Bioinformation
www.bioinformation.net

open access
Hypothesis

Effective feature selection framework for cluster analysis of microarray data

Gouchol Pok[1], Jyh-Charn Steve Liu[2], Keun Ho Ryu[3]*

[1]Yanbian University of science and Technology, Dept. of Computer Science, Yanji, Jilin, China 133000; [2]Texas A&M University, Dept. of Computer Science, College Station, TX, USA; [3]Chungbuk National University, DB Bioinformatics Lab, Cheongju, Chungbuk, Korea. Keun Ho Ryu – E-mail: khryu@dblab.chungbuk.ac.kr, *Corresponding Author

Received May 25, 2009; Revised February 18, 2010; Accepted February 24, 2010; Published February 28, 2010

Abstract:
The microarray technique has become a standard means in simultaneously examining expression of all genes measured in different circumstances. As microarray data are typically characterized by high dimensional features with a small number of samples, feature selection needs to be incorporated to identify a subset of genes that are meaningful for biological interpretation and accountable for the sample variation. In this article, we present a simple, yet effective feature selection framework suitable for two-dimensional microarray data. Our correlation-based, nonparametric approach allows compact representation of class-specific properties with a small number of genes. We evaluated our method using publicly available experimental data and obtained favorable results.

Keywords: gene expression microarray, feature selection, classification, clustering

Background:
Recently feature selection has become an essential process to handle the high dimensional nature of biological data such as gene expression microarrays. The main objective of feature selection, in particular for the gene expression data analysis, is to identify a subset of features without deforming the original representation or distorting the interpretability **[1].** This allows the subset to retain sufficient information in explaining the underlying biological system behaviour like cellular function and pathways **[2].** Therefore, feature selection differs from other conventional dimension reduction techniques such as the projection-based principle component analysis or the information measure-based approaches, which in general do not provide a way to recover the original biological meaning from the reduced features **[1].** In summary, the gene expression data are characterized by the following issues: 1) obviously microarrays are of high dimension, with thousands of genes involved, 2) measured samples or experiments are very few, typically less than 100, and 3) among thousands of gene expressions, only a few of them account for the data variation **[3].**

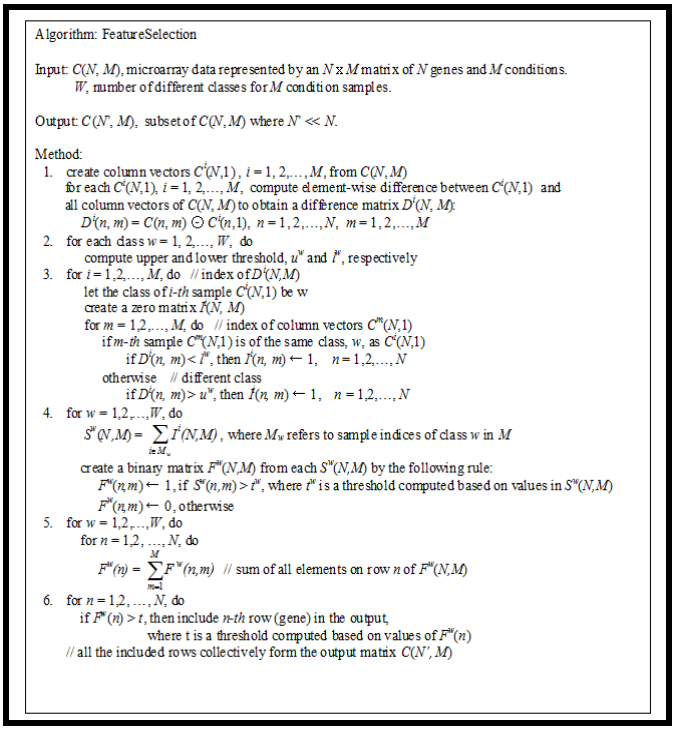
A lot of data mining and pattern recognition techniques have been applied to capture the meaningful patterns in gene expression microarrays. A straightforward approach is to apply standard statistical methods: using the t-test **[4**], the Bayesian approach **[5]**, and the Wilcoxon rank sum test **[6]**. All these methods are the univariate feature selection method which ignores the dependency between features. To take into account the correlation between genes, multivariate models have been developed including exploring bivariate interactions, correlation-based feature selection, the Markov blanket filter method. These filter-based approaches focus only on the general properties of the data itself without considering the associated classifier in evaluating the selected features. The wrapper approach, on the other hand, integrates the feature selection process with the evaluation of the selected features by a classifier.

