Statistica Sinica **12**(2002), 241-262

EVALUATION AND COMPARISON OF CLUSTERING ALGORITHMS IN ANGLYZING ES CELL GENE EXPRESSION DATA

Gengxin Chen[1], Saied A. Jaradat[2], Nila Banerjee[1],

Tetsuya S. Tanaka[2], Minoru S. H. Ko[2] and Michael Q. Zhang[1]

[1]*Cold Spring Harbor Laboratory and*
[2]*National Institutes of Health, U.S.A.*

Abstract: Many clustering algorithms have been used to analyze microarray gene expression data. Given embryonic stem cell gene expression data, we applied several indices to evaluate the performance of clustering algorithms, including hierarchical clustering, k-means, PAM and SOM. The indices were homogeneity and separation scores, silhouette width, redundant score (based on redundant genes), and WADP (testing the robustness of clustering results after small perturbation). The results showed that the ES cell dataset posed a challenge for cluster analysis in that the clusters generated by different methods were only partially consistent. Using this data set, we were able to evaluate the advantages and weaknesses of algorithms with respect to both internal and external quality measures. This study may provide a guideline on how to select suitable clustering algorithms and it may help raise issues in the extraction of meaningful biological information from microarray expression data.

Key words and phrases: Cluster analysis, gene expression, microarray, mouse embryonic stem cell.

1. Introduction

DNA microarray technology has proved to be a fundamental tool in studying gene expression. The accumulation of data sets from this technology that measure the relative abundance of mRNA of thousands of genes across tens or hundreds of samples has underscored the need for quantitative analytical tools to examine such data. Due to the large number of genes and complex gene regulation networks, clustering is a useful exploratory technique for analyzing these data. It divides data of interest into a small number of relatively homogeneous groups or clusters. There can be at least two ways to apply cluster analysis to microarray data. One way is to cluster arrays, i.e., samples from different tissues, cells at different time points of a biological process or under different treatments. This type of clustering can classify global expression profiles of different tissues or cellular states. Another use is to cluster genes according to their expression levels across different conditions. This method intends to group co-expressed genes and to reveal co-regulated genes or genes that may be involved in the same complex or pathways. In our study, we focused on the latter method.

Many clustering algorithms have been proposed for studying gene expression data. For example, Eisen, Spellman, Brown and Botstein (1998) applied a variant of the hierarchical average-linkage clustering algorithm to identify groups of co-regulated yeast genes. Tavazoie et al. (1999) reported their success with k-means algorithm, an approach that minimizes the overall within-cluster dispersion by iterative reallocation of cluster members. Tamayo et al. (1999) used self-organizing maps (SOM) to identify clusters in the yeast cell cycle and human hematopoietic differentiation data sets. There are many others. Some algorithms require that every gene in the dataset belongs to one and only one cluster (i.e., generating exhaustive and mutually exclusive clusters), while others may generate "fuzzy" clusters, or leave some genes unclustered. The first type is most frequently used in the literature and we restrict our attention to them here.

The hardest problem in comparing different clustering algorithms is to find an algorithm-independent measure to evaluate the quality of the clusters. In this paper, we introduce several indices (homogeneity and separation scores, silhouette width, redundant scores and WADP) to assess the quality of k-means, hierarchical clustering, PAM and SOM on the NIA mouse 15K microarray data. These indices use objective information in the data themselves and evaluate clusters without any *a priori* knowledge about the biological functions of the genes on the microarray. We begin with a discussion of the different algorithms. This is followed by a description of the microarray data pre-processing. Then we elaborate on the definitions of the indices and the performance measurement results using these indices. We examine the difference between the clusters produced by different methods and their possible correlation to our biological knowledge. Finally, we discuss the strength and weakness of the algorithms revealed in our study.

2. Clustering Algorithms and Implementation

2.1. K-means

K-means is a well-known partitioning method. Objects are classified as belonging to one of k groups, k chosen *a priori.* Cluster membership is determined by calculating the centroid for each group (the multidimensional version of the mean) and assigning each object to the group with the closest centroid. This approach minimizes the overall within-cluster dispersion by iterative reallocation of cluster members (Hartigan and Wong (1979)).

In a general sense, a k-partitioning algorithm takes as input a set S of objects and an integer k, and outputs a partition of S into subsets $S_1, \ldots, S_k$. It uses the sum of squares as the optimization criterion. Let x_r^i be the rth element of S_i, $|S_i|$ be the number of elements in S_i, and $d(x_r^i, s_s^i)$ be the distance between x_r^i and x_s^i. The sum-of-squares criterion is defined by the cost function $c(S_i) = \sum_{r=1}^{|S_i|}\sum_{s=1}^{|S_i|}(d(x_r^i, x_s^i))^2$. In particular, k-means works by calculating the centroid of each cluster S_i, denoted $\bar{x}^i$, and optimizing the cost function $c(S_i) = \sum_{r=1}^{|S_i|}(d(\bar{x}^i, x_r^i))^2$. The goal of the algorithm is to minimize the total cost: $c(S_1) + \cdots + c(S_k)$.

The implementation of the k-means algorithm we used in this study was the one in S-plus (MathSoft, Inc.), which initializes the cluster centroids with hierarchical clustering by default, and thus gives deterministic outcomes. The output of the k-means algorithm includes the given number of k clusters and their respective centroids.

2.2. PAM (Partitioning around medoids)

Another k-partitioning approach is PAM, which can be used to cluster the types of data in which the mean of objects is not defined or available (Kaufman and Rousseuw (1990)). Their algorithm finds the *representative object* (i.e., medoid, which is the multidimensional version of the median) of each S_i, denoted $\hat{x}_i$, uses the cost function $c(S_i) = \sum_{r=1}^{|S_i|} d(\hat{x}^i, x_r^i)$, and tries to minimize the total cost.

We used the implementation of PAM in the S-plus. PAM finds a local minimum for the objective function, that is, a solution such that there is no single switch of an object with a medoid that will decrease the total cost.

2.3. Hierarchical clustering

Partitioning algorithms are based on specifying an initial number of groups, and iteratively reallocating objects among groups to convergence. In contrast, hierarchical algorithms combine or divide existing groups, creating a hierarchical structure that reflects the order in which groups are merged or divided. In an agglomerative method, which builds the hierarchy by merging, the objects initially belong to a list of singleton sets $S_1, \ldots, S_n$. Then a cost function is used to find the pair of sets $\{S_i, S_j\}$ from the list that is the "cheapest" to merge. Once merged, S_i and S_j are removed from the list of sets and replaced with $S_i \cup S_j$. This process iterates until all objects are in a single group. Different variants of agglomerative hierarchical clustering algorithms may use different cost functions. Complete linkage, average linkage, and single linkage methods use maximum, average, and minimum distances between the members of two clusters, respectively.

In the present study, we used the implementation of average linkage hierarchical clustering in the S-plus package.

