

Robust DNA Microarray Clustering Techniques for Oncological Diagnosis

Robert Beverly
MIT Computer Science and Artificial Intelligence Laboratory
rbeverly@mit.edu

ABSTRACT

Machine learning techniques are increasingly popular tools for understanding complex biological data. Prior research has demonstrated the power of simple statistical clustering algorithms for disease class discovery and prediction. In this work we examine the efficacy of spectral and divisive clustering on gene expression microarray data. In particular we consider simultaneous expression clustering for diagnostically challenging problems such as tumor subclass classification and prediction. We compare spectral and divisive clustering methods against existing cancer classification datasets. Divisive clustering is notably non-parametric, enumerating an estimate of true class count. Using these two clustering methods, we demonstrate a 50-60% prediction error reduction over earlier results.

Categories and Subject Descriptors

J.3 [Computer Applications]: Life and Medical Sciences

Keywords

Microarray, Spectral Clustering, Divisive Clustering, Cancer, Classification

1. INTRODUCTION

Proper identification of cancerous tumors can be clinically challenging. Distinguishing tumors of similar morphological appearance is especially important when each require different treatments. Evidence suggests that tumor-specific therapies are essential to effective treatment with minimal toxicity. The advent of gene microarrays [2] has provided researchers and clinicians the ability to analyze thousands of genes simultaneously. Expression levels from microarrays enable principled methods to differentiate between e.g. tumors and tumor subclasses.

Machine learning is an increasingly popular tool for understanding and building models of complex biological data such as that from gene microarrays. One common technique is supervised and unsupervised clustering to partition the experimental data. For example, clustering is used to find similarities between gene expression patterns, discern disease types and build evolutionary trees.

Often simple clustering algorithms are used to great effect. This paper re-examines two prior works of microarray research for oncological classification: Golub et al. [5] and Nutt et al. [12]. Rather than collecting new data, we ask the question: “*Can more robust clustering techniques yield*

higher diagnostic accuracy than obtained previously?” Our goal is not to decry the results from other researchers, but rather to emphasize the power of simultaneous gene expression monitoring.

As in previous research, we consider the problem of class discovery. We examine the Golub and Nutt datasets with spectral [11] and divisive [10] clustering. We wish to cluster the experimental microarray samples based on inherent features of the data with no a priori knowledge of the true class. We then evaluate whether the resultant groups represent true structure or are simply random. After clustering tumors on the basis of gene expression levels, we measure the correspondence between the clustering and morphological and clinical outcomes.

The primary contributions of our research are:

1. A 50-60% prediction error reduction using spectral clustering over SOM and k-NN techniques used previously.
2. Novel use of non-parametric divisive clustering for tumor class discovery.

2. DATA SETS

The two data sets used in this paper are taken from previously published public clinical results: Golub et al. [5] and Nutt et al. [12].

The Golub work considers the problem of classifying acute leukemias, in particular differentiating between acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). Classification is traditionally performed by analyzing subtle differences in nuclear morphology [6]. Unfortunately these qualitative diagnoses are prone to error. In contrast, Golub et al. use self-organizing maps (SOMs) to perform unsupervised class discovery. SOMs employ a k-means centroid strategy [8] to cluster data points based on their expression patterns. Without knowledge of the underlying AML-ALL distinction, their celebrated result demonstrates SOMs properly clustering tumors with few errors.

A second example of tumors which are difficult to precisely identify on the basis of histological features are high-grade malignant gliomas, i.e. brain tumors [7]. As with leukemias, proper tumor classification impacts both therapeutic course and patient prognosis. Nutt et al. utilize gene expression profiling with a k-Nearest Neighbors (k-NN) predictor. Their

Table 1: Data Sets

Data Set	Examples	Description
Golub	27 ALL 11 AML	Acute Leukemias
Nutt	14 GB 7 AO	Malignant Gliomas

results differentiate between glioblastomas (GB) and anaplastic oligodendrogliomas (AO) with approximately 86% accuracy.

Table 1 summarizes the two data sets. The remainder of this paper focuses on alternate means of clustering the data from these previous studies.

