jet INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2010, VOL. 56, NO. 3, PP. 281-286
Manuscript received July 10, 2010: revised September, 2010. DOI: 10.2478/v10177-010-0037-9

Application of Density Based Clustering to Microarray Data Analysis

Lech Raczynski, Krzysztof Wozniak, Tymon Rubel, and Krzysztof Zaremba

***Abstract*—In just a few years, gene expression microarrays have rapidly become a standard experimental tool in the biological and medical research. Microarray experiments are being increasingly carried out to address the wide range of problems, including the cluster analysis. The estimation of the number of clusters in datasets is one of the main problems of clustering microarrays. As a supplement to the existing methods we suggest the use of a density based clustering technique DBSCAN that automatically defines the number of clusters. The DBSCAN and other existing methods were compared using the microarray data from two datasets used for diagnosis of leukemia and lung cancer.**

***Keywords*—Microarrays, cluster analysis, DBSCAN.**

I. Introduction

IN SPITE of a very quick development of medicine within the last decade, finding the course of the disease for a person diagnosed with cancer can often be puzzling. A definitive diagnosis of cancer involves biopsy, and examining the cells under a microscope. Although the analysis of morphologic characteristics of biopsy specimen is still the standard diagnostic method, it gives very limited information and clearly misses out on a lot of important tumor aspects such as capacity for invasion, and development of resistance mechanisms to certain treatment agents. This often leads to the merging of together different subtypes into one diagnostic class.

To appropriately classify tumor subtypes, the molecular diagnostic methods are needed, such as microarrays. A major advantage of a microarray is a huge amount of molecular information that can be extracted and integrated to find common patterns within a group of samples. Microarrays could be used in combination with other diagnostic methods to add more information about the tumor specimen by looking at thousands of genes concurrently. Microarray experiments are being increasingly carried out in biological and medical research to address the wide range of problems, including the cluster analysis. Recent studies show [4][1] that the analysis of microarray data can help in discovering cancer subclasses. In the medical application of microarray-based cancer diagnosis, the definition of a new tumor would be based on clustering results. Inaccurate cluster assignment could lead to wrong diagnoses and unsuitable treatment protocol.

Numerous clustering algorithms have been applied in the microarray data analysis. The most commonly used methods, such as k-means [12], partitioning around medoids (PAM) [8] or self-organizing maps (SOM) [9] require the specification of the number of clusters in advance. However, the correct estimation of the number of clusters is often a challenging task, especially in the case of complex, multivariate and noisy data from microarray experiments. Furthermore, a major drawback of methods with fixed number of clusters is the fact that they force every sample to be assigned to a cluster, regardless to the quality of the resulting dataset partitioning. In this paper we propose the usage of Density Based Clustering of Applications with Noise (DBSCAN) [3] for microarray data clustering. As the DBSCAN algorithm automatically defines the number of clusters and allows the assignment of outlier samples to a separate noise cluster, it seems to be well suited for gene microarray-based studies. Here we compare the DBSCAN performance with other frequently used clustering algorithms on the basis of the real gene expression data.

L. Raczynski, K. Wozniak, T. Rubel, and K. Zaremba are with the Institute of Radioelectronics, Warsaw University of Technology, Nowowiejska 15/19, 00-665 Warsaw, Poland (e-mail: {lraczyn; kwozniak; trubel; K.Zaremba}@ire.pw.edu.pl).

II. Microarrays

In the microarray experiments, the information about the genes activity is obtained. Genes consist of deoxyribonucleic acid (DNA). DNA contains the code, or blueprint, used to synthesize a protein. Genes vary in size, depending on the sizes of the proteins for which they code. Each DNA molecule is a long double helix that resembles a spiral staircase containing millions of steps.