2.4. SOM (Self-organization map)

Inspired by neural networks in the brain, SOM uses a competition and co-operation mechanism to achieve unsupervised learning. In the classical SOM, a set of nodes is arranged in a geometric pattern, typically 2-dimensional lattice. Each node is associated with a weight vector with the same dimension as the input space. The purpose of SOM is to find a good mapping from the high dimensional input space to the $2-D$ representation of the nodes. One way to use SOM for clustering is to regard the objects in the input space represented by the same node as grouped into a cluster. During training, each object in the input is presented to the map and the best matching node is identified. Formally, when input and weight vectors are normalized, for input sample $x(t)$ the winner index c (best match) is identified by the condition:

$$\text{for all } i,\ ||x(t) - m_c(t)|| \leq ||x(t) - m_i(t)||,$$

where t is the time step in the sequential training, m_i is the weight vector of the ith node. After that, weight vectors of nodes around the best-matching node $c = c(x)$ are updated as $m_i(t+1) = m_i(t) + \alpha h_{c(x),i}(x(t) - m_i(t))$ where α is the learning rate and $h_{c(x),i}$ is the "neighborhood function", a decreasing function of the distance between the ith and cth nodes on the map grid. To make the map converge quickly, the learning rate and neighborhood radius are often decreasing functions of t. After the learning process finishes, each object is assigned to its closest node. There are variants of SOM implementation.

We used the implementation in the SOM Toolbox for Matlab developed by the Laboratory of Information and Computer Science in the Helsinki University of Technology (http://www.cis.hut.fi/projects/somtoolbox/) and adopted the initialization and training methods suggested by the authors that allows the algorithm to converge faster. That is, the weight vectors are initialized in an orderly fashion along the linear subspace spanned by the first two principal components of the input data set. In contrast to the algorithm used in Tamayo et al. (1999), we used a batch-training algorithm implemented in the Toolbox, which is much faster to calculate in Matlab than the normal sequential algorithm, and typically gives just as good or even better results (ref. http://www.cis.hut.fi/projects/somtoolbox/documentation/somalg.shtml). For a batch-training algorithm, learning rate α is not necessary. In our experiments, the radius of the neighborhood function was initialized to be half the lattice edge size and linearly decreased with the training epochs. To allow the SOM network to fully converge, the number of training epochs was set to be proportional to the lattice edge size. With the initialization methods we used, all clustering algorithms studied here are deterministic.

3. Microarray and Data Pre-processing

The microarrays we used were cDNA arrays developed in NIA and representing 15,000 distinct mouse genes (hence named "NIA mouse 15K microarray") (Tanaka et al. (2000)). The cDNA collections were derived from preimplantation mouse embryos and 50% of the represented genes were newly identified. Undifferentiated mouse R1 embryonic stem (ES) cells were induced into differentiation spontaneously upon the withdrawal of leukemia inhibitory factor (LIF) and conditioned media. Total RNAs were extracted from these cells across 6 different time course points ranging from 4 h to 7 days and used for cDNA microarray hybridizations. For each time point, three replicated microarray experiments were done separately.

First, one-way ANOVA was performed to identify genes with significant expression changes during the ES cell differentiation, that is, the expression variations across the time course must be significantly larger than the variations within the triplicates. Using $p < 0.05$ as a filtering criterion, we obtained 3805 genes for further analysis. Next, triplet data at each time point were averaged and the ratio of expression levels of the six different differentiated states to the undifferentiated state were calculated and log-transformed. Since, from a biological point of view, we were primarily interested in the relative up/down-regulation of gene expressions instead of the absolute amplitude changes, Pearson correlation would be an appropriate similarity metric. However, all clustering programs studied here use Euclidean distance as a dissimilarity metric. We normalized each gene expression pattern as a vector to have unit length. After normalization, Euclidean distance between two gene expression patterns has a monotonic relation to their (non-centered) Pearson correlation, and thus the clustering results obtained with our programs were similar to those obtained using Pearson correlation as metric. The input data for cluster analysis consisted of a matrix of dimension 3805 by 6, in which each row vector (expression levels for a particular gene) had length one.

4. Evaluation Indices and Performance Results with ES Cell Data

In this section, we first describe each evaluation index used. Following each description is the performance measurement using that index for the clustering results obtained from different algorithms.

Except for hierarchical clustering, all clustering algorithms analyzed here required setting k in advance (for SOM, k is the number of nodes in the lattice). Determining the "right" k for a data set itself is a non-trivial problem. Here, instead, we compared the performance of different algorithms for different k's in order to examine whether there were consistent differences in the performance of different algorithms, or whether the performances were related to k. To simplify the situation further, we chose k equal to 16, 25, 36, 49 and 64, and the lattices for SOM were all square. To compare hierarchical clustering with other algorithms, we cut the hierarchical tree at different levels to obtain corresponding numbers of clusters. Specific to SOM, we examined two situations where the neighborhood radius approached one or zero. Theoretically, if the neighborhood radius approaches zero, the SOM algorithm approaches the k-means algorithm. However the dynamics of the training procedure may generate different results, and this would be interesting to explore.

4.1. Homogeneity and separation

We implemented a variation of the two indices suggested by Shamir and Sharan (in press): homogeneity and separation. Homogeneity is calculated as the average distance between each gene expression profile and the center of the cluster it belongs to. That is,

$$H_{ave} = \frac{1}{N_{gene}} \sum_i D(g_i, C(g_i)),$$

where g_i is the ith gene and $C(gi)$ is the center of the cluster that g_i belongs to; N_{gene} is the total number of genes; D is the distance function. Separation is calculated as the weighted average distance between cluster centers:

$$S_{ave} = \frac{1}{\sum_{i \neq j} N_{c_i} N_{C_j}} \sum_{i \neq j} N_{c_i} N_{c_j} D(C_i, C_j),$$

where C_i and C_j are the centers of ith and jth clusters, and N_{c_i} and N_{c_j} are the number of genes in the ith and jth clusters. Thus H_{ave} reflects the compactness of the clusters while S_{ave} reflects the overall distance between clusters. Decreasing H_{ave} or increasing S_{ave} suggests an improvement in the clustering results.

We used Euclidean distance as the distance function D. When expression profiles are normalized to have unit length, Euclidean distance and Pearson correlation are equivalent (dis)similarity metrics. However, due to the nonlinear relation between the two metrics, the weighted average of one metric (such as in S_{ave}) may behave differently from another. Since all algorithms in the study used Euclidean distance as the dissimilarity metric, we thought it appropriate to use Euclidean distance in the quality indices as well.

We should also point out that H_{ave} and S_{ave} are not independent of each other: H_{ave} is closely related to within-cluster variance, S_{ave} is closely related to between-cluster variance. For a given data set, the sum of within-cluster variance and between-cluster variance is a constant.

The homogeneity of the clusters for algorithms studied is shown in Figure 1(a). The performances of k-means and PAM were almost identical. When the neighborhood radius was set to approach zero (SOM_r_0), SOM performed as well as k-means and PAM. In contrast, when the neighborhood radius was set to approach one (SOM_r_1), the homogeneity index of the clusters obtained by SOM was not as good as those of k-means and PAM for all k's tested. Average linkage hierarchical clustering was the worst with regard to homogeneity. Figure 1(b) shows the separation of the clustering results. Consistent with homogeneity, k-means and PAM performed as well as SOM_r_0, and all were better than average linkage clustering. However, SOM_r_1 appeared the worst with regard to this index.