2.1 Data Abstraction

Consider experimental data in the form of a set of gene microarray samples. Each sample consists of expression levels corresponding to d genes. In order to computationally analyze the data, we abstract each sample into a data point in a d -dimensional space, \mathbb{R}^d . We compute the Euclidean distance between $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ as

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{(x_1 - y_1)^2 + \dots + (x_d - y_d)^2} \quad (1)$$

With this distance metric, we associate each sample with its k nearest neighbors. The k-NN computation can be used to construct a graph $G = (V, E, W)$ with vertices V , edges E and edge weights w . Vertices represent an experimental sample and two vertices are connected by an edge e if they are nearest neighbors. Weights are defined between all vertices $i, j \in V, i \neq j$, i.e. $W_{i,j}$. If no edge exists between i and j then $W_{i,j} = 0$.

3. SPECTRAL CLUSTERING

Both k-means clustering and self-organizing maps (SOM) are simple and computationally efficient. From points in \mathbb{R}^d for d genes, the algorithm begins by placing k centroid points randomly in the space. These k initial points indicate the number of clusters to form and correspond directly to the number of classes. In each iteration, each data sample is associated with the nearest centroid as measured by the Euclidean distance in Eq. 1. The position of each centroid is recomputed to be the graph theoretic center of all data points associated with that centroid in the current iteration. Once an iteration results in no data points being associated with a different centroid, the clusters are found. The algorithm is guaranteed to converge.

However, the use of centroids force a spherical interpretation of the data and may produce poor results. For instance, [11] contains several degenerate examples of k-means clustering. This section considers an alternate more robust technique, spectral clustering.

3.1 Clustering Methodology

Spectral clustering mimics a Markov random walk¹ [9]. The intuitive notion is that irrespective of starting point, the random walk will transition often between points within clusters and seldom jump between clusters.

Spectral clustering is known to be superior in many cases, for instance when the underlying data cannot be separated by convex regions. By using the spectral properties of the graph, one can mimic a random walk.

For brevity, we omit a full description of spectral clustering; details are available in e.g. [13]. At a high-level, spectral clustering employs the following four-steps:

1. Construct neighbor graph using k-NN to add edges
2. Assign edge weights exponentially proportional to distance
3. Define transition probability over edges
4. Cluster based on eigenvectors of probability matrix

Spectral clustering builds a neighbor graph whose edge weights are proportional to the Euclidean distance in \mathbb{R}^d . To construct the neighbor graph, assign weights based on Euclidean distance with exponential fall-off. If an edge exists between vertices i and j in the graph, then weight $W_{i,j}$ is:

$$W_{i,j} = e^{-\beta \|x_i - x_j\|} \quad (2)$$

In our experiments, $\beta = 1$. In order to model the Markov random walk over the graph, we first normalize edge weights to form transition probabilities. Let $P_{i,j}$ be the probability of transitioning from vertex i to j :

$$P_{i,j} = \frac{W_{i,j}}{\sum_j W_{i,j}} \quad (3)$$

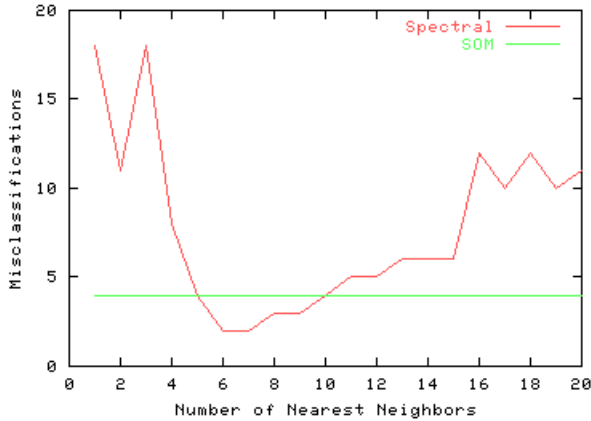
Where the random walk is modeled after t random steps as P^t . The distribution of points converges as t increases. If graph is connected and ergodic, the distribution becomes independent of the starting point. Spectral clustering recovers the random walk effect from the eigenvectors of the graph Laplacian.