In the machine learning world, it is well known that classification/clustering performance could be degraded when the selected features include irrelevant and redundant information **[7]**. Redundancy-removed feature subset allows avoiding overfitting and hence improves the performance of the applied model **[1].** Microarray data presumably also include some gene expressions that are not related to the classification task at hand. In the past, feature subset selection has been extensively studied **[1, 8, 9, 10]**. Two key issues concerning feature subset selection are 1) how to evaluate selected features and 2) how to perform search, except determining the starting and stopping conditions.

Feature evaluation methods are further divided into groups based on modeling strategies, namely filter, wrapper, and embedded techniques **[1].** Filter techniques compute statistical correlations between gene expressions and sample classes. Selection of genes is computationally simple and fast to implement, easily scalable to high-dimensional data, and independent of classification algorithm. Filter techniques in general, however, evaluate the features one by one to determine their relevance to the classification task. As a result, they can not provide correlative information between two sets of gene expression, which would be valuable in selecting a biologically meaningful subset of genes. Wrapper methods integrate the feature search procedure with the classifier training in such a way that evaluation of feature subset is performed in accordance with classifier testing. This interaction between feature selection and classifier design enables to consider feature dependencies **[11].** In the context of finding maximally influencing genes from microarray data, though taking different strategies, all feature subset selection methods are concerned with handling the following issues: (1) Removal of irrelevancy; (2) Removal of redundancy; (3) Maintaining of class-discriminating power.

In this paper we present a simple, yet efficient feature selection framework which is appropriate to extract class-specific properties as a small number of genes from two-dimensional microarray data. Our feature selection method addresses the key issue in feature selection: removal of irrelevance and redundancy without performance degradation. The main contribution of our approach, however, lies in providing a way to identify what features (gene expressions and samples) characterize each class. This is somewhat different from just identifying the sample phenotypes in different classes, in the sense that we consider the genes and samples at the same time in selecting the most influential genes. In other words, the class-specific features retain the original matrix form, and hence it is easy to see which genes are related with which samples in each class. Another advantageous point is that our method can be implemented without great effort.

In spite of its simple process, however, our method allows the interpretable process of gene selection as well as superior performance in classification. Simulation with the widely used benchmark microarray data shows that our method yield compact representation of gene expressions (less than 100 out of thousands) while the results are favourably compared with the published results **[3, 12].**



Algorithm: FeatureSelection

Input: $C(N, M)$, microarray data represented by an N x M matrix of N genes and M conditions.
W, number of different classes for M condition samples.

Output: $C(N', M)$, subset of $C(N, M)$ where $N' \ll N$.

Method:

1. create column vectors $C^i(N,1)$, $i = 1, 2, \ldots, M$, from $C(N,M)$
   for each $C^i(N,1)$, $i = 1, 2, \ldots, M$, compute element-wise difference between $C^i(N,1)$ and all column vectors of $C(N, M)$ to obtain a difference matrix $D^i(N, M)$:
   $D^i(n, m) = C(n, m) \ominus C^i(n,1)$, $n = 1, 2, \ldots, N$, $m = 1, 2, \ldots, M$
2. for each class $w = 1, 2, \ldots, W$, do
   compute upper and lower threshold, u^w and l^w, respectively
3. for $i = 1, 2, \ldots, M$, do // index of $D^i(N,M)$
   let the class of i-th sample $C^i(N,1)$ be w
   create a zero matrix $I^i(N, M)$
   for $m = 1, 2, \ldots, M$, do // index of column vectors $C^m(N,1)$
   if m-th sample $C^m(N,1)$ is of the same class, w, as $C^i(N,1)$
   if $D^i(n, m) < l^w$, then $I^i(n, m) \leftarrow 1$, $n = 1, 2, \ldots, N$
   otherwise // different class
   if $D^i(n, m) > u^w$, then $I^i(n, m) \leftarrow 1$, $n = 1, 2, \ldots, N$
4. for $w = 1, 2, \ldots, W$, do
   $S^w(N,M) = \sum_{i \in M_w} I^i(N,M)$, where M_w refers to sample indices of class w in M
   create a binary matrix $F^w(N,M)$ from each $S^w(N,M)$ by the following rule:
   $F^w(n,m) \leftarrow 1$, if $S^w(n,m) > t^w$, where t^w is a threshold computed based on values in $S^w(N,M)$
   $F^w(n,m) \leftarrow 0$, otherwise
5. for $w = 1, 2, \ldots, W$, do
   for $n = 1, 2, \ldots, N$, do
   $F^w(n) = \sum_{m=1}^{M} F^w(n,m)$ // sum of all elements on row n of $F^w(N,M)$
6. for $n = 1, 2, \ldots, N$, do
   if $F^w(n) > t$, then include n-th row (gene) in the output,
   where t is a threshold computed based on values of $F^w(n)$
   // all the included rows collectively form the output matrix $C(N', M)$

Figure 1: A feature subset selection algorithm that is specifically targeted for 2-D microarray data.

Methodology:
See supplemenatary material for methdology.

Discussion:
We tested the feature selection method using the publicly available data of three different cancer types: acute leukemia, medulloblastoma, central nervous system tumors **[12].** All simulations were carried out using MATLAB® software on a 2.80GHz Petium-4 workstation. To verify the effectiveness of the proposed approach, we compare the experimental results with those obtained by the basic NMF [13] and those by the sparse NMF [5]. We measured the accuracy of the clustering by the formula given in equaltion 7 **(see supplementary material).** Note that our goal here is not just to apply a black box-type classifier aiming for a good result without reasoning but to find how well samples are grouped into compactly formed clusters, which would be useful for further analysis. The details of the experiments over each data set are as follows.

Leukemia dataset
This data set is widely used as a benchmark in the cancer classification to compare various methods **[12]**. The classification task with this data set is to discriminate acute lymphoblastic leukemia (ALL) type from acute myelogenous leukemia (AML) type, and, within the ALL type, to classify ALL-T cell subtype from ALL-B subtype. The data set is composed of 5000 genes from 38 bone marrow samples: 19 samples of ALL- B type, 8 samples of ALL-T type, and 11 samples of AML type. Thus, this data set poses two classification problems: 1) distinction of AML and ALL types and 2) distinction of all subtypes of AML, ALL-B, and ALL-T. The first problem can be relatively easily addressed by using SOM or hierarchical clustering (HC), though some kind of tuning of parameters (such as number of clusters and number of input genes) is required to get optimal solutions **[3, 12].** For the second problem, however, distribution of samples seemingly does not form compact and distinct clusters. Rather, depending on the used metric and inputs (for HC) or the starting condition (for SOM), these methods yield varying and unstable classification results. Brunet et al. **[12]** reported that basic NMF recovered successfully the cluster structure intrinsic to the data for both problems. Later Gao and Church **[3]** presented the sparse NMF (SNMF) and reportedly improved the classification accuracy, measured by Equation (7), from 0.947 (2 incorrectly classified out of 38) using basic NMF to 0.974 (1 incorrect out of 38) using SNMF. Our goal here is first to identify most influencing subset of genes from leukemia data in such a way that they retain the class- specific features while discarding irrelevant gene features, and then to show that performance can be improved or favorable compared to the results above. Recall that the NMF algorithm decomposes the gene expression data, $C(N',M)$, into $V\,H$ where H has dimension of $\kappa \times M$, where κ clusters of samples are formed. Utilizing this property, we can see if each sample is correctly classified into well-defined clusters by examining the maximum value in each column of H. More specifically, if a sample belongs to class w and has the max value at column w, then it is correctly classified, and otherwise not. Using the feature selection method described in Method section, we have selected 64 genes. The class-specific feature selection criteria as described at step 7 of Algorithm in Figure 1 are set to as follows: $F^1(\cdot) > 23$, $F^2(\cdot) > 26$, and $F^3(\cdot) > 26$. The results of 50 runs of the NMF algorithm with κ set to 3 consistently show that all the samples except one have been correctly classified. The incorrectly recognized sample is the 29th sample annotated as "AML 13" in the original data set, which is classified as ALL-B type. This result is similar to that of Gao and Church [3], but somewhat different from the result of Brunet et al. **[12]** in which two incorrectly recognized samples are the 6-th and the 10-th, both of type ALL-B. Investigating the source from which this discrepancy comes seems to be an interesting, but challenging task for the future work.