Microarrays depend on the basic principle: complementary sequences of nucleotides hybridize to one another. For example, one strand of the DNA molecule with the sequence TCATGC will hybridize to another strand with the sequence AGTACG to form a double-stranded DNA. The information about the gen activity is obtained from the concentration of corresponding messenger RNA (mRNA) which is the molecule produced when a gene is expressed. DNA microarray profiling uses a small, flat chip that has thousands of single-stranded DNA embedded on its surface. The mRNA extracted from the cell is applied to the chip, where it sticks to the complementary pieces of DNA. Each region of the array, checking the presence of specific nucleotide sequence in the sample, is called a probe. The information about the level of the gene activity is obtained by the fluorescent marker added to the mRNA in the sample. In the array scanning process, for each probe the fluorescence intensity of the marker excited by the laser light is measured and stored as pixel intensity levels in the image. The resulting image is used to observe different patterns of gene expression between different tumors.

The main two technologies based on such an overall scheme are complementary DNA (cDNA) microarrays and the

high-density oligonucleotide microarrays. In the first case, the long DNA sequences, chosen from a clone library and amplified by the polymerase chain reaction (PCR), are spotted on a glass slide [13]. In this experiment, microarrays are hybridized with RNA from two sources labeled with different fluors. The two color channels are referred to by convention as red and green. The fluorescent red and green cDNA samples are then applied to a microarray. Computer programs calculate the red to green fluorescence ratio in each spot. The calculated ratio for each spot on the array reflects the relative expression of a given gene in the two samples. The latter type of microarrays are produced by Affymetrix and sold as GeneChip. In this microarrays, the probes are made up of a number of short, fixed length fragments called oligonucleotides. The nucleotide sequences are synthesized using photolithography on silicon substrate [11]. In the contrary to the first technique, in high-density oligonucleotide microarrays the absolute expression level in one cell population is measured.

After the image preprocessing, the data from a microarray experiments may be presented in the form of a matrix. Each element of the matrix addressed by an indexes i, j is a value proportional to relative or absolute (depending on type of used microarray) level of the i^{th} gene in the j^{th} sample. A single column of the matrix corresponds to the expression profile of all genes in the sample. Each row represents the expression pattern of one gene over all samples. In all of the microarray experiments, the resulting array has much more rows than columns. In consequence, a single microarray can measure the expression level of thousands of genes at the same time, while the number of the arrays used in the experiment is typically lower than one hundred.

III. Cluster Analysis

Clustering is a basic multivariative technique that groups a number of samples into clusters on the basis of a specific similarity/dissimilarity measure. Cluster analysis approaches entail making several choices, such as what algorithm to use in determining the cluster solution, which metric to use to quantify the similarity/dissimilarity among pairs of samples, and how many clusters to include in the solution. From a large number of clustering algorithms, the most common are k-means [12], partitioning around medoids (PAM) [8], self-organizing maps (SOM) [9] and hierarchical clustering [7]. All clustering algorithms use a similarity/dissimilarity measurement between samples to compare the patterns. There are dozens of the available metrics, for example: Manhattan distance, Euclidian distance, Pearsons correlation coefficient, or averaged dot product. Regardless of the selected algorithm and the metrics, the main aspect of the clustering problem is to accurately estimate the number of clusters in a dataset. Typical clustering algorithms [12] [8] [9] require the number of clusters as an input parameter. For those methods, the most common approach to estimate the number of clusters is based on finding the K clusters in a dataset that provides the strongest significant evidence against the hypothesis that there are no clusters ($K = 1$). Numerous methods have been proposed for testing the hypothesis that $K = 1$ and estimating the number of clusters in a dataset [8] [10] [14] [2] [5]. For example, Kaufman and Rousseeuw [8] suggest selecting the number of clusters basing on the silhouette plot. The silhouette width of i^{th} sample in a dataset is defined as:

$$sil_i = (b_i - a_i)/max(a_i, b_i), \tag{1}$$

where a_i denotes the average dissimilarity between i and all other samples in the cluster to which i belongs, and b_i is the minimum average dissimilarity of i to other samples in other clusters. For a given number of clusters K, the overall average silhouette width for the clustering is simply the average of sil_i over all samples i,

$$asil = \sum_i sil_i/n. \tag{2}$$

Kaufman and Rousseeuw suggest estimating the number of clusters K by that which gives the largest average silhouette width, asil.