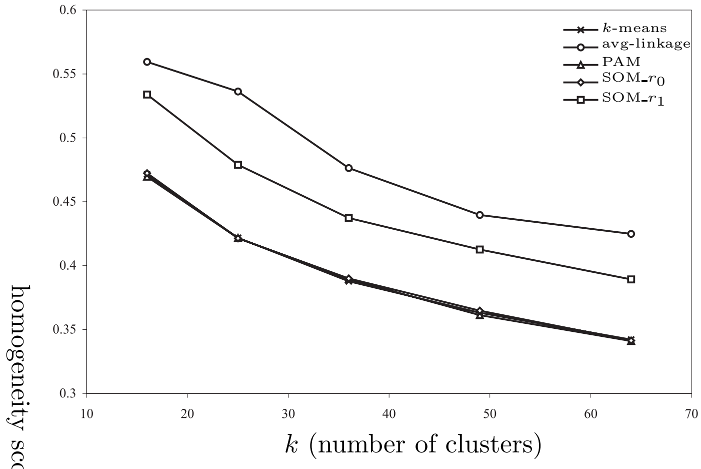


Figure 1a. Homogeneity score for clustering outputs of k-means, avg-linkage, PAM, SOM_r_0 and SOM_r_1 across $k = 16, 25, 36, 49$ and 64.

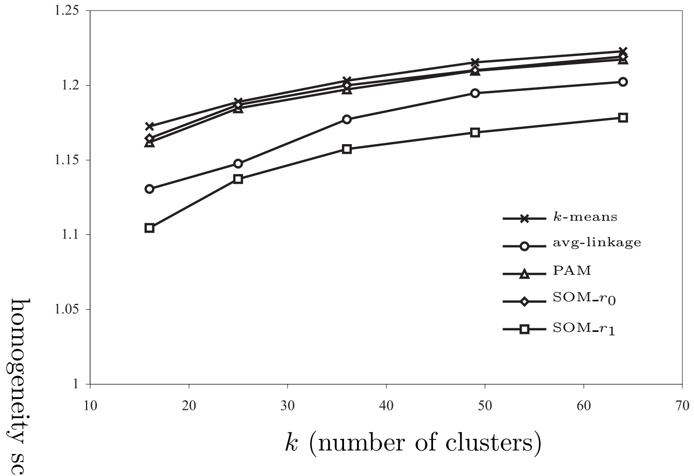


Figure 1b. Separation score for clustering outputs among k-means, avg-linkage, PAM, SOM_r_0 and SOM_r_1.

4.2. Silhouette width

The second index we used to evaluate clustering results was the silhouette width proposed by Rousseeuw (1987) (also MathSoft, Inc. (1998, Chap. 20), Vilo et al. (2000)). *Silhouette width* is a composite index reflecting the compactness and separation of the clusters, and can be applied to different distance metrics. For each gene i, its silhouette width $s(i)$ is defined as

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}},$$

where $a(i)$ is the average distance of gene i to other genes in the same cluster, $b(i)$ is the average distance of gene i to genes in its nearest neighbor cluster. The average of $s(i)$ across all genes reflects the overall quality of the clustering result. A larger *averaged silhouette width* indicates a better overall quality of the clustering result.

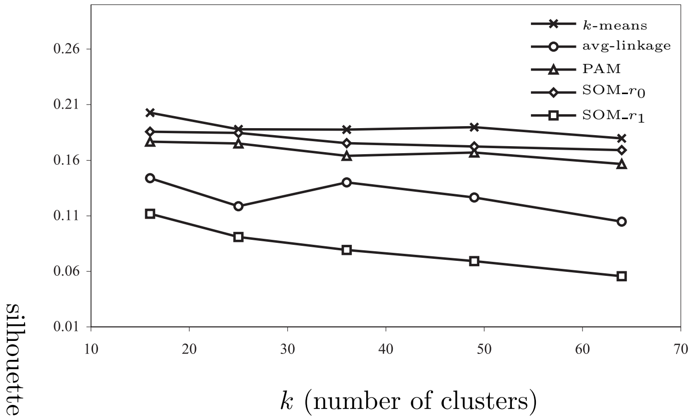


Figure 2. Average silhouette width for clustering outputs among k-means, avg-linkage, PAM, SOM_r_0 and SOM_r_1.

Figure 2 shows the *averaged silhouette widths* obtained in our study. The score for k-means was very close to those for PAM and SOM_r_0, which were slightly better than average linkage. Again, SOM_r_1 had the lowest score. It should be noted that the scores for all the clustering methods in this study were below 0.2, which is rather low, suggesting the clusters might not be well separated and the underlying structure in our expression data was likely "blurry".

4.3. Redundant scores

In our ES cell data set, there was a small portion of redundant genes, i.e., some cDNA clones on the chip actually represented the same gene. After filtering as described previously, there were 253 such clones, which represented 104 genes. These included duplicates, triplicates, up to quintuplicates. Since identical cDNA clone probes should give similar expression patterns (aside from experimental noise), a good cluster result should cluster those redundant genes together with high probability. We tried to make use of these redundant genes to measure the quality of our clustering results, by calculating a separation score $RSS = \sum_g \frac{C_g}{R_g}$, where R_g is the number of clones in a redundant group g, C_g is the number of clusters these clones are separated into. Ideally, C_g should be one for every redundant group g. Because this score is biased to favor small number of clusters, we also calculated a control score with 253 randomly picked genes put into the same 104 groups. The difference of redundant separation scores (DRSS) between the control and redundant gene sets was used as a measurement of clustering quality. If this score is high, it suggests that the redundant genes are more likely to be clustered together than randomly picked genes.

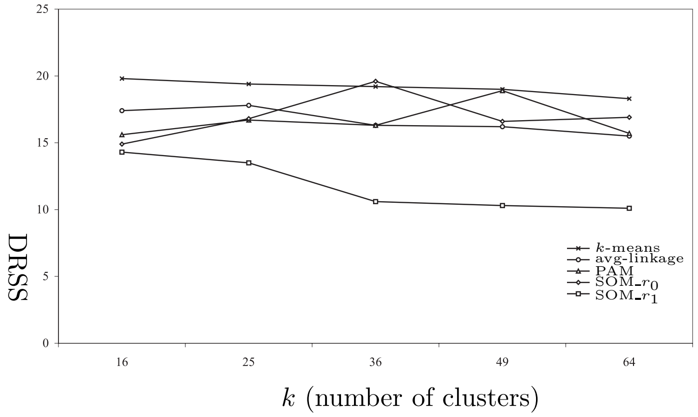


Figure 3. Difference of redundant separation scores (DRSS) for clustering outputs among k-means, avg-linkage, PAM, SOM_r_0 and SOM_r_1.

Redundant scores for the clustering results are given in Figure 3. Here, k-means appeared to perform better than average linkage clustering consistently through all k's tested. Redundant scores for SOM_r_1 tended to be lower than those of other algorithms, especially when k was relatively large. PAM and SOM_r_0 were intermediate to k-means and average linkage clustering, without obvious and consistent relation to them or to each other.

One cautionary point should be made. The DRSS scores in Figure 3 suggest that for all methods, a portion of the redundant genes were not clustered together. Besides the measurement noise and sample preparation variations in the experiments, an important factor is clone identity. The clones were verified with complete or partial sequencing and BLAST against the GenBank nr repository. Two clones were considered identical if they hit the same GenBank record with high enough scores in BLAST. However, it is possible that two clones contain homologous genes, of which one is not characterized and deposited into GenBank, and thus they both map to the same gene in GenBank. When we examined the clustering results, we found several cases where a "redundant" pair of clones had quite different BLAST scores and were separated into different clusters. Those "redundant" pairs of clones might not really be identical clones. Nevertheless, the tendency of the "redundant" genes to be clustered together was significantly larger than for randomly picked control genes. The difference between the scores of "redundant" genes and the mean scores of control genes was typically more than two or three times the standard deviation of the control scores.

