximation is $O(ln^2)$, where l is the number of levels that one calculates using DS.

The hierarchy produced by HMA involves top-down "growing" clusters; the algorithm starts with the densest point and then repeatedly merges an additional point in each iteration, either (1) starting a new cluster, (2) merging points in an existing cluster, or (3) merging two existing clusters. In contrast, though each of the independent iterations of DS produce an HMA level, a point having the same cluster label in two different levels of HMA may not have the same label when labeled using the corresponding two runs of DS. In order to produce labels that also correspond to the HMA labels on a hierarchical basis, a re-labeling of the cluster labels needs to be performed as follows, proceeding from levels with higher n_c to lower n_c: between two levels, for each cluster in the higher n_c level, if all points in the cluster belong to either a single cluster or the "don't care" set in the second level with lower n_c, then the same cluster ID is assigned to both the levels. If a cluster in the level with higher n_c splits into multiple child clusters in the level with lower n_c, then new cluster IDs are assigned to the child clusters.

VISUALIZING HMA

We now present a powerful, novel and intuitive visualization of the computed HMA hierarchy. We organize the cluster labels at each level of the hierarchy into a matrix. Each row represents a point in the dataset while each column is a level of the hierarchy. We first perform a dictionary sort on the rows of the $n \times l$ HMA label matrix, where the labels corresponding to a higher n_c level are given higher precedence. The matrix is then plotted in 2-D, with the rows oriented along the x-axis. Each label in the matrix is plotted in a unique color, with the "don't care" labels plotted in a background color (dark blue in figure 1(a)).

The visualization forms a compact, easy to understand 2-D visual representation of the high dimensional data (such as the 6,151 dimensional data in figure 1(a)). The x-axis corresponds to the number of dense points clustered in an HMA level, and is plotted on a log scale to enhance the visibility of the densest clusters. The y-axis corresponds to a projection of the points from the original high-d space onto a 1-d space, where points that are topologically or spatially close to each other appear close to each other on the y-axis. The visualization shows the spatial and topological relationship between all the HMA clusters very clearly, thus making it easy to explore and select clusters. The visualization also allows one to identify the clusters at any single level corresponding to a particular run of DS. Such an exploration allows one to interactively refine the tree by first choosing a few intermittent levels of the HMA hierarchy to create using DS, and then going back and exploring areas of interest in the hierarchy by creating more refined HMA levels skipped previously.

EXPERIMENTAL EVALUATION

Experimental Setup: We tested our framework on the Gasch dataset [3], a widely used benchmark for testing clustering algorithms on microarray data consisting of 6,151 genes of yeast *Saccharomyces cervisiae* responding to diverse environmental conditions over 173 microarray experiments. Since the labels for the experiments in the Gasch dataset were available, we performed evaluation on clustering of the microarray experiments rather than the genes.

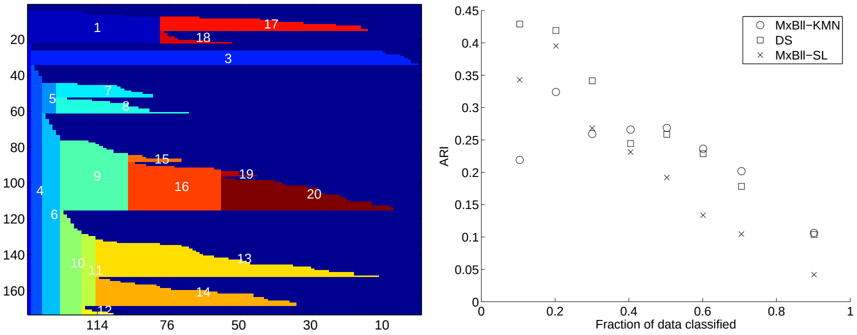
We performed two types of evaluations on our framework. For the HMA approximation, we evaluated individual clusters by examining the experiment labels directly. For each of the l HMA levels produced by DS, we used Adjusted Rand Index (ARI) [5] as our evaluation metric. Note that the points in the background or the "don't care" set were excluded from the evaluation.

Most labeled evaluation measures for clustering are sensitive to the number of clusters discovered and the percentage of data clustered. To get around this problem we ensured that the benchmark algorithms used the same n_c and k as our methods by applying the following procedure that we call *MaxBall*. First, k clusters are found, where k is given by the number of clusters found by DS for a particular n_c. Second, a cluster center (the mean of the member points) is computed for each cluster. Finally, the n_c points closest to their closest cluster center are clustered, while the remaining $n - n_c$ points are assigned to the "don't care" set. Using the MaxBall technique, we modified K-Means and Single Link agglomerative clustering.