Other type of clustering methods, are those which do not specify the number of clusters prior to running the algorithm, like for example Quality Threshold (QT) clustering [6] and DBSCAN [3].

The focus of the QT algorithm is to find the large clusters that have a quality guarantee. The method requires two parameters; $MinPts$ that is the minimum number of samples in any cluster; d that is a cluster diameter, which means that any two samples in a cluster have a jackknife correlation value that is at least $1-d$. The cluster diameter can range from 0 to 2, because the jackknife correlation lies in the interval [-1,1]. For each sample in the dataset, the group is formed with the samples that have the greatest jackknife correlation with it. The number of candidate clusters is equal to the number of samples, and many candidate clusters overlap. At this point, the largest cluster is selected and retained. The samples it contains are not considered and the entire procedure is repeated on the smaller set. A termination criterion is to stop when the largest remaining cluster has fewer than $minPts$ samples.

DBSCAN algorithm is a clustering method that uses density of the samples as a parameter guiding the clustering procedure. The method requires two parameters; $MinPts$ that is the minimum number of samples in any cluster; Eps that is the maximum distance of the sample to at least one other sample within the same cluster.

In the first step, all samples are divided into three groups; core points that is samples with at least $MinPts$ samples within Eps; border points that is samples that have at least one core point within Eps; noise - all the remaining samples. In the second step, the clusters are constructed by starting from any unclustered core point, that is a seed for a new cluster, and iteratively checking all the neighbors within Eps. If the neighbor is a core point as well, then its neighbors are checked, too. The border points are just added to the cluster without verifying their neighbors.

In DBSCAN method a two parameters have to be defined. While $MinPts$ is naturally defined as the minimum size of the cluster wanted to be discovered, choosing appropriate Eps value is not obvious. The DBSCAN authors proposed the approach for determining the parameter Eps. For a given

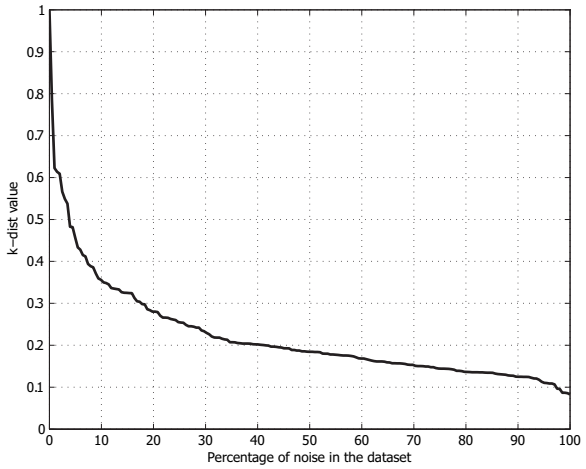


Fig. 1. The example of sorted k-dist graph.

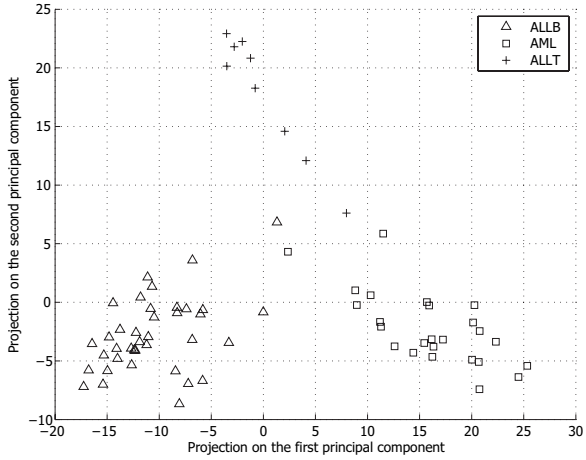


Fig. 2. Leukemia dataset.

$MinPts$ they defined a function k-dist mapping each point to the distance from its k_{th} nearest neighbor. When sorting the points of the database in descending order of their k-dist values and choosing arbitrary Eps value, the resulting noise cluster size is no greater than the number of points with greater k-dist value than Eps. In other words, the sorted k-dist function enables to estimate the Eps value on the basis of the assumed percentage of noise in the dataset. The example of sorted k-dist graph is presented in Fig. 1. For instance, for prespecified 20% of noise in the dataset, Eps value used for clustering should be approximate to 0.28 basing on the curve in Fig. 1.