4.4. WADP

A critical issue is the robustness of clustering results. That is, if input data points deviate slightly from their current values, will we get the same clustering? This is important in microarray expression data analysis because there is always experimental noise in the data. A good clustering result should be insensitive to the noise and able to capture the real structure in the data, reflecting of the biological processes under investigation. To test the robustness of the results obtained from different algorithms, we used the method proposed by Bittner et al. (2000). Briefly, each gene expression profile was perturbed by adding a random vector of the same dimension. Each element of the random vector was generated from a Gaussian distribution with mean zero. We used standard deviation $\sigma = 0.01$ for the perturbation, preliminary observation suggested that this level of perturbation was relatively representative. After re-normalization of the perturbed data, clustering was performed. For each individual cluster, a cluster-specific discrepancy rate was calculated as D/M. That is, for the M pairs of genes in an original cluster, count the number of gene pairs, D, that do not remain together in the clustering of the perturbed data, and take their ratio. The overall discrepancy rate for the clustering is calculated as the weighted average of those cluster-specific discrepancy rates. This process was repeated many times and the average overall discrepancy rate, the weighted average discrepant pairs (WADP) was obtained (see Supplementary Information in Bittner et al. (2000)). WADP equals zero when two clustering results match perfectly. In the worst case, WADP is close to one.

Figure 4 shows the clustering robustness as measured with WADP, in which clusters obtained with SOM_r_1 appeared to be significantly more stable than all the other algorithms. WADP scores for k-means and average linkage were relatively high regardless of k, and were not much different from each other. WADP scores for PAM and SOM_r_0 appeared to be related to k. When k was 16 and 25, the clustering results with PAM and SOM_r_0 were relatively more stable than k-means and average linkage. When k was large, the clustering stability of PAM and SOM_r_0 were about the same as k-means and average linkage.

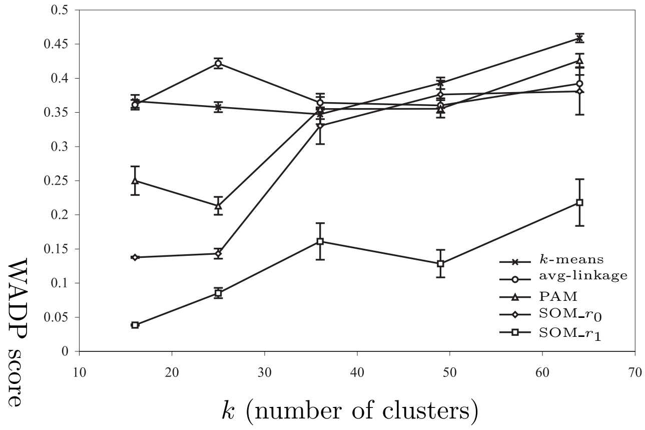
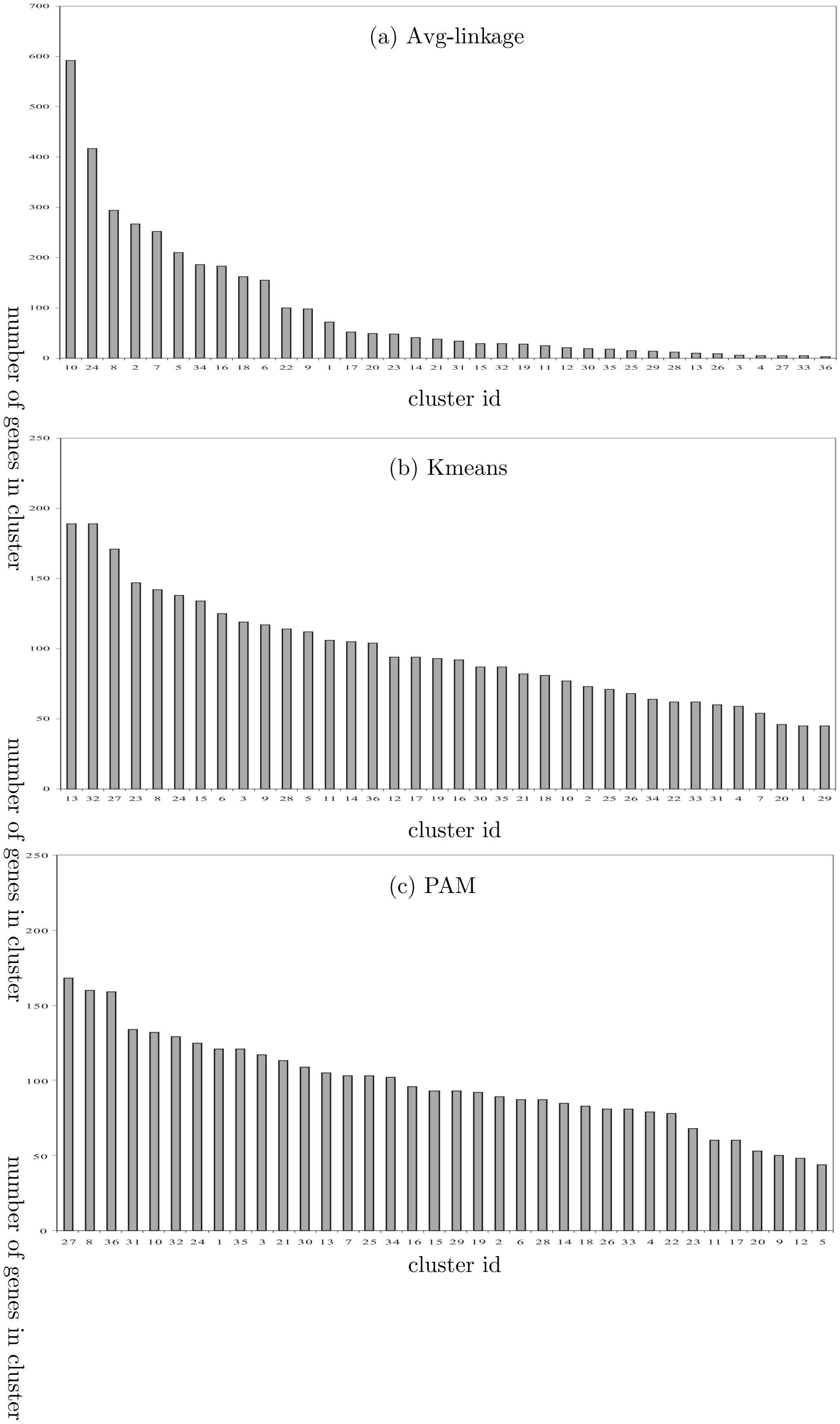


Figure 4. WADP (weighted average discrepancy pair) score for clustering outputs among k-means, avg-linkage PAM, SOM_r_0 and SOM_r_1. For all algorithms except PAM the results were averaged over 40 runs, while for PAM, results were averaged over 10 runs due to its slowness. The error bars show the standard error of means.

5. Comparison of Cluster Sizes and Consistency

One issue that may be related to the structural quality of clusters is the cluster size distribution (number of genes in each cluster). Figure 5(a)-(e) show the cluster sizes for each method in our study, with k equal to 36. Average linkage clustering tended to give variable sizes of clusters, a few large clusters containing hundreds of genes and many small clusters having only a few genes (note the scale of y-axis in Figure (a) is different from all the other). Cluster sizes for PAM and SOM_r_0 appeared to vary the least. The cluster size variability of k-means was close to that of PAM and SOM_r_0, while the variability of SOM_r_1 was somewhat larger but better than average linkage. There appeared to be a systematic bias in the cluster sizes related to the location of the nodes in the SOM lattice when the neighborhood interaction was maintained as in SOM_r_1. That is, clusters represented by the nodes at the corners or edges (such as cluster 6, 36 and cluster 32, 13, respectively) of the SOM lattice tended to have more genes than those represented by the inner nodes. Having some large, not necessarily dense, clusters due to its "greedy" algorithm might be a possible reason that average linkage scored poorly in homogeneity.