3.2 Results

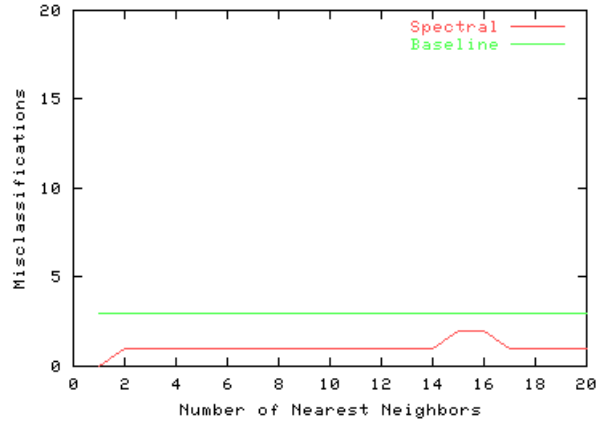
We apply spectral clustering to the Golub and Nutt data sets. We compute the number of misclassifications as a function of the number of nearest neighbors. Figure 1(a) depicts the clustering performance on the Golub data where the baseline performance of four misclassifications is given by the SOM result. Two points bear immediate notice. First, the performance is sensitive to the number of nearest neighbors. However, given a proper choice of nearest neighbors, the number of misclassifications is half that of SOM.

Despite this mixed result, Figure 1(b) shows the clustering performance on the Nutt data. For all choices of nearest neighbors spectral clustering provides improved classification results.

¹Other interpretations of spectral clustering exist including graph cut; we focus on random walks as most intuitive.



(a) Golub ALL/AML Clustering Performance



(b) Nutt Gliomas Clustering Performance

Figure 1: Spectral Clustering of Gene Expression Microarray Data into Oncological Class

4. DIVISIVE CLUSTERING

Next we examine a divisive clustering algorithm. From a k -nearest neighbors graph, we determine the “betweenness centrality,” i.e. the number of shortest paths traversing each edge in the graph. We then remove edges in decreasing betweenness value, calculating a modularity score at each step. The graph with the best modularity score determines the optimal clustering. This algorithm is attractive because it does not require a priori knowledge of the number of clusters.

4.1 Betweenness Centrality

We use a recently developed divisive rather than agglomerative algorithm from Newman [10] that has empirically been shown to produce better results. Newman’s algorithm relies on the notion of “betweenness centrality” of nodes.

For $G = (V, E)$, let σ_{st} be the number of shortest paths from s to t in G . By convention, $\sigma_{ss} = 1$. Define $\sigma_{st}(v)$ as the number of shortest paths from s to t on which $v \in V$ lies. Let the pair-dependency of nodes s, t on v be $\delta_{st}(v) = \frac{\sigma_{st}(v)}{\sigma_{st}}$. Thus the pair-dependency is simply the ratio of the number of shortest paths between s and t that v lies on.

The betweenness centrality [3] of a vertex $v \in V$ is the ratio of the number of shortest paths involving v , to the total number of shortest paths, taken over all vertex pairs $(i, j) \in V$. In other words betweenness centrality is a metric of the *relative importance* of a particular node in the network.

Formally, the betweenness centrality for a node v is:

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}} = \sum_{s \neq v \neq t \in V} \delta_{st}(v) \quad (4)$$

In the naive approach, calculating betweenness centrality is dominated by computing the sum of pair-dependencies which has a running time complexity of $O(n^3)$. Fortunately, Brandes presents a fast algorithm that requires $O(nm)$ time and $O(n + m)$ space [1].

We modify Brandes’ algorithm for vertex betweenness centrality to compute edge betweenness, i.e. the number of shortest paths traversing each edge in the network. Thus, edge betweenness can be seen as the traffic flow along an edge when all nodes in the graph source traffic to all other nodes. Formally, we define the betweenness centrality for an edge e as:

$$C_B(e) = \sum_{s \neq t \in V} \frac{\sigma_{st}(e)}{\sigma_{st}} \quad (5)$$

Where $\sigma_{st}(e)$ is the number of shortest paths from s to t that contain edge e .

The output of our algorithm is an adjacency matrix B where $B_{i,j} = B_{j,i}$ corresponds to the betweenness centrality of the undirected edge between nodes i and j in G . Formal correctness guarantees are given in detail in Brandes’ full paper [1].

4.2 Newman Community Algorithm

Both divisive and agglomerative methods suffer from a common problem: when to stop adding or removing edges. Newman proposes the “modularity” function Q . Let ϕ_{ij} be the fraction of edges in the network that connect vertices in group i to group j . Groups, which we expand upon when presenting the algorithm, are the connected components of the graph. Then:

$$Q = \sum_i (\phi_{ii} - a_i^2) \quad (6)$$

where $a_i = \sum_j \phi_{ij}$. Intuitively, Q is the fraction of edges that lie within communities minus the expected value of the fraction of edges that lie within communities in a randomly constructed graph. As pointed out by Newman, a value of $Q = 0$ is indicative of community structure that is no more prevalent than would be expected if the graph were constructed randomly with the same degree as the original graph.