Medulloblastoma dataset
The medulloblastoma data set contains 34 samples related to childhood brain tumors. Although pathogenesis of medulloblastoma is not well understood and its diagnosis is highly subjective, due to some attributes observable under the microscope, medulloblastoma could be divided into two sub-classes, classic and desmoplastic **[12, 16]**. The samples are composed of 25 classic and 9 desmoplastic medulloblastoma. SOM and HC can not recognize the clustering

topology intrinsic to this two-class data. The basic NMF successfully captured the distinctiveness between two classes for $\kappa = 2$ to 5, but two desmoplastic and one classic sample are incorrectly classified. When the SNMF was applied to this data set, though it captured the intrinsic clustering structure, classification performance was not satisfactory: only three out of nine desmoplastic samples were correctly classified. Considering that SNMF seeks sparse representation of genes, which is equivalent to the objective of our approach by and large, comparison of our method's performance against that of SNMF would be meaningful. However, comprehensive comparison is difficult because Gao and Church **[3]** only described the outline of the SNMF algorithm and the implication of sparseness, without providing any information about the degree of data sparseness used (or how compactly the data are reduced). Using the feature selection method, we have selected 74 genes out of 5893 in the data set. In this case, the class-specific feature selection criteria specified at step 7 of Algorithm are set to as follows: $F^1(\cdot) > 14$ and $F^2(\cdot) > 10$. These values are a little bit smaller compared to those in the leukemia data case. By varying the value of κ from 2 to 5, we carried out several tests and observed the following results. For $\kappa = 2$, the experiment consistently yielded three incorrectly classified sample, all belonging to classic medulloblastoma. For $\kappa = 3$, the experiment also yielded three incorrectly classified sample, two from classic and one from desmoplastic medulloblastoma. For $\kappa = 4$ and 5, only one classic medulloblastoma, the 6-th sample annotated as "Brain MD 49", was incorrectly classified. This sample was consistently misclassified across all the experiments. These results, compared with others above, are very suggestive of our method's effectiveness in feature selection.

Central Nervous System Tumors Data

The data set is composed of 34 samples of central nervous system embryonal tumors **[16].** The samples come from four different types: 10 classic medulloblastomas, 10 malignant gliomas, 10 rhabdoids, and 4 normals. The basic NMF method well captured four-class clustering structure with two misclassifications: the 18-th sample ("Brain Rhab 10") of glioma type as a rhabdoids and the 30-th sample ("Brain MGlio 8") as normal. The SNMF algorithm showed similar results with only one (the 18-th) sample misclassified as a rhabdoid. We have selected 97 genes out of 7129 in the data set. The class-specific feature selection criteria are set to as follows: $F^1(\cdot) > 24$, $F^2(\cdot) > 23$, $F^3(\cdot) > 22$, and $F^4(\cdot) > 24$. We applied the basic NMF to this set, getting one misclassification for the 18-th sample. We note that the 18-th sample is persistently misclassified as a rhabdoid across all the three methods.