To compare the consistency of clusters produced by different methods, we again adopted WADP as a measurement. Because WADP puts the number of


(a) Avg-linkage
700
600
500
400
300
200
100
0
10 24 8 2 7 5 34 16 18 6 22 9 1 17 20 23 14 21 31 15 32 19 11 12 30 35 25 29 28 13 26 3 4 27 33 36
cluster id
number of genes in cluster
(b) Kmeans
250
200
150
100
50
0
13 32 27 23 8 24 15 6 3 9 28 5 11 14 36 12 17 19 16 30 35 21 18 10 2 25 26 34 22 33 31 4 7 20 1 29
cluster id
number of genes in cluster
(c) PAM
250
200
150
100
50
0
27 8 36 31 10 32 24 1 35 3 21 30 13 7 25 34 16 15 29 19 2 6 28 14 18 26 33 4 22 23 11 17 20 9 12 5
cluster id
number of genes in cluster

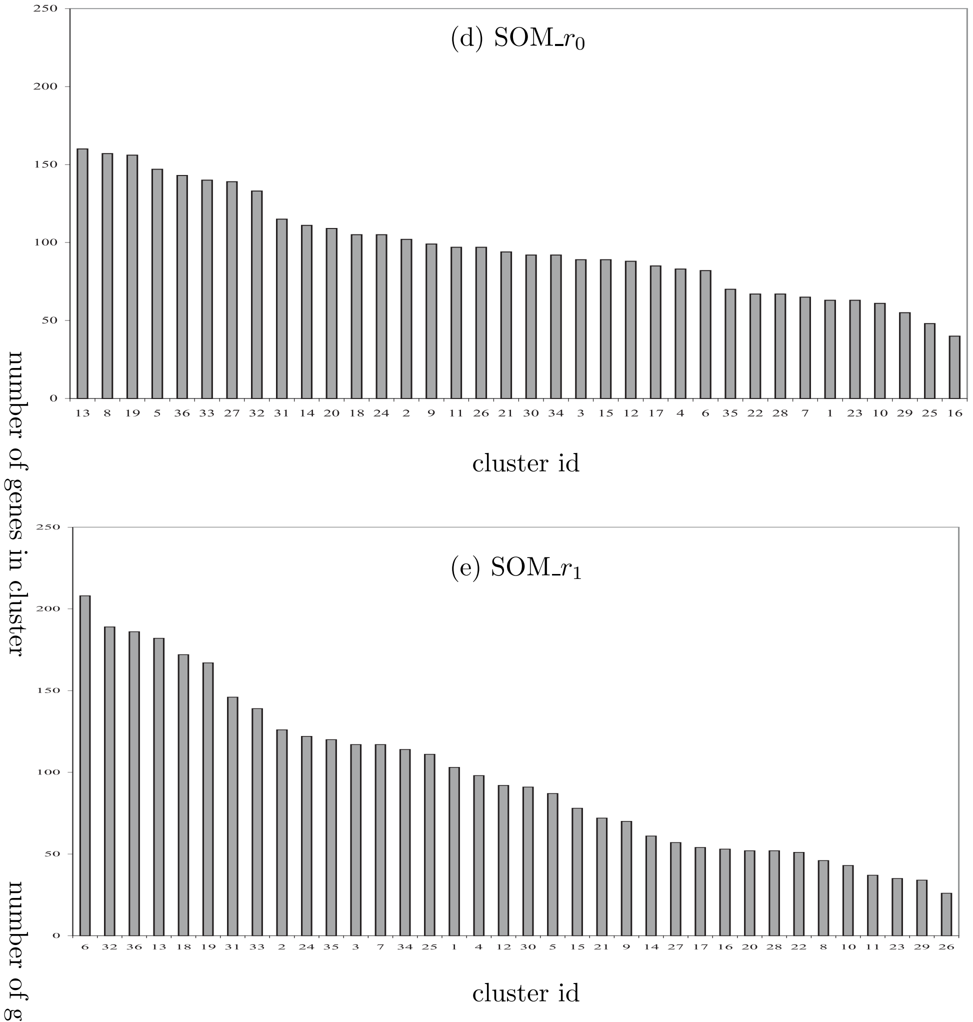


Figure 5. The cluster sizes for each method in our study when k was equal to 36.

pairs of genes in the first cluster result in the denominator, it is not symmetric, i.e. WADP(A, B) is typically not WADP(B, A). Thus, we used the average of WADP(A, B) and WADP(B, A) as the distance between cluster method A and B. Based on this distance, a hierarchical tree was built to display the similarity or dissimilarity of clusters generated by different algorithms. Figure 6 shows the result when k was 36. It can be seen that k-means was similar to PAM, while average linkage and SOM_r_1 tended to produce clusters not overlapping with those of other methods. However, note that even the distance between k-means and PAM was larger than 0.45, which meant more than 45% of gene pairs in one clustering result were separated by the other method. This suggests that clustering results from different methods were only partially consistent, and that caution needs to be taken when we interpret these results.

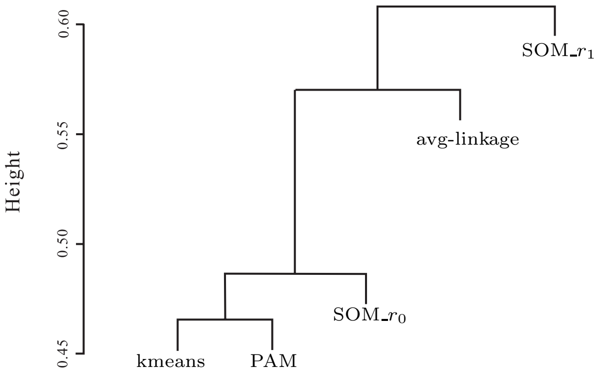


Figure 6. The hierarchical tree generated with average linkage using average discrepancy rate of gene pairs as distance between clustering results of different methods. The tree height represents the distance between the two merging nodes.

6. Biological Interpretation of the Clusters

The biological functions of several genes, as well as their interaction in certain pathways governing the ES cell pluripotency, have been identified (Jaradat et al. (to be submitted)). The *Pou*5*f*1(*Oct*-3/4) gene, which encoded the transcription factor Oct-3/4 and expressed specifically in totipotent embryonic cells and germ cells (reviewed by Pesce and Scholer (2000)), is widely accepted as a marker that measures the pluripotency of ES cells. In our data, Oct-3/4 down regulated immediately in response to the withdrawal of LIF and the conditioned media, as shown in Figure 7(a). The down regulation of other genes, of which many are unknown, at both 4 hours and 8 hours post-LIF withdrawal suggested these genes might carry a similar function to Oct-3/4, or that they might be used as alternative markers for ES cell pluripotency. Two examples of these genes are *p*45 *Nf*-*e*2 and *Baff*. Both p45 Nf-e2 and Baff are transcription factors important in erythroid and lymphocyte lineages, respectively (Chui, Tang and Orkin (1995), Schneider et al. (1999)). In combination with an unidentified protein complex called Rox-1, Oct-3 enhanced the expression of the *Zfp*42 gene, which encoded an acidic zinc finger protein named Rex-1 (Ben-Shushan, Thompson, Gudas and Bergman (1998)). Finally, Oct-3/4 and *Hmg1* have been reported to interact with each other at the protein level (Butteroni, De Felici, Scholer and Pesce (2000)). There were two copies of Hmg1 genes (H3027D07, H3059H04, http://lgsun.grc.nia.nih.gov/) in our data set. Another group of genes that exert similar functions included *Ezh*2*, rae*-28 and *Cytocine*-5-*methyl transferase* 3.