Newman’s algorithm [10] for finding community structure is the following four-steps:

1. Determine edge betweenness centrality for all edges. Sort edges by non-increasing centrality.
2. While edges remain, remove the edge with the highest centrality. Ties are broken randomly.
3. Compute the modularity, Q , for each resultant graph.
4. Once all edges are removed, output the graph with the highest modularity score. This graph represents the inherent community structure in the original input graph.

4.3 Results

We apply the divisive clustering algorithm to the Golub and Nutt data sets. While the modularity score eliminates the need to know the number of classes a priori, we must still create the initial connected graph. We again use k-NN from the gene expression data. To visualize the clusterings simultaneously with the true tumor subclass type, we use the Graphviz [4] package.

Figure 2(a) shows a representation of a $k = 6$ nearest neighbors graph while 2(b) depicts the graph structure after divisive clustering. Note that the red and green vertex colors indicate the true tumor subclass, either ALL or AML. The visual interpretation of the results clearly shows two distinct clusters. There is one misclassification in each of the green and red clusters. However, the misclassified green node is only weakly connected to the strongly connected red nodes. While $k = 6$ gives the best performance, we obtain similar results when varying k , suggesting that the divisive clustering algorithm works well. Without a priori knowledge of the number of actual classes, unsupervised divisive clustering in this example found the two true classes.

Unfortunately, divisive clustering performs poorly on the Nutt data set. Figure 3(a) gives the $k = 4$ nearest neighbor graph of the data where red and green vertices indicate the true tumor type. After divisive clustering, the graph contains five connected components. While the accuracy is high (the five node component contains no misclassifications and the nine node component contains only one misclassification), the recall is poor if there are indeed only two true classes. Again, we obtain similar results when varying k , implying that the divisive clustering has mixed performance on this data set. However, we entertain the possibility that additional unknown subclasses may exist.

5. CONCLUSION

In this work, we use existing oncological data sets and applied two robust clustering techniques. Spectral clustering outperformed traditional methods by as much as 60%, but appears sensitive to parameter selection. Future research includes finding a different approach to graph edge weight selection.

To understand the inherent classes present in the data, we use a new non-parametric divisive clustering method from Newman. This unsupervised clustering algorithm performed

very well on the first data set, but produced mixed results on the second.

Computational methods combined with gene expression data is giving new light to diagnostically challenging classification and prediction problems. Our results suggest that researchers may wish to consider the robust methods of spectral and divisive clustering for DNA microarray data analysis.

6. REFERENCES

- [1] BRANDES, U. A faster algorithm for betweenness centrality. *Mathematical Sociology* (2001).
- [2] DERISI, J., PENLAND, L., AND BROWN, P. O. Use of a cDNA microarray to analyse gene expression patterns in human cancer. *Nature Genetics* 14, 4 (1996), 457.
- [3] FREEMAN, L. C. A set of measures of centrality based on betweenness. *Sociometry* 40 (1977), 35–41.
- [4] GANSNER, E. R., AND NORTH, S. C. An open graph visualization system and its applications to software engineering. *Software — Practice and Experience* 30, 11 (2000), 1203–1233.
- [5] GOLUB, T., SLONIM, D., TAMAYO, P., HUARD, C., GAASENBEEK, M., MESIROV, J., COLLIER, H., LOH, M., DOWNING, J., CALIGIURI, M., BLOOMFIELD, C., AND LANDER, E. Molecular classification of cancer: Class discovery and class prediction by gene expression. *Science* 286, 5439 (Oct. 1999).
- [6] HAYHOE, F., AND QUAGLINO, D. *Hematological Cytochemistry*. Churchill Livingstone, 1994.
- [7] IRONSIDE, J. W., MOSS, T. H., LOUIS, D. N., LOWE, J. S., AND WELLER, R. O. *Diagnostic Pathology of Nervous System Tumours*. Churchill Livingstone, 2002.
- [8] LLOYD, S. Least-squares quantization in pcm. *IEEE Transactions on Information Theory* 28, 2 (1982).
- [9] MEILA, M., AND SHI, J. A random walks view of spectral segmentation, 2001.
- [10] NEWMAN, M. E. J., AND GIRVAN, M. Finding and evaluating community structure in networks. *Phys. Rev.* (2004).
- [11] NG, A. Y., JORDAN, M. I., AND WEISS, Y. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems* (2001).
- [12] NUTT, C., MANI, D., BETENSKY, R., TAMAYO, P., CAIRNCROSS, J., LADD, C., POHL, U., HARTMANN, C., McLAUGHLIN, M., BATCHELOR, T., BLACK, P., VON DEIMLING, A., POMEROY, S., GOLUB, T., AND LOUIS, D. Gene expression-based classification of malignant gliomas correlates better with survival than histological classification. *Cancer Research* 63 (2003).
- [13] VON LUXBURG, U. A tutorial on spectral clustering. Tech. Rep. TR-149, Max Planck Institute of Biological Cybernetics, Aug. 2006.