Class-Specific Features

In our feature selection method, the features that are specific to class w is saved in $F^w(N,M)$ as described at *step5* of Algorithm in Figure 1. $F^w(N,M)$ is a binary-valued matrix in which values of 1 refers to feature components distinguished for class w. Therefore, for any two different classes w_1 and w_2, the lesser common entries of value 1 $F^{w1}(N,M)$ and $F^{w2}(N,M)$ have, the better discrimination between w_1 and w_2 we expect. Table 2 (**see supplementary material**) shows that the feature matrices obtained from the leukemia data share few common elements of value 1. F^w denotes the number of 1-valued elements in $F^w(N,M)$, and $F^i - F^j$ donotes the number of elements that have 1 for both $F^i(N,M)$ and $F^j(N,M)$, and so on. In Table 1 (**see supplementary material)**, one can see that 1-valued elements that are commonly occurred for any two classes are on the average about 10% of the total number of 1-valued elements. This clearly explains why our method is effective in capturing class-specific features. For the case of medulloblastoma data, the features of two classes are more uncorrelated as shown in Table 2 (**see supplementary material**). Only 72 out of over 20,000 elements are marked as common. Table 3 (**see supplementary material**) shows the statistics for the central nervous system tumors data, where "Avg." refers to the average value of four $F^w(N,M)$'s. Three-class common membership statistics show similar behavior (10% on the average) as other data sets above, except for the $F^2 - F^3 - F^4$ case where the number of common elements is remarkably small. This explains that the three classes are expectedly well separated from each other.

Conclusion:

We present a feature subset selection framework that is effective in selecting a subset of influencing genes from microarray data. The proposed method provides an explicit representation of class-specific features. This scheme will be useful to identify biologically meaningful genes associated with a certain diagnosis. Our approach is distinct from typical dimension reduction methods that do not consider preserving the unit property of individual features in the reduced representation. Typical dimension reduction of microarray data is carried out either by reducing only the number of rows (gene expression levels) or by creating new reduced dimensions without considering the unity of original features, such as employed by PCA. In this work, we approached row-wise dimension reduction by using feature selection technique, while applying the clustering technique for column-wise reduction. One point to note about this work and most other existing works is that the dimension reduction in row-wise and column-wise direction is not coordinated with each other. The next step of our work will be directed to the coordination scheme in which selection of genes is well coordinated with identification of sample phenotypes characterizing each class based on the selected genes.

Acknowledgements :

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF No. 2009-0063257 and NRF No. R01-2007-000-10926-0), and by the Korean Ministry of Education, Science and Technology(The Regional Core Research Program / Chungbuk BIT Research-Oriented University Consortium).

Reference :

[1] Y. Saeys & I. Inza, *Bioinformatics*, **23**:2507 (2007) [PMID : 17720704]
[2] T. Moloshok *et al*., *Bioinformatics*, **18**:566 (2002) [PMID : 12016054]
[3] Y. Gao & G. Church, *Bioinformatics,* **21**:3970 (2005) [PMID : 16244221]
[4] C. Tsai *et al*., *Nucl. Acids. Res.*, **31**:e52 (2003) [PMID : 12711697]
[5] P. Baldi & A. Long, *Bioinformatics*, **17**:509 (2001) [PMID: 11395427]
[6] Anatoniadis *et al*., *Bioinformatics,* **19**:563 (2003) [PMID: 12651713]
[7] W. Kuo *et al*., J *Biomed. Inform.*, **37**:293 (2004) [PMID : 15465482]
[8] D. Hwang *et al*., *Bioinformatics,* **18**:1184 (2002) [PMID: 12217910]
[9] J. Liu *et al*., *Bioinformatics*, **24**:i86 (2008) [PMID : 18586749]
[10] C. Sima & E. Dougherty, *Bioinformatics*, **22**:2430 (2006) [PMID : 16870934]
[11] Inza *et al*., *Artif. Intelli. Med.*, **31**:91 (2004) [PMID : 15219288]
[12] J.-P. Brunet *et al*., *PNAS*, **101**:4164 (2004) [PMID : 15016911]
[13] Devarajan, PLOS *Comp. Biol.*, **4**:1 (2008) [PMID : 18654623]
[14] D. Lee & S. Seung, *Nature,* **401**:788 (1999) [PMID : 10548103]
[15] X. Ge & S. Iwata, *Neural Netw.*, **15**:285 (2002) [PMID : 12022515]
[16] S. Pomeroy *et al.*, *Nature*, **415**:436 (2002) [PMID : 11807556]

Edited by Tan Tin Wee
Citation: Pok *et al.*, Bioinformation 4(8): 385-389 (2010)

Bioinformation
www.bioinformation.net

open access
Hypothesis

Supplementary Material:

Methdology :

Feature Selection:

Overall algorithmic steps of our feature subset selection method are illustrated in **Figure 1**. We denote as $C(N,M)$ a microarray data in a matrix form of N gene expression levels and M samples of experiment conditions. Let $C^i(N, 1)$ denote i-th column vector of $C(N,M)$. Referring to Figure 1, the details of the feature selection steps are as follows.