All of these three genes play an important role in suppression mechanism at the genomic levels (reviewed in Satijn and Otte (1999)).

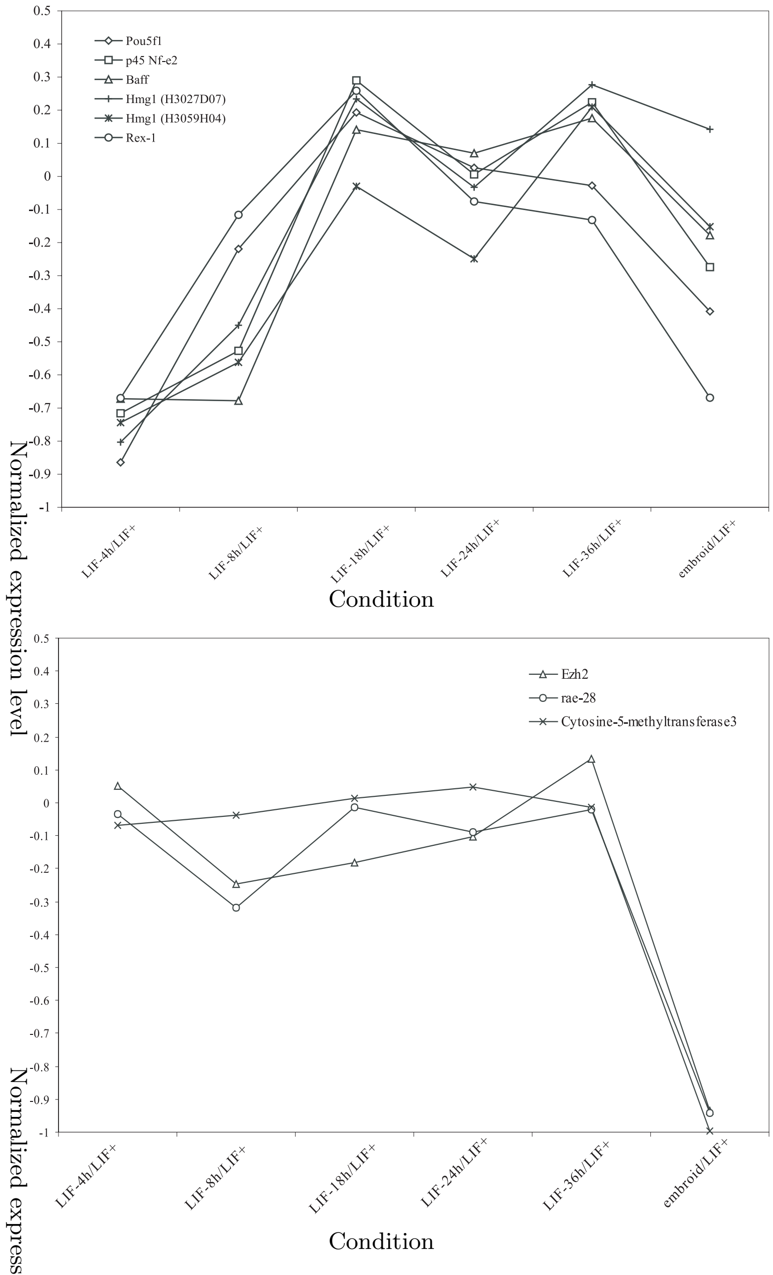


Figure 7. The normalized expression profiles of two groups of functionally related genes: (a) group of genes related to Oct-3/4; (b) three genes with a role in suppression mechanism at the genomic levels.

The expression profiles of these two groups of genes are displayed in Figure 7(a) and 7(b), respectively. As an example, the locations of those genes in the clusters produced by each method (when $k = 36$) are listed in Table 1. It can be seen that five out of six genes in the first group were grouped together in cluster #27 by k-means. They were also in the same cluster (#31) according to SOM_r_0. In addition, note that although the six genes were placed in three different clusters by SOM_r_1, those three clusters were represented by three adjacent nodes in the SOM lattice. The three genes in the second group were clustered together by three of the methods we applied and the other two methods grouped two genes together.

Table 1. Two groups of functionally related genes and their locations in clusters ($k = 36$).

Clone	k-means	average linkage	PAM	SOM_r_0	SOM_r_1	Description
H3028H01	27	1	35	31	25	Mus musculus POU domain, class 5, transcription factor 1 (Pou5f1), mRNA
H3054B12	27	12	35	31	31	Mus musculus p45 NF-E2 related factor 2 (Nrf 2) mRNA, complete cds
H3053A01	27	12	30	31	31	Mus musculus B-cell activating factor (Baff) mRNA, complete cds
H3027D07	27	1	30	31	31	Mus musculus high mobility group protein 1 (Hmg1), mRNA
H3059H04	27	12	8	31	31	M.musculus HMG1 gene
H3036F04	23	24	36	19	19	Mouse REX-1 mRNA, complete cds
H3141B05	24	24	31	13	13	Mus musculus enhancer of zeste homolog 2 (Drosophila) (Ezh2), mRNA
H3105A03	24	24	31	13	13	rae-28=polyhomeotic gene homolog clone Rae-2812 [mice, embryonal carcinoma F9 cells, mRNA, 3542 nt]
H3094C02	24	24	25	13	7	Mus musculus partial mRNA for carcinoma-5-methyltransferase 3-like protein (Dnmt3l gene)

The numbers in each column are the cluster ID's determined by each clustering program, respectively. For SOM, the cluster ID numbers correspond to the locations of the nodes in the lattice, with #1, #6, #31 and #36 at the four corners. For other algorithms, there are no particular relations between the cluster ID's.

To further access the biological meaning of the clusters, we examined the distribution of sets of functionally classified genes. Among the 15K cDNA clones on the microarray, 4027 clones were functionally classified according to their homology to know genes or sequence match to know functional motifs of proteins

(Kargul et al. (2001)). Those genes were in nine gross functional categories, such as apoptosis, cell cycle, etc. After the filtering process described previously, 1279 out of the 3805 genes used in clustering were assigned to those functional categories. Among the nine functional categories, five categories contained more than 100 genes (see Table 2). The other four categories were ignored in the following analysis since sample sizes were small.

Table 2. X^2 scores of clustering results based on functional categories ($k = 36$).