IV. Experimental Section

In this paper, our main concern is to estimate the number of clusters in a dataset. We decided to analyze the DBSCAN properties, in comparison to the three other clustering methods, which are widely used for clustering gene expression data. The algorithms belong to two categories in respect of the ability to automatically define the number of clusters. The first category includes two algorithms which use the number of cluster as an input parameter: k-means and SOM. Latter category, besides the DBSCAN, consists of the second described method which does not require the number of clusters the QT algorithm. To estimate the number of clusters produced by k-means and SOM we used the average silhouette width parameter described above. We have tested those four algorithms on the real gene expression data.

To assess the quality of algorithms, we need some objective external criteria. Since the DBSCAN and QT define the noise cluster, while k-means and SOM does not, comparing of the results is difficult. In order to compare clustering results against the true class information, we employ the number of bad classified samples (b) and the number of noise samples (n). Second value (n) is defined only for DBSCAN and QT, because k-means and SOM does not define the noise cluster.

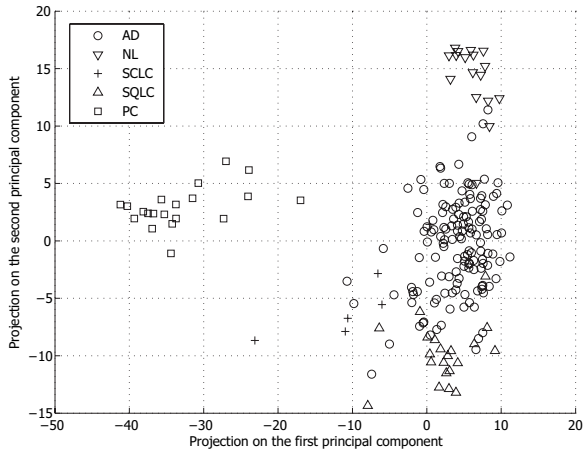


Fig. 3. Lung cancer dataset.

A. Microarray Data

The algorithms were applied to gene expression data from two recently published microarray studies: the leukemia dataset [4], and the lung cancer dataset [1]. The leukemia dataset comes from the study of gene expression in the three types of acute leukemia: B-cell acute lymphoblastic leukemia (ALLB), T-cell acute lymphoblastic leukemia (ALLT) and acute myeloid leukemia (AML). The gene expression levels were measured using Affymetrix high-density oligonucleotide microarrays containing 6817 human genes. The data consist of 38 cases of ALL B-cell, 9 cases of ALL T-cell and 25 cases of AML. According to [4], three preprocessing steps were applied to the data. First, a floor of 100 and a ceiling of 16000 was set; second, the data were filtered to include only genes with $max/min > 5$ and $(max - min) > 500$, where min and max corresponds to the minimum and maximum value in a single row, indicating a single gene; and third, the data were transformed to base 10 logarithms. The lung cancer dataset comes from a study of gene expression in the five types of lung cancer. The gene expression levels

were measured using Affymetrix high-density oligonucleotide microarrays containing 12600 human genes. The data consist of 139 cases of lung adenocarcinoma (AD), 21 cases of squamous cell lung carcinoma (SQLC), 20 cases of pulmonary carcinoids (PL), 6 cases of smallcell lung carcinoma (SCLC) and 17 cases of normal lung (NL). The preprocessing steps performed to the data were the same as in case of leukemia dataset described above.

Microarray experiments allow to determine the expression levels of thousands of genes, however a nearly constant expression level of gene for all samples is a common situation. Those genes are not likely to be useful in cluster analysis; therefore, we exclude the low variance genes from the clustering process. In this paper, the 200 most variable genes were used to analyze the leukemia, and the 500 most variable genes were used for the lung cancer dataset. For visualization purposes we use a well known dimension reduction method called principal component analyze (PCA). A visualization of the samples distribution from the leukemia and the lung cancer datasets on the plane of the two first principal components is presented in Fig. 2 and 3.