We performed comparisons using different values of n_c. Since varying n_c for DS results in varying k, the corresponding k was used as an input to the benchmark algorithms. That is, since k is discovered by our framework and is given as an input to the benchmarks, they are not a viable alternative to our framework for finding dense regions automatically. Finally, the results for MaxBall K-Means were averaged over 10 trials while the other algorithms are deterministic. Pearson distance was used for all algorithms.

Results: Figure 1(b) compares DS with benchmarks on the Gasch data. In general, for smaller n_c that correspond to dense regions, DS tends to perform very well. Qualitatively, the clusters discovered by our HMA approximation are quite pure and meaningful. Figure 1(c-f) shows example experiment clusters along with the actual descriptions of the experiments. Note that the descriptions were not used in the clustering process. In addition, the hierarchy found by HMA is quite compact and easy to interpret (figure 1(a)). Many of the sibling clusters in the hierarchy are very interesting discoveries. For example, figure 1, (c) and (d) lists two sibling clusters that both contain a mix of hydrogen peroxide and Menadione experiments. In one particularly striking example (figure 1, (e) and (f)), both sibling clusters contain heat shock experiments. Interestingly, the heat shock experiments in the cluster in figure 1(e) involve a constant heat (37 degrees) and variable time, while the heat shock experiments in figure 1(f) involve variable heat and constant time.

Acknowledgments: This research was supported by NSF grants IIS-0325116 and IIS-0307792.



(a)

(b)

constant 0.32 mM H2O2 (10 min) redo 5
constant 0.32 mM H2O2 (80 min) redo 5
constant 0.32 mM H2O2 (100 min) redo 5
constant 0.32 mM H2O2 (120 min) redo 5
constant 0.32 mM H2O2 (160 min) redo 5
1 mM Menadione (10 min)redo 6
1 mM Menadione (120 min)redo 6
1 mM Menadione (160 min) redo 6

(c) Cluster 17 in figure (a)

constant 0.32 mM H2O2 (50 min) redo 5
constant 0.32 mM H2O2 (60 min) redo 5
1 mM Menadione (30 min) redo 6
1mM Menadione (40 min) redo 6
1 mM Menadione (50 min)redo 6
1 mM Menadione (80 min) redo 6
1 mM Menadione (105 min) redo 6

(d) Cluster 18 in figure (a)

Heat Shock 05 minutes hs-1 1
Heat Shock 10 minutes hs-1 1
Heat Shock 15 minutes hs-1 1
Heat Shock 20 minutes hs-1 1
Heat Shock 30 minutes hs-1 1
Heat Shock 40 minutes hs-1 1
Heat Shock 60 minutes hs-1 1
Heat Shock 80 minutes hs-1 1

(e) Cluster 7 in figure (a)

Heat Shock 17 to 37, 20 minutes 1
Heat Shock 21 to 37, 20 minutes 1
Heat Shock 25 to 37, 20 minutes 1
Heat Shock 29 to 37, 20 minutes 1
Heat Shock 33 to 37, 20 minutes 1
DBY7286 37degree heat - 20 min 12
DBYyap1-37degree heat-20 min(redo)9
DBY7286 + 0.3 mM H2O2 (20 min) 9
DBYyap1- + 0.3 mM H2O2 (20 min) 9

(f) Cluster 8 in figure (a)

Figure 1: (a): Visualization of HDS hierarchy on Gasch data using the sorted HMA label matrix. The x-axis, shows the number of dense points clustered at each HMA level, and the y-axis are the sorted rows. (b): ARI comparisons of DS on Gasch data. (c-f): examples of sibling clusters found (pair 1: (c) and (d); pair 2: (e) and (f)

REFERENCES

[1] M. Ankerst, M. Breunig, H. P. Kriegel, and J. Sander. Optics: Ordering points to identify the clustering structure. In *ACM SIGMOD*, 1999.

[2] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *In Proc. KDD-96*, 1996.

[3] Gasch A. P. et al. Genomic expression programs in the response of yeast cells to environmental changes. *Mol. Bio. of the Cell*, 11(3):4241–4257, December 2000.

[4] G. Gupta and J. Ghosh. Robust one-class clustering using hybrid global and local search. In *In Proc. ICML*, pages 273–280, Bonn, Germany, August 2005.

[5] L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, pages 193–218, 1985.

[6] D. Wishart. Mode analysis: A generalization of nearest neighbour which reduces chaining effects. In *Proc. Numerical Taxonomy*, pages 282–308, Univ. of St. Andrews, Fife, Scotland, September 1968. Academic Press.