	X^2 Score				
Functional Category (gene number)	k-means	linkage (gene number)	PAM	SOM_r_0	SOM_r_1
Energy/Metabolism ($n = 201$)	36.9	37.8	48^*	52.7^*	65.7^{**}
Matrix/Structural Proteins ($n = 298$)	64.5^{**}	58.8^{**}	63.8^{**}	70.7^{**}	67.2^{**}
Protein Synthesis /Translational Control ($n = 262$)	96.1^{**}	98.6^{**}	83.2^{**}	77.8^{**}	81.8^{**}
Signal Transduction ($n = 220$)	38.4	31.8	38.6	53.6^{**}	43.8
Transcription/Chromatin ($n = 159$)	27.0	41.7	26.9	37.0	28.3

$^*p < 0.05$
$^{**}p < 0.01$

For each category of genes, we calculated a X^2 score for each clustering result as

$$X^2 = \sum_c \frac{(O_c - E_c)^2}{E_c},$$

where O_c is the observed frequency of genes in a cluster c, and E_c is the expected frequency of genes in that cluster based on cluster size distribution. The X^2 scores for the clustering results of the five methods we used (when $k = 36$) are shown in Table 2. This X^2 score is sometimes referred as a *chi*-square score, but its distribution only approximates the *chi*-square distribution when the sample size (gene number) and the expected frequency E are relatively large. In our study, E was relatively small for some clusters and the cluster size distributions of different clustering results could be quite different (e.g., hierarchical clustering vs. other methods). Therefore, we obtained the levels of statistical significance with a Monte Carlo simulation for each clustering method and functional category. The stars in Table 2 denote the p-value levels based on the data from 1000 random clusterings. For functional category "matrix/structural proteins" and "protein synthesis/translational control", X^2 scores for all five clustering methods reached the $p < 0.01$ significant level, suggesting that the functionally related genes in those two categories had some tendency to be clustered together. For the functional classification of genes, we need to be cautious that on one hand, one gene may have multiple functions and that on the other hand, genes in the same functional category may be involved in different pathways and are turned on/off in different biological processes. Such complicated relationships among genes cannot be captured with a simple classification.

7. Discussion

Our experiments with ES cell data set indicated that the success of the clustering methods we tried was limited, suggesting the intrinsic structure in the data might be blurry. However, the clustering results appeared to reflect certain biological relations among the genes, as shown in Section 6. Different algorithms displayed different properties: k-means generated clusters with slightly better structural quality; k-means and SOM_r_0 appeared more consistent with the biological information implicated in the redundant clones and the several known genes involved in the same pathways. However, k-means was relatively sensitive to noise perturbation in the data. On the other hand, when neighborhood interaction was maintained, SOM gave relatively stable clusters but of relatively low structural quality. Average linkage hierarchical clustering was the worst among the four algorithms in this particular test situation and PAM appeared to be close to k-means.

These results are consistent with recent work of Yeung, Haynor and Ruzzo (in press). They developed a *figure of merit* particularly suitable to time course data and evaluated a number of clustering algorithms with several public microarray data sets. In their report, k-means initialized using average linkage appeared to perform slightly better than k-means initialized randomly. Regardless of the initialization methods, k-means outperformed average linkage clustering most of the time. In almost all cases, single linkage clustering performed poorly, likely due to a "chaining" effect.

The relatively low quality of agglomerative hierarchical clustering (such as average linkage) is probably due the "greediness" of the algorithm — when two similar clusters are merged, it is not possible to do any refinement or correction later.

The neighborhood constraint posed on SOM seemed to have a dual-effect — it helped to improve the stability of the clustering but prevented further optimization in the clustering structure. A comparison of SOM with different neighborhood radius functions revealed a trade-off between the cluster stability and structural quality. Since a unique feature of SOM is the topographic relation between the mapping nodes, we could calculate the topographic error (TE) to measure the topology preservation of the map units (ref. http://www.cis.hut.fi/projects/somtoolbox/documentation/), which appeared to be correlated to the performance of SOM. When the neighborhood interaction was maintained (as in SOM_r_1), TE for SOM was very low, and the clusters obtained were relatively stable but not very compact. When the neighborhood interaction was gradually removed (as in SOM_r_0), TE for SOM was much higher and the clusters obtained became more compact, but at the cost of stability.

Theoretically, the SOM algorithm reduces to k-means if the neighborhood radius is set to zero. This is confirmed in our study. The quality of clusters obtained with SOM_r_0 was very similar to that of k-means, when evaluated with *homogeneity, separation, silhouette width and redundant scores.* However, there were some subtle differences in the WADP scores. When k was relatively small (16 and 25), SOM_r_0 appeared to be more stable than k-means, as shown in Figure 4. When k was 36 or larger, the total average of WADP scores for SOM_r_0 and k-means were close to each other. However, if we looked into the WADP scores for each individual run, we could see a bi-modal distribution with SOM_r_0, which was not present with k-means. (In fact, WADP scores for individual runs for SOM_r_1 also had this kind of bi-modal distribution, but the frequencies at the high score region were much lower.) This bi-modal distribution was also reflected in the relatively large standard errors of WADP scores for SOM in Figure 4. These observations suggest that the neighborhood interaction in the early training phase still had some effects.

Indices such as *homogeneity, separation, silhouette width* and WADP only examine the data themselves and the performance of clustering algorithms with them. They may be categorized as "internal criteria" in the sense of Jain and Dubes ((1988), Chap. 4). On the other hand, the redundant clones present in the NIA microarray provided us with a unique opportunity to evaluate the clustering with some *a priori* knowledge of the data. The redundant score may be categorized as an "external criterion" in Jain and Dubes (1988), although our *a priori* knowledge was only about a small subset of the genes. The current redundant clones were randomly generated during the clone screening processes, it may be more desirable to intentionally include duplicated gene representations in the design of microarray.

There is no single "best" clustering method for all possible data sets, or for all quality measures, different clustering algorithms have different features and properties. The appropriateness of a particular algorithm is dependent on the nature of the data. For example, PAM uses representative objects (medoids) instead of means to represent cluster centers. It can handle data sets in which only (dis)similarity between objects is defined but not the mean of objects. A drawback is that the S-plus implementation is very slow. As a referee pointed out, there is a much faster C-implementation of PAM written by Jenny Bryan, who is now at University of British Columbia. If the data themselves contain a hierarchical structure, hierarchical clustering methods will be more appropriate - partition algorithms, such as k-means, will not be able to capture this type of information. A good feature of SOM is that clusters are represented by nodes arranged in a topological order correlated to the similarity of the clusters. Thus, it is easier for one to observe relations between clusters. This feature is particularly valuable to achieve "soft" clustering when the data are distributed diffusely and cannot be clearly segregated into isolated groups. Of course, the payoff for this SOM feature is that clusters tend to be less compact than those of an algorithm without the topological constraint.

In addition, the choice of algorithms depends on the information sought. For example, k-means and PAM tend to produce "spherically" shaped clusters. This property may be desirable for clustering gene expression profiles to find co-expressed genes, because all the genes in a "spherical" cluster have sufficient pairwise similarity, while the expression profiles of genes at the ends of an elongated cluster may be quite different.

Of course there are many clustering algorithms including refinements and extensions of the basic ones investigated here. Proposals and attempts have also been made to combine the strength of different algorithms. For example, one can use k-means or SOM to obtain gross partitions of data, then use hierarchical clustering to refine each of them. Or, conversely, one can use k-means or SOM to obtain many small clusters and then use hierarchical clustering to identify the connection between those small clusters.