Step 1: First, column vectors $C^i(N, 1)$, $i = 1, 2,\ldots, M$, are created from $C(N,M)$. Each of these column vectors actually corresponds to one sample of gene expression data. Each column vector $C^i(N, 1)$ is fed into a difference operation (denoted as $\circ$) which computes the element-wise difference between $C^i(N, 1)$ and all the columns of $C(N,M)$. For each i, this operation outputs a difference matrix,

$$D^i(N,M) = C(N,M) \circ C^i(N, 1), \quad i = 1,2,..., M, \tag{1}$$

where each entry of $Di(N;M)$ is computed as the difference between two real numbers,

$$D^i(r, c) = C(r, c) - C^i(r, 1), \quad r = 1,2,\ldots,N, \quad c = 1,2,\ldots,M. \tag{2}$$

Let $D(N, j)$ denote j-th column of matrix $D(N,M)$. By definition, $D^i(N, j)$ contains the measure about how i-th sample $C^i(N, 1)$ differs from j-th sample $C^j(N, 1)$. One advantage of this vector-based scheme is that between-gene difference information is also kept in the column vectors together with between-sample differences. This is useful in examining how all the genes are correlated as will be shown shortly.

Step 2: Magnitude of elements of $D^i(N,M)$ can be used as a measure to determine how useful each gene (row) is in classification. Our strategy here is based on a simple idea: differences between two samples in the same class will be small for most genes, while two samples coming from different classes will show large differences for many genes. Our objective is to identify and select out those genes that behave according to the conjecture. In order to mark the genes which take big or small values in $D^i(N,M)$, we introduce upper threshold u and lower threshold l, which are set to 75-percentile and 25-percentile, respectively, of the values in $D^i(N,M)$. As the range of difference values could be varying depending on class, thresholds are determined using the values within a class and represented as u^w and l^w for class w.

Step 3: Marking of the values in $D^i(N,M)$ is carried out as follows. For an element $D^i(n,m)$ given, if i-th sample and m-th sample belong to the same class, then the absolute value of $D^i(n,m)$ would be expectedly small for gene n. Otherwise if they belong to different classes, then $D^i(n,m)$ would take a large value for gene n. In any case, if this expectation is met for gene n, we mark the gene by setting $I^i(n,m)$ to 1. This marking implies that n-th gene is useful for describing the (inverse) correlation between i-th to m-th sample in terms of classification. Also it should be noted that this naturally provides hints about where irrelevant features arise. However, final decision on usefulness of a gene in classification should be postponed until the gene proves to be useful for all the samples involved, which is taken care of in the next step.

Step 4: The objective of this step is to construct class-specific features from a set $\{ I^i(N,M), i = 1,2, \ldots, M\}$, which holds individual sample-based information. As we assume M samples are collected from W different classes, M columns can be decomposed into a partition of W blocks,

$$I^i(N, M) = I^i(N, M_1 + M_2 + ... + M_W), \tag{3}$$

where M_w refers to the number of samples in class w. With this scheme, for each w, we element-wise add $I^i(N,M)$'s within class w to get $S^w(N, M)$. For example, if we suppose M_2 consists of 3 samples indexed from 5 to 7, the features specific to class 2 are computed by:

$S^2(n,m) = I^5(n,m) + I^6(n,m) + I^7(n,m), \quad n = 1,2,...,N, \quad m = 1,2,...,M.$ Once we have constructed all $S^w(N,M)$'s, we identify the elements taking significant values by applying a threshold which is set to 90-percentile of the values in $S^w(N,M)$ for each class w. This threshold operation produces a binary matrix $F^w(N, M)$. It should be noted that $F^w(N, M)$ holds a useful measure for determining how much each gene n contributes to the classification of each sample m.