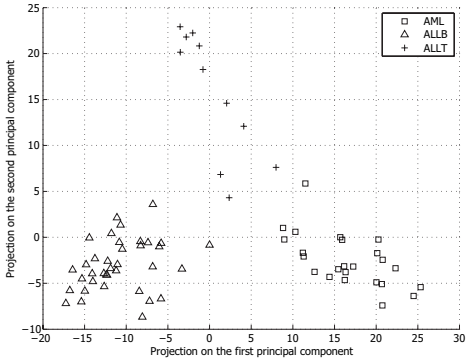
B. Results

The four methods: k-means, SOM, QT and DBSCAN were applied to estimate the number of clusters for each of the two microarray datasets. The experimental results are listed in Table 1. Fig. 4 and 5 display the clustering results on the plane of the two first principal components for leukemia and lung cancer datasets respectively. For more details, in Table 2 the number of badly classified samples (b) and the number of noise samples (n) calculated for both datasets are presented. The numbers in brackets in Table 2 show the number of badly classified samples while k-means and SOM clustering algorithms performed with the correct number of clusters. The DBSCAN and QT uses the same value of $minPts$ parameter: 5 for the leukemia dataset and 3 for the lung cancer dataset. Lesser value of $minPts$ used in the lung cancer dataset is necessary to discover a small SCLC class. In DBSCAN method, we estimate the Eps value assuming 20% of the noise to the all samples ratio, which corresponds to Eps equal to 20.0 and 3.5 in the leukemia and the lung cancer datasets respectively. Similar results were obtained with Eps value calculated from the range of 15% to 25% of the noise to the all samples ratio. In QT method, the best results were obtained with the jackknife correlation threshold 0.7 in both datasets. Changing the threshold in QT from 0.65 to 0.75 does not change the number or size of clusters.

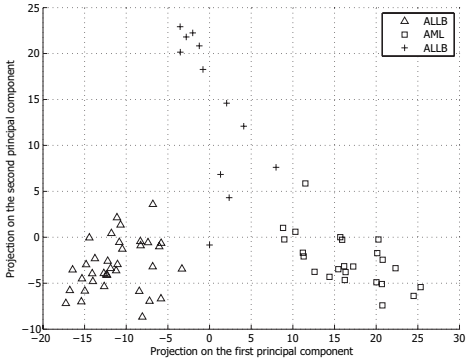
All the methods estimate correctly the presumed number of classes for leukemia dataset, but only DBSCAN does not mismatch the samples from different classes. The results show

TABLE I
ESTIMATING THE NUMBER OF CLUSTERS FROM MICROARRAY DATA

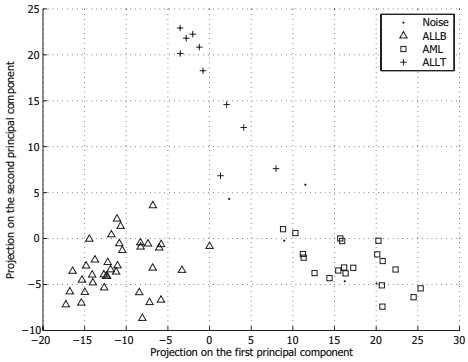
	k-means	SOM	QT	DBSCAN	Correct
Leukemia	3	3	3	3	3
Lung cancer	2	2	4	4	5