In any event, caution is required, as different algorithms tend to produce somewhat different clusters. This is, on one hand, due to the nature of the present data. On the other hand, it is due to the fact that these algorithms form exhaustive and mutually exclusive clusters that are locally optimal. (Similar problems are addressed by Goldstein, Ghosh and Conlon in this issue of the journal, although they focus on clustering tissues (arrays)). Therefore, when we examined the relations between genes, we did not limit ourselves to the cluster boundaries forced by these algorithms, but also examined the expression profiles of the genes in "similar" clusters nearby. For example, it is known that the expression of Rex-1 is enhanced by Oct3. As shown in Table 1 and Figure 7(a), although Rex-1 was not grouped with Oct-3/4, its expression pattern appeared to be more similar to Oct-3/4 than Hmg1. It is likely that Oct-3/4 was near the boundary of a cluster, e.g., #27 for k-means, and Rex-1 was located in an adjacent cluster. It was informative to see that SOM_r_1 assigned Oct-3/4 to cluster #25, which was between cluster #19 and #31 in the SOM lattice.

In conclusion, cluster analysis requires experience and knowledge about the behavior of clustering algorithms, and can benefit from *a priori* knowledge about the data and underlying biological processes. When *a priori* knowledge about the data is not available or insufficient, it may be desirable to try different algorithms to explore the data and get meaningful clustering results through comparisons.

Acknowledgement

The authors wish to thank Michael Radmacher and Yidong Chen for providing their S-plus script to calculate WADP. The editor and two anonymous referees provided useful comments. This work is supported by NIH grants GM60513 and DA13748.

References

Ben-Shushan, E., Thompson, J. R., Gudas, L. J. and Bergman, Y. (1998). Rex-1, a gene encoding a transcription factor expressed in the early embryo, is regulated via Oct-3/4 and Oct-6 binding to an octamer site and a novel protein, Rox-1, binding to an adjacent site. *Mol. Cell. Biol.* **18**, 1866-1878.

Bittner, M., Meltzer, P., Chen, Y., Jiang, Y., Seftor, E., Hendrix, M., Radmacher, M., Simon, R., Yakhini, Z., Ben-Dor, A., Sampas, N., Dougherty, E., Wang, E., Marincola, F., Gooden, C., Lueders, J., Glatfelter, A., Pollock, P., Carpten, J., Gillanders, E., Leja, D., Dietrich, K., Beaudry, C., Berens, M., Alberts, D., Sondak, V., Hayward, N. and Trent, J. (2000). Molecular classification of cutaneous malignant melanoma by gene expression profiling. *Nature* **406**, 536-540.

Butteroni, C., De Felici, M., Scholer, H. R. and Pesce, M. (2000). Phage display screening reveals an association between germline-specific transcription factor Oct-4 and multiple cellular proteins. *J. Mol. Biol.* **304**, 529-540.

Chui, D. H., Tang, W. and Orkin, S. H. (1995). cDNA cloning of murine Nrf 2 gene, coding for a p45 NF-E2 related transcription factor. *Biochem. Biophys Res. Commun* **209**, 40-46.

Eisen, M. B., Spellman, P. T., Brown, P. O. and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA* **95**, 14863-14868.

Goldstein, D. R., Ghosh, D. and Conlon, E. (2002). Statistical issues in the clustering of gene expression data. *Statist. Sinica* **12**, 219-240

Hartigan, J. A. and Wong, M. A. (1979). A k-means clustering algorithm. *Appl. Statist.* **28**, 100-108.

Jain, A. K. and Dubes, R. C. (1988). *Algorithms for Clustering Data.* Prentice Hall, Englewood Cliffs, NJ.

Jaradat, S. A., Tanaka, T. S., O'Neill, L., Chen, G., Banerjee, N., Zhang, M. Q., Boheler, K. R. and Ko, M. S. H. (2001). Microarray analysis of the genetic reprogramming of mouse ES cells during differentiation. *Keystone Symposium on Pluripotent Stem Cells: Biology and Application.* Durango, Colorado.

Kargul, G. J., Dudekula, D. B., Qian, Y., Lim, M. K., Jaradat, S. A., Tanaka, T. S., Carter, M. G. and Ko, M. S. H. (2001). Verification and initial annotation of the NIA mouse 15K cDNA clone set. *Nat. Genet* **28**, 17-18

Kaufman, L and Rousseeuw, P. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis.* John Wiley, New York.

MathSoft, Inc. (1998). *S-Plus* 5 *for UNIX Guide to Statistics.* Data Analysis Products Division, MathSoft, Seattle.

Pesce, M. and Scholer, H. R. (2000). Oct-4: control of totipotency and germline determination. *Mol. Reprod Dev.* **55**, 452-457.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Computat. Appl. Math.* **20**, 53-65.

Satijn, D. P. and Otte, A. P. (1999). Polycomb group protein complexes: do different complexes regulate distinct target genes? *Biochim. Biophys Acta.* **1447**, 1-16.

Schneider, P., MacKay, F., Steiner, V., Hofmann, K., Bodmer, J. L., Holler, N., Ambrose, C., Lawton, P., Bixler, S., Acha-Orbea, H., Valmori, D., Romero, P., Werner-Favre, C., Zubler, R. H., Browning, J. L. and Tschopp, J. (1999). BAFF, a novel ligand of the tumor necrosis factor family, stimulates *B* cell growth. *J. Experiment. Med.* **189**, 1747-56.

Shamir, R. and Sharan, R. (in press). *Algorithmic Approaches to Clustering Gene Expression Data.* Current Topics in Computational Molecular Biology, MIT Press, Boston, MA.

Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E. S. and Golub, T. R. (1999). Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *Proc. Natl. Acad. Sci. USA* **96**, 2907-12.

Tanaka, T. S., Jaradat, S. A., Lim, M. K., Kargul, G. J., Wang, X., Grahovac, M. J., Pantano, S., Sano, Y., Piao, Y., Nagaraja, R., Doi, H., Wood, W. H., 3rd, Becker, K. G. and Ko, M. S. (2000). Genome-wide expression profiling of mid-gestation placenta and embryo using a 15, 000 mouse developmental cDNA microarray. *Proc. Natl. Acad. Sci. USA* **97**, 9127-32.

Tavazoie, S., Hughes, J. D., Campbell, M. J., Cho, R. J. and Church, G. M. (1999). Systematic determination of genetic network architecture. *Nat. Genet* **22**, 281-5.

Vilo, J., Brazma, A., Jonassen, I., Robinson, A. and Ukkonen, E. (2000). Mining for putative regulatory elements in the yeast genome using gene expression data. *Ismb* **8**, 384-394.

Yeung, K. Y., Haynor, D. R. and Ruzzo, W. L. (2000). Validating clustering for gene expression data. *Bioinformatics* **17**, 309-318.

Cold Spring Harbor Laboratory 1 Bungtown Rood Hershey Building, Cold Spring Harbor, NY 11724, U.S.A.

E-mail: cheng@cshl.org

Laboratories of Genetics, National Instituteon Aging, National Institutes of Health, Baltimore, MD 21224, U.S.A.

E-mail: jaradats@grc.nia.nih.gov

Cold Spring Harbor Laboratory 1 Bungtown Rood Hershey Building, Cold Spring Harbor, NY 11724, U.S.A.

E-mail: banerjee@cshl.org

Laboratories of Genetics, National Instituteon Aging, National Institutes of Health, Baltimore, MD 21224, U.S.A.

E-mail: tanakat@grc.nia.nih.gov

Laboratories of Genetics, National Instituteon Aging, National Institutes of Health, Baltimore, MD 21224, U.S.A.

E-mail: kom@grc.nia.nih.gov

Cold Spring Harbor Laboratory 1 Bungtown Rood Hershey Building, Cold Spring Harbor, NY 11724, U.S.A.

E-mail: mzhang@cshl.org

(Received March 2001; accepted November 2001)