Step 5: After we collect the within-class features in $F^w(N, M)$, we then move on to selecting the most influencing genes. Selection of genes at this step is rather trivial, because all useful information has already gathered in $F^w(N, M)$. We just count the marked elements in $F^w(N, M)$ for each gene n, whose value is denoted by $F^w(n)$, and use it as a final measure to determine the usefulness of gene n in the classification task as described in **step 6**.

Clustering of Samples

Gene expression data are typically given without any information about the phenotype of genes within each class. In handling such case of lacking a priori knowledge of representative patterns, nonnegative matrix factorization (NMF) has proved to be successful in capturing biologically meaningful clusters in the unsupervised manner **[3, 12, 13]**. In contrast to holistic methods such as principle component analysis (PCA) and self-organizing map (SOM), NMF yields a sparse, parts-based decomposition of data without discarding the original interpretation of features [14]. Suppose gene expression data is represented as $N \times M$ nonnegative matrix A which is $C(N', M)$ after feature selection. The number N of genes is usually in the thousands. NMF method decomposes A into two nonnegative matrices, V of size $N \times \kappa$ and H of size $\kappa \times M$, so that $A \sim V H$. The rank κ of factorization defines the number of metagenes, which reflects the degree of latent factors. In a classification scheme, the value of κ represents the number of clusters, and the goal of NMF is to find two nonnegative matrices V and H such that · clusters optimally characterize the intrinsic structure of samples in A. The NMF algorithm starts by initializing V and H to random values and iteratively updates their values to minimize the distance between A and $V H$. A number of the divergence functionals have been proposed to measure the distance, including Euclidian distance and Kullback-Leibler (KL) divergence [12, 13, 14]. The KL divergence functional is given by the Poisson likelihood of generating A from V and H,

$$KL(A \| VH) = \sum [A_{ij} \log \frac{A_{ij}}{(VH)_{ij}} - A_{ij} + (VH)_{ij}] \quad (4)$$

The divergence $KL(\cdot)$ is non-increasing under the following multiplicative update rules [15],

$$H_{a\mu} \leftarrow H_{a\mu} \frac{\sum_i V_{ia} A_{i\mu} / (VH)_{i\mu}}{\sum_k V_{ka}} \quad (5)$$

$$V_{ia} \leftarrow V_{ia} \frac{\sum_\mu H_{a\mu} A_{i\mu} / (VH)_{i\mu}}{\sum_\nu H_{a\nu}} \quad (6)$$

The updates of the two matrices are iteratively performed until the divergence of Equation (4) converges to a (local) minimum. Each sample is then considered to determine its membership to one of the κ clusters by the highest value of metagene expression pattern (column of H).

$$q = \sum_{j=1}^{M} \delta(j) \Big/ M, \quad (7)$$

where $\delta(j)$ is 1 if j-th sample is correctly classified and 0 otherwise.

Table 1: Number of 1-valued elements in $F^w(N,M)$ for the Leukemia data. F^i - F^j denotes the number of elements whose values take 1 both in class i and j. Also F^i - F^j - F^k denotes the number of elements that have 1 for all three classes i, j, and k.

F^1	F^2	F^3	F^1 - F^2	F^1 - F^3	F^2 - F^3	F^1 - F^2 - F^3
23,115	22,758	24,785	1,640	1,127	4,468	7

Table 2: Number of 1-valued elements in $F^w(N,M)$ for the Medulloblastoma data

F^1	F^2	F^1 - F^2
22,665	20,913	72

Table 3: Number of 1-valued elements in $F^w(N,M)$ for the Central nervous system tumors data. The value in the first column is the average of four F^w's.

Avg. of F^w's	F^1 - F^2 - F^3	F^1 - F^2 – F^4	F^1 – F^3 – F^4	F^2 – F^3 – F^4	F^1 - F^2 - F^3 - F^4
26,710	3,627	1,829	2,682	210	98