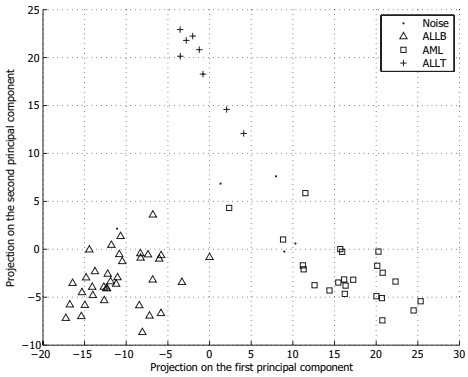
(a) K-means



(b) SOM

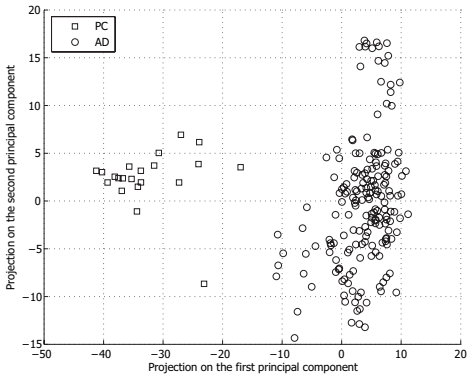


(c) QT

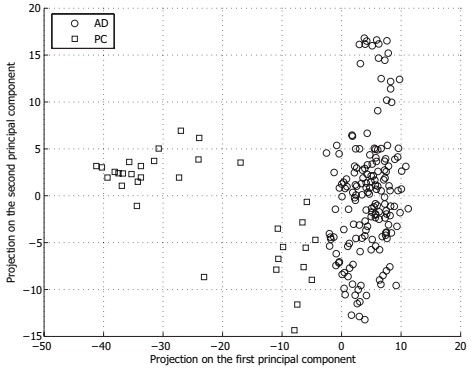


(d) DBSCAN

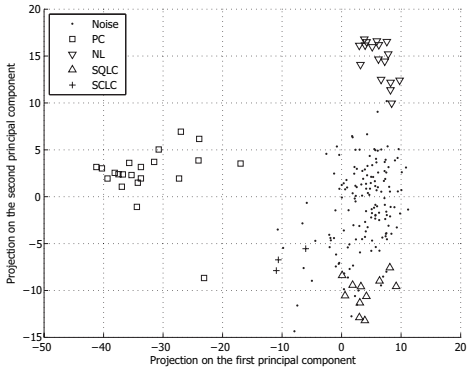
Fig. 4. Results of different clustering algorithms for leukemia dataset.



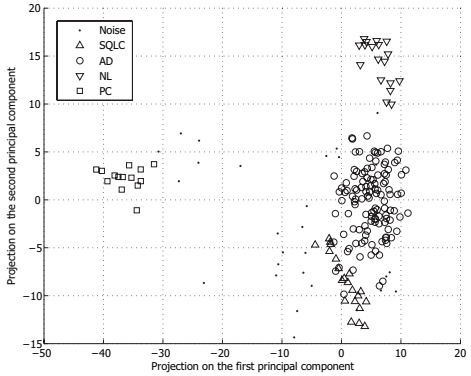
(a) K-means



(b) SOM



(c) QT



(d) DBSCAN

Fig. 5. Results of different clustering algorithms for lung cancer dataset.

TABLE II
THE NUMBER OF BADLY CLASSIFIED SAMPLES (B) AND NUMBER OF NOISE SAMPLES (N) FOR TWO MICROARRAY DATASETS

	Leukemia		Lung cancer	
	B	**N**	**B**	**N**
k-means	2	-	44(76)	-
SOM	3	-	50(95)	-
QT	1	5	2	151
DBSCAN	0	5	14	29

that clustering of lung cancer dataset was a much difficult task than leukemia dataset for all of the methods. In this studies, DBSCAN and QT performs significantly better than the k-means and the SOM, that merged a different clusters to finally produce only two out of five existing groups. The experimental results show that the k-means and SOM are very sensitive to the outliers. The main disadvantage of the k-means and SOM approach is the assigning of every sample to a cluster.

The QT algorithm has a problem to define a large and vast cluster (AD group in the lung cancer dataset), because every pair of samples in a cluster have to have enough jackknife correlation value. If the prespecified jackknife correlation threshold is smaller than 0.65, many small clusters are created instead of one AD group. On the other hand, only QT algorithm successfully discovers SCLC group in the lung cancer dataset, matching correctly three out of six samples from that group. The DBSCAN has an opposite problem. While discovering a large cluster is not difficult, the density based clustering does not successfully match small clusters like SCLC group. It is noteworthy, that small SCLC group falls into the noise cluster as outliers. In DBSCAN, the noise cluster requires an additional inspection, while the outliers might form an undiscovered group. In case of QT method, the noise cluster is too large to proceed an additional investigation (over 150 samples). This is the main reason why DBSCAN results are much better than the QT and the others.