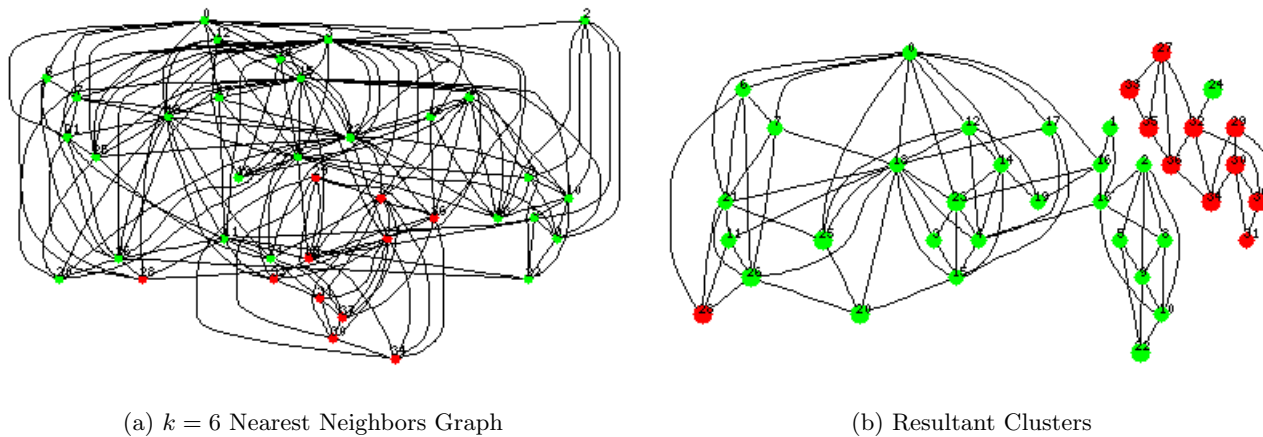


Figure 2: Divisive clustering of Golub ALL/AML data. Vertex colors indicate true tumor subclass type.

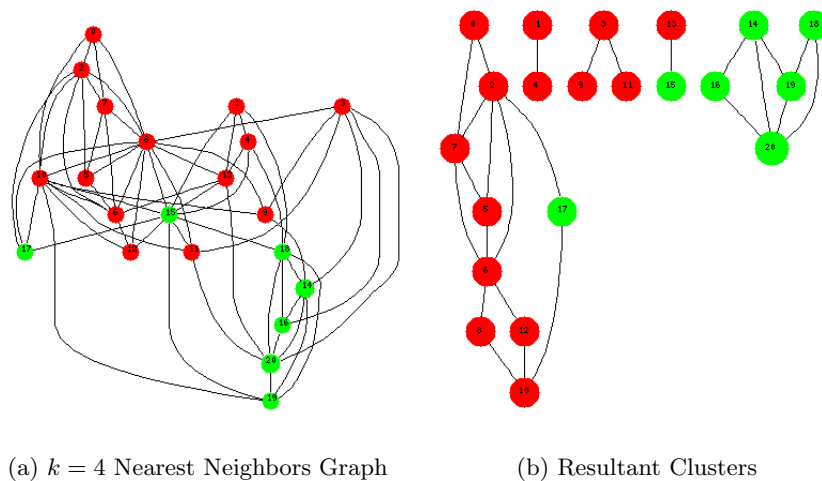


Figure 3: Divisive clustering of Nutt Gliomas data. Vertex colors indicate true tumor subclass type.