V. CONCLUSIONS

This work reports the application of DBSCAN algorithm that proved to be useful in determining the number of clusters in a microarray experiments. The density based clustering algorithm was confirmed to help in identification of the correct clusters in the data set. The DBSCAN differs from related works in several aspects. The density-based method is insensitive to the shape of a cluster and to the outlier effect. The DBSCAN successfully discovers all clusters in the presented two microarray studies beside the smallest one in the lung cancer dataset (SCLC). As it was shown, the small SCLC group is a part of the noise cluster defined by the DBSCAN. Only with the DBSCAN method, an additional investigation of a noise cluster leads to the correct decision about the number of clusters in the dataset. In the comparative studies, DBSCAN was found to give better results than the three other methods. The results show that DBSCAN may be considered as an effective tool for microarray data analysis.

REFERENCES

[1] A. Bhattacharjee, W. G. Richards, J. Staunton, C. Li, S. Monti, P. Vasa, C. Ladd, J. Beheshti, R. Bueno, M. Gillette, M. Loda, G. Weber, E. J. Mark, E. S. Lander, W. Wong, B. E. Johnson, T. R. Golub, D. J. Sugarbaker, and M. Meyerson, "Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses," *Proc. Natl. Acad. Sci. U.S.A.*, vol. 98, pp. 13 790–13 795, Nov 2001.

[2] S. Dudoit and J. Fridlyand, "A prediction-based resampling method for estimating the number of clusters in a dataset," *Genome Biol.*, vol. 3, p. RESEARCH0036, Jun 2002.

[3] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*. AAAI Press, 1996, pp. 226–231.

[4] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, and C. D. Bloomfield, "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, pp. 531–537, 1999.

[5] J. A. Hartigan, "Statistical theory in clustering," *J. Classification*, vol. 2, no. 1, pp. 63–76, 1985. [Online]. Available: http://dx.doi.org/10.1007/BF01908064

[6] L. J. Heyer, S. Kruglyak, and S. Yooseph, "Exploring expression data: identification and analysis of coexpressed genes," *Genome Res.*, vol. 9, pp. 1106–1115, Nov 1999.

[7] S. C. Johnson, "Hierarchical clustering schemes," *Psychometrika*, vol. 32, pp. 241–254, Sep 1967.

[8] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis (Wiley Series in Probability and Statistics)*. Wiley-Interscience, March 2005. [Online]. Available: http://www.worldcat.org/isbn/0471735787

[9] T. Kohonen, *Self-organization and associative memory*, ser. Springer Series in Information Sciences. Berlin: Springer-Verlag, 1984, vol. 8.

[10] W. J. Krzanowski and Y. T. Lai, "A criterion for determining the number of groups in a data set using sum-of-squares clustering," *Biometrics*, vol. 44, no. 1, pp. 23–34, 1988. [Online]. Available: http://dx.doi.org/10.2307/2531893

[11] R. J. Lipshutz, S. P. Fodor, T. R. Gingeras, and D. J. Lockhart, "High density synthetic oligonucleotide arrays," *Nat. Genet.*, vol. 21, pp. 20–24, Jan 1999.

[12] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. Fifth Berkeley Sympos. Math. Statist. and Probability (Berkeley, Calif., 1965/66)*. Berkeley, Calif.: Univ. California Press, 1967, pp. Vol. I: Statistics, pp. 281–297.

[13] E. E. Schadt, C. Li, C. Su, and W. H. Wong, "Analyzing high-density oligonucleotide gene expression array data," *J. Cell. Biochem.*, vol. 80, pp. 192–202, Oct 2000.

[14] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *Journal Of The Royal Statistical Society Series B*, vol. 63, no. 2, pp. 411–423, 2001. [Online]. Available: http://ideas.repec.org/a/bla/jorssb/v63y2001i2p411-423.html